ADVENTURE e commodore 64

EDIZIONE ITALIANA

Mike Grace

GRUPPO EDITORIALE JACKSON

ADVENTURE e commodore 64

Mike Grace

GRUPPO
EDITORIALE
JACKSON

Copyright per l'edizione originale:
Sunshine Books - 1983
Titolo originale: COMMODORE 64 ADVENTURES

Copyright per l'edizione italiana:
GRUPPO EDITORIALE JACKSON - Febbraio 1985

TRADUZIONE: Flavio Brunelli
GRAFICA E IMPAGINAZIONE: Cristina De Venezia
SUPERVISIONE TECNICA: Daria Gianni
REVISIONI E PROGRAMMI: Emilio Delle Piane
COPERTINA: Marcello Longhini
FOTOCOMPOSIZIONE: SYSTEM GRAPHIC - Cologno Monzese (MI)
STAMPA: Centro Grafico Milano - Via A. Gatto, 14 - Ponte Sesto di Rozzano (MI)

Tutti i diritti sono riservati. Stampato in Italia. Nessuna parte di questo libro può essere riprodotta, memorizzata in sistemi di archivio, o trasmessa in qualsiasi forma o mezzo, elettronico, meccanico, fotocopia, registrazione o altri senza la preventiva autorizzazione scritta dell'editore.

Nota di Programmazione

Un certo numero di funzioni del Commodore 64, come di altre macchine Commodore, viene selezionato dal "carattere di controllo", contenuto nelle stringhe ordinarie, che diventa operativo quando la strigna viene stampata. I caratteri di controllo possono di solito essere riconosciuti dal fatto che essi sono caratteri inversi (il loro colore e quello dello sfondo sono scambiati).
Le funzioni con cui si controllano tali caratteri comprendono la posizione del cursore, i colori di stampa, inverso (RVS) ON ed OFF, "cursore home" (HOME) e "pulisci lo schermo" (CLR).

Nero		**Verde chiaro**	
Bianco		**Azzurro**	
Rosso		**Grigio 3**	
Cyan		**RVS on**	
Porpora		**RVS off**	
Verde		**Su**	
Blu		**Giù**	
Giallo		**Destra**	
Arancio		**Sinistra**	
Marrone			
Rosso chiaro			
Grigio 1			
Grigio 2			

INDICE

Introduzione all'edizione italiana IX

Prefazione XI

PARTE I 1

CAPITOLO 1
Il mistero e la magia 3

CAPITOLO 2
Muoversi in un'Adventure 11

CAPITOLO 3
Oggetti e comandi 27

CAPITOLO 4
Trabocchetti e Pericoli 43

PARTE II 51

CAPITOLO 5
La stesura della trama 53

CAPITOLO 6
Definiamo la Scena 65

CAPITOLO 7
Serve aiuto? 73

CAPITOLO 8
Sprite ed Effetti Sonori 81

CAPITOLO 9
Dove si va da qui? 99

CAPITOLO 10
Prendiamolo proprio ora! 117

CAPITOLO 11
Non per quella via 129

CAPITOLO 12
Bella giornata per un tuffo 141

CAPITOLO 13
Discorrendo di amenità 157

CAPITOLO 14
Un bacio tempestivo 169

CAPITOLO 15
Guardando al futuro 179

Conclusione 180

APPENDICI 181

APPENDICE A
Tabella delle variabili per "Il Pianeta dell'incubo" 183

APPENDICE B
I numeri di linea 185

APPENDICE C
Diagramma di flusso completo per "Il Pianeta dell'Incubo" 191

APPENDICE D
Programma A completo 193

Programma B completo 201

INDICE DETTAGLIATO

CAPITOLO 1
Il mistero e la Magia
Gli inizi delle Adventure, cos'è un gioco di Adventure, Sotterranei e Draghi, suggerimenti su come giocare un'Adventure, pensiamo a scriverne una.

CAPITOLO 2
Muoversi in un'Adventure
Tracciare e interpretare mappe, conversione di mappe in griglie, alcuni onsigli sulla programmazione, inserite le locazioni nel vostro programma.

CAPITOLO 3
Oggetti e comandi
L'uso delle matrici, l'inserimento di oggetti in griglie, trattamento elementare di stringhe, inventario, routine di "raccolta" e "abbandono" di oggetti.

CAPITOLO 4
Trabocchetti e Pericoli
Aggiungiamo emozioni alla nostra Adventure, la programmazione dei Pericoli, l'uso di variabili per indicare rischi, il superamento delle difficoltà.

CAPITOLO 5
La stesura della trama
Iniziamo il racconto, scelta dell'ambientazione, sclta della meta, selezione dell'eroe, scelta dei personaggi principali, stesura di una sinossi, una mappa semplificata, come mettere in tavole grafichela trama.

CAPITOLO 6
Definiamo la Scena
Concetti iniziali, il progetto del titolo, cominciamo a programmare, linee generali di programmazione di testi.

CAPITOLO 7
Serve aiuto?
Forniamo istruzioni per il gioco, LOAD, e SAVE da nastro.

CAPITOLO 8
Sprite ed Effetti Sonori
Disegnamo e creiamo sprite, il movimento sprite, Victor e Mealy Bug, la selezione degli effetti sonori, l'aggiunta del suono alle risposte.

CAPITOLO 9
Dove si va da qui?
La creazione della mappa per "Il Pianeta dell'Incubo", l'inserimento delle Locazioni, il Movimento nell'Adventure.

CAPITOLO 10
Prendiamolo proprio ora
Airlock e altre cose, la manipolazione di stringhe, l'aggiunta dell'inventario, il raccogliere e lasciare oggetti.

CAPITOLO 11
Non per quella via
I sei pericoli di base, la visualizzazione dei pericoli, come impedire la fuga dalle locazioni.

CAPITOLO 12
Bella giornata per un tuffo
Routine di uccisione, il pericolo anguilla, l'uso di variabili per impedire movimenti, una struttura generale di pericoli in un'Adventure.

CAPITOLO 13
Discorrendo di amenità
Ora vedete Aurora - ora non più, alle prese con le sabbie mobili, un dinosauro e un serpente, scalare alberi, l'aggiunta di loop di ritardo.

CAPITOLO 14
Un bacio tempestivo
Routine di aiuto, sparare e far fuco, trattare il comando "mangia", l'ultimo bacio.

CAPITOLO 15
Guardando al futuro
L'estensione del vostro gioco: è proprio alla vostra portata.

INTRODUZIONE ALL'EDIZIONE ITALIANA

Parecchie caratteristiche rendono i giochi d'Adventure molto diversi dagli altri giochi per Computer, monotematici e in gran parte destinati a saggiare l'abilità manuale ed i riflessi del giocatore, proponendo scenari in cui l'aspetto grafico e sonoro assume importanza rilevante. Queste ultimi, in fin dei conti, arrivano direttamente dalle sale giochi, o si ispirano alle macchine elettroniche dei bar.
I giochi d'Adventure si caratterizzano innanzitutto per lo SCOPO, mettere alla prova le capacità mnemoniche e deduttive del giocatore (così come la pazienza, spesso!), proponendo un obiettivo da raggiungere; quindi per il MEZZO, la presentazione di un evolversi di situazioni in cui il giocatore è coinvolto; infine per l'ASPETTO, di solito straordinariamente sobrio, almeno se si confrontano con i giochi e le battaglie spaziali. In questo ultimo aspetto risentono ancora dell'originale ambito in cui sono nati, i grossi centri di calcolo, di Università o industrie, in cui si configuravano come eccitanti esperimenti destinati a mettere alla frusta le capacità deduttive del giocatore, e a dimostrare l'abilità inventiva dell'ideatore.
Il continuo miglioramento delle caratteristiche di velocità è memoria dei computer, ha permesso l'ingresso dei giochi d'Adventure nel mondo dei Micro, in cui il CBM 64 occupa stabilmente una posizione di predominio.
Il libro di Mike Grace appare quindi in un momento in cui si tende alla realizzazione di giochi più coinvolgenti e personalizzati, e per le sue caratteristiche di leggibilità e gradualità d'approccio al problema, costituisce uno strumento utilissimo per chi voglia avvicinarsi al mondo dell'ideazione e programmazione di Adventure. Anche i programmatori esperti troveranno stimoli e suggerimenti impagabili, specialmente per quanto riguarda la fase di progettazione del gioco e la pianificazione della trama.
Il libro è diviso in due parti. Nella prima, attraverso un esempio introduttivo, il lettore è portato gradualmente alla fase di ideazione e progetto di un'Adventure, prendendo contatto con i concetti di percorsi, griglie, locazioni, oggetti, obiettivi, pericoli, azioni. Contemporaneamente allo svolgersi di questa operazione didattica, vengono presentati i moduli del programma separati, che concludono le varie sezioni tematiche, e di cui si commentano ampiamente le istruzioni.
Alla fine risulta un programma semplice, ma non per questo meno istruttivo.
La seconda parte del libro approfondisce gli aspetti della prima, introducendo anche esempi di interventi della grafica e del suono, e affinando alcune caratteristiche lasciate in ombra nella prima, per rendere più accattivante e appassionante il gioco. È qui che viene progettato e costruito il vero programma di gioco, che alla fine risulterà costituito da circa un migliaio di istruzioni.
L'aspetto più interessante del libro è senz'altro costituito dalla modularità della trattazione, che permette anche al neofita di programmazione di capire il significato delle varie fasi e operazioni, e consente altresì all'esperto di aggiungere, dove lo voglia, gli interventi che reputa più utili per migliorare il gioco, o introdurre nuovi aspetti. L'autore stesso ha previsto queste possibilità, e le incoraggia, anzi.
Il programma di questa seconda parte occupa praticamente tutta la memoria del CBM

64, e quindi eventuali modifiche devono tener conto di ciò. Per facilitare questo tipo di interventi è stata aggiunta l'appendice D, con l'intero listato dei programmi delle due parti del libro, come risultano dalla fusione di tutti i singoli moduli. Questo permette anche di poter inserire in un'unica sessione il programma, anche se questa procedura viene sconsigliata dall'autore, dato il carattere eminentemente didattico del libro.
Ho ritenuto opportuno tradurre i messaggi del programma, spesso idiomatici nell'originale, così che sarebbero potuti apparire ermetici e sibillini a chi non avesse una profonda conoscenza dell'inglese. Ho cercato comunque di conservarne l'aspetto scherzoso ed ironico.

Flavio Brunelli

PREFAZIONE

Questo libro tratta dei giochi d'Adventure e di come crearsene di propri. Sebbene sia principalmente un libro di "come si fa", ho tentato pure di tener desta la passione di divertirsi con questo tipo di gioco, ed ho sparso qua e là molti suggerimenti e consigli che potrebbero aiutare gli appassionati d'Adventure (siano essi principianti od esperti) ad indulgere gioiosamente alla loro passione.
Quando ho cominciato ero un programmatore abbastanza inesperto (suppongo di esserlo in parte ancora) ed il pensiero di scrivere un gioco che dovesse contenere profondità e mistero mi sembrava terribile. Sono lieto di affermare che, a dispetto dell'apparente complessità e lunghezza del programma finale, ciò è stato veramente facile – una volta svelati i meccanismi di base della costruzione della struttura per il mio programma. I miei ringraziamenti vanno a David Lawrence che mi ha orientato nella giusta direzione in occasioni in cui ogni cosa sembrava richiedere troppo sforzo. Così, anche se siete un principiante nella programmazione, questo libro è ben adatto alle vostre capacità.
Mentre il libro vi permetterà ovviamente di inserire e quindi utilizzare il mio gioco – *Il Pianeta dell'Incubo* – vi fornirà anche (e di più verso la fine) un piano per creare la vostra personale Adventure proprio da zero. Non è per niente un libro del tipo 'copiami e gioca' – e sospetto che per molti appassionati d'Adventure l'eccitazione di creare il loro proprio capolavoro (per sconcertare e divertire gli amici) sia la massima soddisfazione del programma e giocare.
Nel corso della scrittura ho imparato a conoscere il Commodore 64 – una macchina superba per i patiti di Adventure di tipo "Text") aggiungeranno quel tocco di professionalità al vostro gioco che rende un programma di bell'aspetto.
Il libro è diviso in due sezioni. La prima riguarda i meccanismi per la creazione di una struttura per la vostra Adventure – come costruire le locazioni e muoversi tra esse, come raggiungere gli oggetti e spostarli nella vostra struttura, e come creare rischi e pericoli da far incontrare ai vostri giocatori. Questa sezione contiene un programma prova chiamato GIOCO DI AVVENTURA il quale è puramente una struttura semplificata per illustrare i concetti (che possono essere usati ripetutamente).
La seconda sezione tratta della creazione dei disegni, della stesura della trama, e della programmazione sul vostro 64 di un'Adventure – dalla concezione al completamento.
Il programma IL PIANETA DELL'INCUBO (che sembrerà all'inizio spaventosamente lungo) è esposto attraverso i vari capitoli in forma modulare per aiutarvi quando inizierete a creare il vostro. Mentre è possibile sedersi e inserire l'intero lavoro in una sessione maratona (!!), se lavorerete capitolo per capitolo introducendolo in uno stadio per volta, ciò vi aiuterà più tardi nel progetto del vostro programma.

Mentre questo libro può essere letto sul treno o davanti al focolare, esso è in realtà un libro da usarsi vicino al 64 o con una matita e parecchi fogli di carta per lavorarci.
Alcune pagine contengono liste o diagrammi che ho trovato inestimabili nella mia progettazione – e se potete usare una fotocopiatrice vi suggerirei di farvene delle copie come aiuto. Ad ogni buon conto divertitevi!
Quando ho iniziato (come ho detto prima) ero proprio un vero giocatore entusiasta d'Adventure. Durante il tempo che ho consumato sono diventato un migliore programmatore e senza dubbio anche un buon giocatore. Infatti il mio entusiasmo sembrava crescere ad ogni capitolo che scrivevo.
I miei ringraziamenti vanno alla Commodore per avermi reso disponibile il 64, a David Lawrence per la sua guida ed il suo aiuto, ed ai miei famigliari per la loro pazienza e gli utilissimi suggerimenti durante la prova della mia Adventure.

PARTE I

CAPITOLO 1

IL MISTERO E LA MAGIA

Il petto ansimante, il sudore gocciolante sul suo corpo color del bronzo, i muscoli doloranti come se fosse stato stirato per una settimana su una ruota di tortura, Caronte barcollava sull'orlo di un vasto precipizio che era apparso sotto i suoi piedi, come da un perverso disegno piuttosto che da uno scherzo della natura. I suoi occhi tremavano guardandosi dietro, nell'udire i cani. Lo avrebbero sicuramente raggiunto da un momento all'altro!

Ma quale via riprendere? Dove fuggire? Dietro di lui la potente armata dell'imperatore Draloor ed il tre volte maledetto Zamrian che caccia con i cani (capace, correva voce, di inseguire un uomo da solo attraverso il vasto oceano) si stavano avvicinando. A destra e sinistra le pareti delle montagne, a cento passi se non di più; ma per i nascondigli che offrivano avrebbero potuto benissimo essere anche a pochi passi! E davanti un precipizio spalancato – una spaccatura naturale nel fondo della valle troppo larga da saltare, anche per i suoi potenti muscoli.

Intrappolato. Troppo tardi ora per pentirsi della sua avventatezza nel corteggiare Darleen (e come poteva sapere che la giovane era la figlia dell'imperatore se ella mai glielo disse?). Sollevò la sua formidabile spada nella mano destra, e si girò ad affrontare i suoi persecutori.

Ai suoi piedi un piccolo anello d'oro brillo' nella luce della sera, richiamando la sua attenzione. Si fermò per raccoglierlo ed ecco all'improvviso che i cani erano su di lui...

Il precedente brano potrebbe benissimo provenire da un libro di "heroic fantasy" (fiaba epica) – un altro termine per "sword and sorcery" (spada e stregoneria), un modello per la scrittura di "fantasy" reso popolare durante gli anni 30 sui giornali dell'epoca, o parte della sceneggiatura per un film. Potrebbe essere l'epilogo di un libro "comic", o anche l'inizio di un gioco chiamato *Draghi e Sotterranei.*

Oppure, ovviamente, potrebbe essere una scena da un gioco di Adventure.

Vi sono delle possibilità che voi, leggendo questo libro, conosciate già, i giochi d'Adventure e abbiate giocato un po' – ma ciò è difficile se il 64 è il vostro primo computer, poiché le possibilità di scelta di questo tipo di software sono alquanto scarse.

Comunque avreste potuto sentire o leggere qualcosa a proposito – ma non vi sarete davvero adoperati per trovare il tempo di inserire i programmi riportandoli da una rivista (un'impresa da maratona a dir poco) o meglio ancora per riuscire a giocare una partita fino alla spesso amara conclusione. Tuttavia il concetto di giochi d'Adventure ha un fascino piuttosto universale e le Adventure stanno moltiplicandosi dappertutto.

Sotterranei e draghi

Se siete ai primi approcci al concetto di 'giocare un ruolo', dedicherò qualche istante a precisare alcuni concetti. I giochi d'Adventure sono così chiamati poiché coinvolgono VOI – il giocatore – in un'avventura con un esito od un altro.

Invece di stare ad osservare come spettatore (come per un film od un libro) ora diventate

l'eroe o l'eroina, e in una certa misura controllate il vostro proprio destino durante il gioco. In nessun istante potete essere sicuri di giungere ad un lieto fine, anzi piuttosto il contrario!
L'idea era stata in origine generata per un altro tipo di gioco con un ruolo attivo sviluppato nel lontano 1970 chiamato *Sotterranei e Draghi*. Diversamente dalla maggior parte di giochi da tavolo o di carte, i giocatori usano la loro mente e l'immaginazione per muoversi attraverso un mondo fantastico creato da un Maestro dei Sotterranei (qualcuno che dirige il gioco, piuttosto che del genere Banchiere del Monopoli), e assume anche il ruolo di vari personaggi da introdurre via via per dare maggiore autenticità al gioco. Nella fase iniziale questa partecipazione con la mente e l'immaginazione è tutto quello che è richiesto, ma ulteriori versioni hanno tentato di assecondare un mercato potenziale più vasto, introducendo una scena, per dare ai giocatori un centro tangibile per la loro attenzione.
Questo ha l'effetto (secondo me) di soffocare le potenzialità creative dei partecipanti, se non si può usare il microcomputer.
Forse è questa una delle ragioni per cui le Adventure stanno guadagnando ora in popolarità.
Un esempio di gioco può assumere il seguente aspetto:
I giocatori scelgono vari ruoli: un elfo, un mago, un guerriero, una principessa, un buffone, ecc. (La maggior parte dei giochi originali si concentrano sul mondo fantastico dei racconti di fate e di folclore – risentendo di una forte inclinazione verso *Il Signore degli Anelli* di Tolkien).
Il Maestro dei Sotterranei crea lo schema principale del gioco (di solito poco prima che arrivino i giocatori). Avrà anche tracciato una intricata mappa di stanze (sono i sotterranei) dentro cui i giocatori si muoveranno. Avrà inoltre creato parecchi tesori che essi dovranno raccogliere, e dei pericoli che farà scattare su di essi finché non saranno giunti in una particolare stanza della mappa. (vedi **Figura 1.1**).
Il gioco inizia. Ciascun giocatore muove a turno – di solito verso est, ovest, nord, sud, su o giù. Non appena il giocatore arriva in una stanza il Maestro dei Sotterranei segnerà il tipo di stanza, gli oggetti che lì occorre trovare, e, se attivabile, il pericolo.
Il giocatore deve quindi usare la presenza di spirito e l'immaginazione (così come l'abilità nel giocare) o nel combattere e sconfiggere il pericolo o nel fuggire, preferibilmente con l'oggetto.
Il fascino del gioco dipende dalla combinazione di mistero e di magia del mondo fantastico, ed in più dalla smisurata ampiezza dell'immaginazione umana.
Nessun pericolo di giocare la stessa partita due volte, data la grande varietà dei possibili schemi. Si dice che alcuni giochi continuino per settimane, non appena giocatori esperti si inoltrano nel profondo dedalo di gallerie, e le trappole de Maestro sono state intuite.

Gli inizi delle Adventure

Già negli ultimi scorsi anni l'idea di introdurre il gioco nei computer ha messo radici nelle menti dei programmatori di grossi computer. Esiste una leggenda sugli operatori intrappolati davanti al loro Mainframe – gli occhi arrossati e annebbiati dal sonno perduto vagando dentro lo smisurato complesso di passaggi e posizioni.
Qualsiasi sia la verità, il gioco non solo è continuato sotto il dominio del micro – in piena fioritura – ma sta guadagnando in popolarità ad ogni anno che passa.

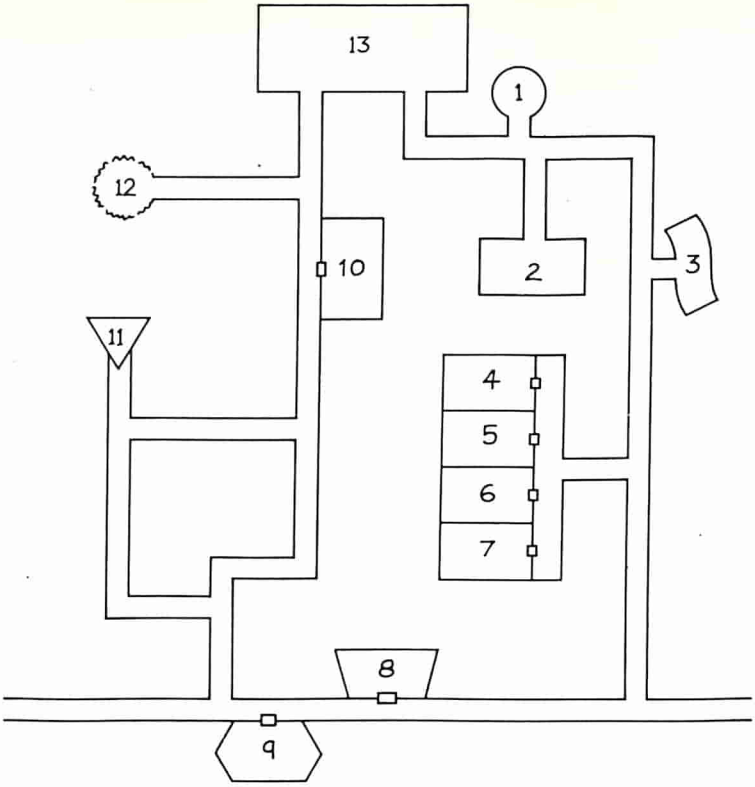


Figura 1.1 Una mappa dal gioco "Dungeon and Dragons"

1. Ingresso in un pozzo del giardino.
2. Stanza con mappa, spada e coltello.
3. Curvatura dello spazio per accedere al livello 6.
4. Cella con un ragno gigante che sorveglia il tesoro.
5. Cella con sei folletti maligni.
6. Cella vuota.
7. Cella vuota – la porta si chiude di scatto dietro di voi.
8. Stanza del Mago. Dovete essere 'magici' per fuggire (livello 4).
9. Passaggio che conduce giù al livello 2. Scala scivolosa.
10. Stanza con un cappello del Mago. Esplorabile solo con i poteri magici più elevati (livello 8).
11. Sotterraneo. Nessuna possibilità di fuga.
12. Passaggio che conduce a morte.
13. Vasta sala del trono con statua gigante dagli occhi formati da rubini. Asportando i rubini la statua prende vita.

La differenza principale tra l'idea originale di *Draghi e Sotterranei* ed il *Gioco d'Adventure* è che quest'ultimo diventa più personale e (nella nostra società cosciente del valore del tempo) più accettabile. È possibile salvare la maggior parte dei programmi su nastro, dimenticarseli e ritornare ad essi più tardi.
Non avrete bisogno di radunare insieme delle persone per giocare. In qualche modo il computer è diventato un valido avversario da battere – con lacci o lacciuoli!
Le prime variazioni si concentrarono sulle Adventure solo di testi, tanto da farle sembrare forse un noioso gingillo, per il fan degli Arcade (orientati alla grafica) colpito soprattutto dagli effetti ottici. Ed è tuttavia questa assoluta mancanza di immagini che dà fascino a questo genere. Attraverso le parole vi si può aiutare ad immaginarvi nella mente situazioni più vivide di quanto possano mai fare dei grafici. Finché i programmatori non potranno produrre immagini altrettanto piacevoli dei cartoni di Walt Disney, (e ciò non è proprio così remoto), continuo a preferire le Adventure basate sui testi alle loro rivali grafiche.
Ma non voglio togliervi le vostre opinioni sulle Adventure basate sui testi, poiché ne esistono alcune che prendono una forma completamente differente da queste ultime, finché non si confondono con una versione ibrida nella quale il personaggio viene manovrato col joystick su un certo numero di locazioni "grafiche". La mia principale obiezione è sull'uso della preziosa memoria (che può essere altrimenti utilizzata per creare un'adventure più lunga o un numero più elevato di rischi e oggetti lungo il percorso). I grafici tendono ancora ad essere un sostegno povero alla mia visualizzazione del mondo fantastico, che la maggior parte delle Adventure permettono di immaginare. Ma non ho ancora giocato *Gli Hobbit!*.
A causa di questa mia preferenza mi sono concentrato, in questo libro, sul genere basato su testi. Ho detto che ci sono parecchie buone versioni grafiche divertenti da giocare – ma io le metterei proprio in una categoria differente.
Non posso in verità finire questa brevissima introduzione senza ricordare Scott Adams, probabilmente il nome più noto nel mondo delle Adventure per Microcomputer, il quale ha in origine scritto questo gioco per il TRS-80, adattandolo poi a parecchie altre macchine famose (compreso il VIC 20 cui probabilmente seguirà il 64). Questi giochi hanno stabilito gli standard che parecchi altri hanno seguito – ed io ne ho sfrontatamente copiato lo stile nella mia Adventure.

Cenni sul come giocare nelle adventure

Sebbene la maggior parte degli appassionati d'Adventure imparino con l'esperienza i trucchi del mestiere lungo il duro cammino, ci sono pochissime vie standard per aiutare il principiante (e, mi si conceda, anche il giocatore esperto qualche volta) a migliorare il proprio gioco. Come per ogni hobby, occupazione o sport, un po' di pratica ed un pizzico di esperienza tendono ad accrescere il divertimento piuttosto che guastarlo.
Tuttavia, se volete condurre a buon fine da soli l'avventura e fare da voi le vostre scoperte, saltate del tutto questa sezione.

Le mappe

Spenderò un po' di tempo sulle mappe nei Capitolo 2 e 9, ma voglio aggiungere qui che le mappe non sono indispensabili, ma vi aiutano per capire in maniera definitiva in quale parte dell'adventure siete. Le disegno sempre su un grande foglio di carta, e tento

di muovere da una locazione all'altra all'inizio, proprio per prefigurarmi dove sia sistemata ogni cosa.
È cosa preziosa annotare la posizione degli oggetti, in quale stanza essi sono, ogni condizione speciale (come una luce che arriva dal fondo di una galleria non ancora esplorata) ed ogni altra traccia. Ogni volta che riterrete che una *qualsiasi* cosa sullo schermo ha una qualche rilevanza, aggiungetela alla mappa. Provate ad usare un foglio di carta 10 volte più grande di quante supponevate necessario. Troverete che è troppo piccolo!
Un gioco con un labirinto veramente complicato, con corridoi, curve, livelli, su e giù ecc. può confodervi. Un modo per tentare di tracciare la vostra avanzata è di trasportare vari oggetti nel labirinto lasciandoli cadere in posizioni strategiche. Quando tornate nella stessa locazione l'oggetto dovrebbe essere lì, e potete quindi chiarire esattamente dove siete.

Gli oggetti

C'è una regola d'oro nelle Adventure – rilevare ogni cosa che potete. È una buona idea, dal momento che parecchi giochi tendono ad avere un oggetto in un posto per una precisa ragione (oscura tuttavia), e potete stare certi che ne avrete bisogno prima o poi. Il problema reale si presenta come un limite posto al vostro inventario, ed è quindi necessaria abilità nel scegliere gli oggetti più opportuni, che andranno sistemati nelle apposite locazioni.
Un problema in più è rappresentato dalla subdola norma di aver bisogno di due o tre oggetti alla volta, per esempio un fucile o un coltello. Un'altra piccola informazione: in alcuni giochi potrete dover ripetere un'azione. Per esempio in "*Terra d'Adventure*" (il primo dei giochi di Scott Adams) trovate una lampada; in seguito al comando SFREGA LA LAMPADA appare un genio che fornisce un utile insieme di informazioni. Se sfregate la lampada una seconda volta viene mostrata un'altra e più utile risposta. Ma guardatevi bene dal farlo una terza volta...
Uno dei trucchi preferiti nei giochi è quello di creare una postazione in cui c'è troppo buio per vederci. Troverete di solito una lampada, una torcia, dei fiammiferi o alcuni altri oggetti con cui illuminare la scena – spesso in qualche altra parte dell'adventure. Ma prima ci potrebbe essere un intralcio, come un limite di tempo posto alla luce, una corrente che spegne il fiammifero, batterie che si esauriscono. La maggior parte dei giochi comprendono un inventario – ed un semplice consiglio consiste nel controllare il vostro inventario, quando iniziate il gioco, di modo che potete anche scoprire che avete già trovato alcuni oggetti.

Guardandosi intorno

Alcune Adventure vi permettono di usare il comando GUARDA, e ciò è spesso necessario per scoprire oggetti o tracce. Provate sempre ad utilizzare il comando GUARDA, se il programma ve lo permette, non appena arrivate in una nuova locazione. Alcune volte il comando mostrerà esattamente la vostra posizione (come nel mio gioco "*Il pianeta dell'Incubo*"). Con l'esperienza la risposta del computer vi dà spesso l'indizio che vi aspettavate. Svariati giochi contengono risposte singole, come una replica standard, per esempio "NON SO COME" seguita da un verbo che non è nel vocabolario del gioco. Potete usare questa particolarità a vostro vantaggio, durante il gioco. Per esempio

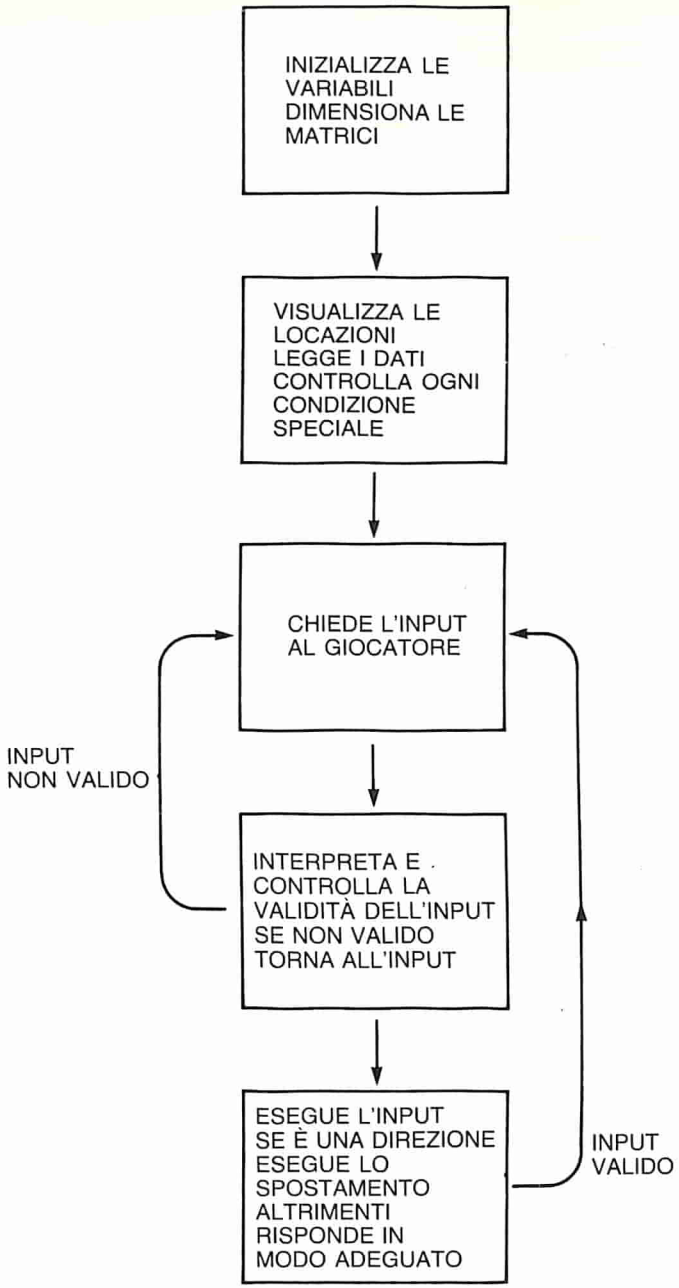
se arrivate in una stanza e inserite APRI LA PORTA ricevendo la risposta NON SO COME APRIRE, allora è ovvio che la parola chiave è qualcos'altro, forse SBLOCCA o proprio SUPERA LA PORTA.
Tuttavia se la risposta è NON PUOI FARE CIO' ANCORA allora potete essere praticamente sicuri che APRI è una parola riconosciuta, ma avete bisogno di qualcos'altro prima che vi sia concesso di aprire la porta, forse una chiave.

Il progetto della stesura della vostra adventure

Suppongo che potremo in realtà partire chiedendoci perché la gente vuole giocare un'ADVENTURE, prima di pensare di scriverne una. Ho già suggerito alcune delle ragioni – l'alone di magico delle nostre fiabe d'infanzia, quando le regine fate potevano esaudire tre desideri e ogni cosa si aggiustava, o i terribili giganti seminavano terrore nei nostri vividi incubi, mentre eravamo coricati al buio.
C'è qualcosa di ancestrale nel mondo della fantasia e dell'orrore che esercita un fascino universale per la gente di ogni paese – soprattutto, chi non drizza le orecchie quando si sta raccontando una buona storia di fantasmi (specie se uno la crede vera)!
Così i giochi di adventure (che hanno le radici nel mondo dell'oscuro e del misterioso) posseggono nella loro struttura questa attrazione e noi crediamo di essere parte di essi. E qui sta la seconda (e più importante suppongo) attrattiva – il fatto di essere investiti di un ruolo nella storia che si svolge. Il concetto di giocare un ruolo è ancora allo stadio infantile (parecchi libri, tratti di peso dal tema di *Sotterranei e Draghi*, sono disponibili già, comunque). Ma siamo ancora parecchio lontani dall'idea, propria della fantascienza, di inserire nella nostra mente un "registratore di sogni", in modo da poter immaginare un'adventure di nostra scelta, nella quale diventare l'eroe o il malvagio. Stiamo comunque andando in questa direzione.
Ancora non siamo completamente coinvolti, ma il microcomputer ci ha definitivamente aiutati ad incamminarci nella giusta direzione. Con una programmazione accurata il giocatore sembra avere una varietà di decisioni da prendere, le quali indirizzeranno lo svolgersi della storia – e nel momento in cui questa procede, la sensazione di partecipare ad una situazione aumenta.
A tutti noi piace dirigere, se possibile, ed è questo che costituisce particolare attrattiva nello scrivere il vostro personale gioco – VOI potete dirigere la storia – VOI potete decidere l'ambiente – VOI create i personaggi – VOI potete proporre indovinelli ed enigmi per far arrabbiare il vostro sfortunato giocatore – VOI possedete il controllo. E senza alcun dubbio, la sensazione di aver successo nel produrre un'opera d'arte (devo infilare la testa nel proverbiale nodo, per aver definito opera d'arte un gioco al computer – ma è pur sempre un atto creativo che produce piacere, così sono giustificato!) è buona ricompensa per lo sforzo.

Il progetto dell'inizio del gioco

Mi si conceda di terminare questo capitolo con una nota più pratica. Abbiamo deciso di iniziare a scrivere la vostra Adventure, ma da dove cominciare?
Se volete iniziare progettando la vostra storia in modo diretto vi suggerisco di passare al Capitolo 5, poiché la prossima sezione del libro tratta concetti di programmazione. Ma prima di farlo non è cattiva idea avere in mente un piano chiaro.



Flowchart 1.1 Un'Adventure di base

Diagramma 1.1. È un diagramma immaginario che descrive le basi per la creazione di un'Adventure.
Il blocco 1 comincia inizializzando alcune variabili (come la posizione del giocatore) e dimensionando le matrici. (Se siete un po' disorientati dal concetto di matrice non è il caso di preoccuparsi a questo stadio. Qualsiasi spiegazione vi dessi ora, per esempio immaginarsele come le pagine di un libro o un orario ferroviario, vi confonderebbe solamente – e, ad esser sinceri non è necessario capire ciò per scrivere un'Adventure!).
Il blocco 2 non è altro che un metodo per visualizzare sullo schermo le informazioni necessarie. Di solito consistono in una breve descrizione di dove vi trovate e di ogni oggetto o pericolo che vi può essere nei dintorni. Anche questo stadio del programma controllerà se siete in alcune locazioni, a cui si applicano condizioni speciali (p.e. potete trovare, entrando in una particolare stanza, che la porta è stata richiusa a chiave dietro di voi, e non potete più fuggire per quella via).
Il blocco 3 chiede un input al giocatore (di solito facendo apparire il messaggio COSA VORRESTI FARE ORA?) e attende quindi la risposta.
Il blocco 4 attua un semplice controllo di validità dell'input. Se non è valido, il programma risponderà in maniera standard (tipo NON SO COME FARE CIO'!) e quindi ripete la richiesta di input nel blocco 3.
Il blocco 5 è il vero nocciolo del programma. Qui tutte le differenti risposte (come AIUTO o VAI A NORD) sono interpretate e quindi attuate. Il programma quindi ritorna al blocco 3 e attende il successivo input.
Naturalmente ci vuole un po' di più di questo per scrivere un gioco completo — ma l'essenziale c'è. Ora affronteremo il vero problema di una seria programmazione.
Sebbene io abbia accennato al capitolo 5 per dare alcune idee sul come creare la vostra storia — se non siete un esperto programmatore il poco tempo in più che dovrete usare per leggere i Capitoli 2-4 sarà prezioso, poiché contengono una chiara spiegazione dei meccanismi di base della strutturazione del vostro programma. Sospetto che le Adventure abbiano bisogno di una pre-pianificazione più di ogni altra cosa.

CAPITOLO 2

MUOVERSI IN UN'ADVENTURE

Dobbiamo capire i meccanismi di base di un gioco di Adventure prima di partire a creare il nostro capolavoro contenente un bel numero di locazioni, pieno di pericoli, oggetti sparpagliati attorno in abbondanza — e tutte le altre ghiottonerie che creano il divertimento (e la frustazione) nella programmazione e nel gioco. Prima di partire ed accendere il 64, è importante avere un'idea di cosa si tratterà nel gioco. Ciò va bene sia per il semplice gioco che ho creato per illustrare varie procedure, sia per il chilometrico gioco della seconda parte del libro.
Probabilmente la prima più importante caratteristica di un gioco interessante è che voi abbiate una grande varietà di locazioni — e che vi possiate muovere in esse in maniera varia. Per fare ciò avete bisogno di una mappa.

Mappe – come tracciarle ed interpretarle

Mentre è possibile giocare tutta un'Adventure senza una mappa, quasi tutti i giochi saranno più gratificanti e facili se potete tracciare la vostra avanzata mentre marciate. Infatti una dei modi più facili per iniziare ogni gioco è di muovere da una locazione all'altra annotando dove siete e quali uscite conducano i nuovi posti. Per LOCAZIONE in una Adventure si può intendere qualsiasi posto, per esempio una stanza, la cima di un albero, dentro un ascensore, fuori da una finestra, ecc.
In una delle Avventure di Scott Adams, *The Count* (Il Conteggio), voi vi alzate dal letto: nella vostra stanza c'è una finestra. Se inserite VAI ALLA FINESTRA (ed essa è già aperta), allora vi spostate in un'altra LOCAZIONE, sul davanzale della finestra. Se poi andate GIÙ (usando un metodo che non rivelo), vi muovete in un'altra LOCAZIONE, il cornicione della finestra, che a sua volta conduce in un'altra stanza nel castello. In ogni caso il cornicione della finestra ed il davanzale della finestra contano come LOCAZIONI separate.
Senza dubbio ci sono tante maniere di tracciare una mappa quante persone capaci di farlo — ma il metodo che ho trovato più facile sembra estremamente semplice sicché lo userò per illustrare le mappe attraverso il libro. Ogni volta che sto per cominciare una nuova Adventure schizzo la mia prima LOCAZIONE nel centro della pagina come una scatola contenente la parola adatta. Uso sempre il centro della pagina a questo stadio poiché non so ancora se sto per cominciare un viaggio verso est, nord, sud od ovest.
Dopo aver disegnato la scatola segno le uscite che mi sono concesse (la maggior parte delle Adventure aiutano dicendovi quali uscite sono disponibili – ma altre invece attendono che voi battiate, per esempio:

VAI A NORD

prima di rispondervi

e quindi aspetta ancora che indichiate un'altra direzione).
Personalmente trovo questo ultimo approccio una perdita di tempo, così come non produce buoni risultati continuare a tentare direzioni differenti finché non trovate quella giusta. Quindi io segno delle frecce (una sola direzione in questa fase) che dalla mia LOCAZIONE vanno nella direzione appropriata.

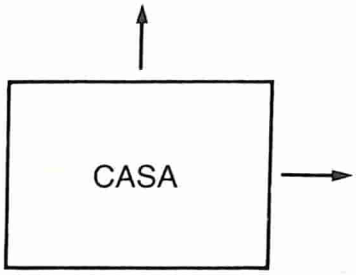


Figura 2.1 Tracciamento della mappa 1

La **Figura 2.1** mostra l'inizio di un'Adventure in cui mi sono trovato in una casa e mi è stato detto che potevo muovere a NORD o a EST.
La prossima fase è di andare in una direzione possibile verso NORD o EST e di annotare la nuova LOCAZIONE. Nell'esempio immaginario avendo inserito NORD o solo N (come in molte Avventure, vi si permette di usare solo la prima lettera di ciascuna direzione), vi venite a trovare su un sentiero ed ora le possibili uscite sono OVEST o SUD (vedi **Figura 2.2**).

A questo stadio posso aggiungere una freccia che torna dal sentiero alla casa (in parecchie Avventure se potete andare in una direzione arrivando in una nuova locazione, potete di solito tornare indietro – ma non sempre), ed una nuova freccia verso OVEST. Questo processo continua finché avrò mappato la maggior parte delle LOCAZIONI logiche, e potrò quindi cominciare a muovermi tra di esse.
Non tutte le Avventure hanno LOCAZIONI ovvie, e nella maggior parte di esse le direzioni SU e GIÙ sono aggiunte alle direzioni solite, rendendo la mappa iniziale piuttosto difficile da disegnare, potendo benissimo avere parecchi livelli. Un esempio di ciò può essere una delle più note Avventure in assoluto – *Terra di Adventure* di Scott Adams – nella quale la mappa iniziale contiene solo una manciata di LOCAZIONI. È solamente dopo la scoperta di un certo "albero" che, una volta scalato, rivela una via GIÙ nel dedalo di passaggi e caverne, che la mappa può essere tracciata in tutta la sua estensione.
Come ho già riferito prima, un intoppo di partenza è che non sapete dove siano le possibili LOCAZIONI – sicché è possibile che vi troviate fuori dalla pagina (vedi **Figura 2.3**). Una piccola ristesura della mappa è tutto quello che è necessario.

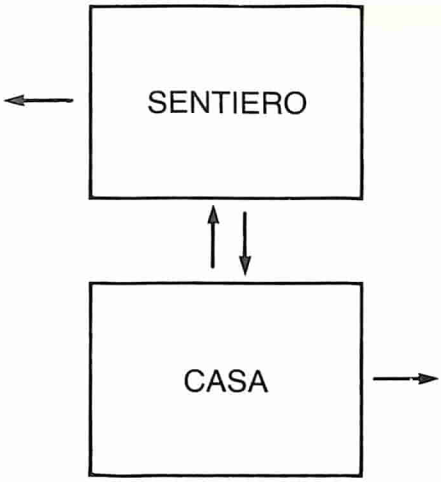


Figura 2.2 Tracciamento della mappa 2

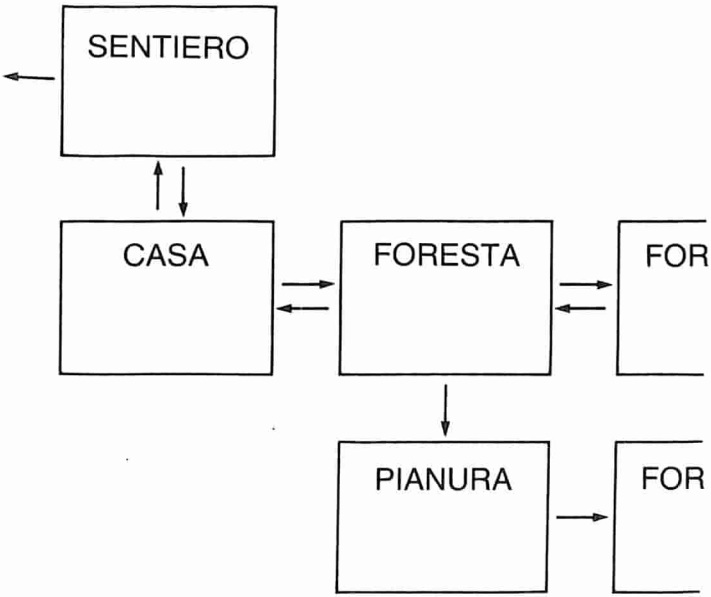


Figura 2.3 Tracciamento della mappa 3

La conversione della mappa in griglia

L'essenza di tutti giochi di Adventure è l'abilità nello spostarsi di posto in posto, e ci sono parecchi modi per far compiere ciò al computer. La più semplice probabilmente è quella di convertire la vostra mappa in una griglia e quindi dare al computer le informazioni per permettergli di accettare certe aree della griglia (le attuali locazioni) come posizioni in cui è permesso muoversi. Se queste aree hanno un numero di riferimento allora è molto facile convincere il computer a "muovere" dentro questa area e visualizzare la locazione.

È più facile capire ciò probabilmente, con un diagramma, (vedi **Figura 2.4**): Nella **Figura 2.4 (a)** la nostra mappa è pronta per la conversione. Ci sono sei locazioni (numerate da 1 a 6 per convenienza, in questa fase), che iniziano con un campo, e quindi

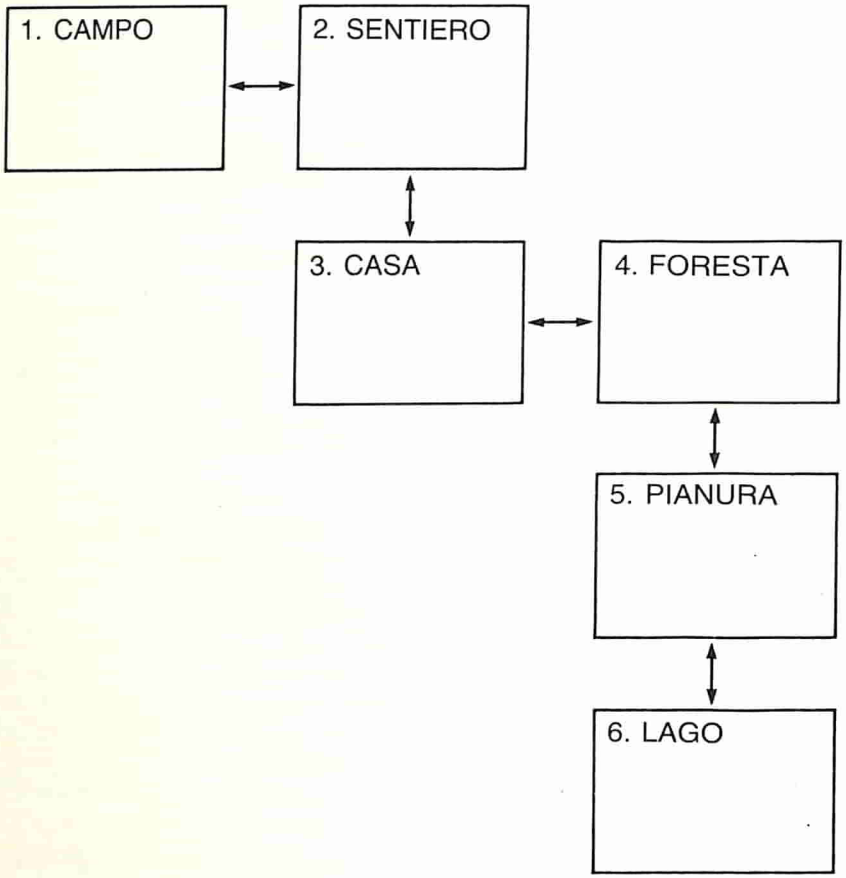


Figura 2.4a Conversione della mappa in una griglia

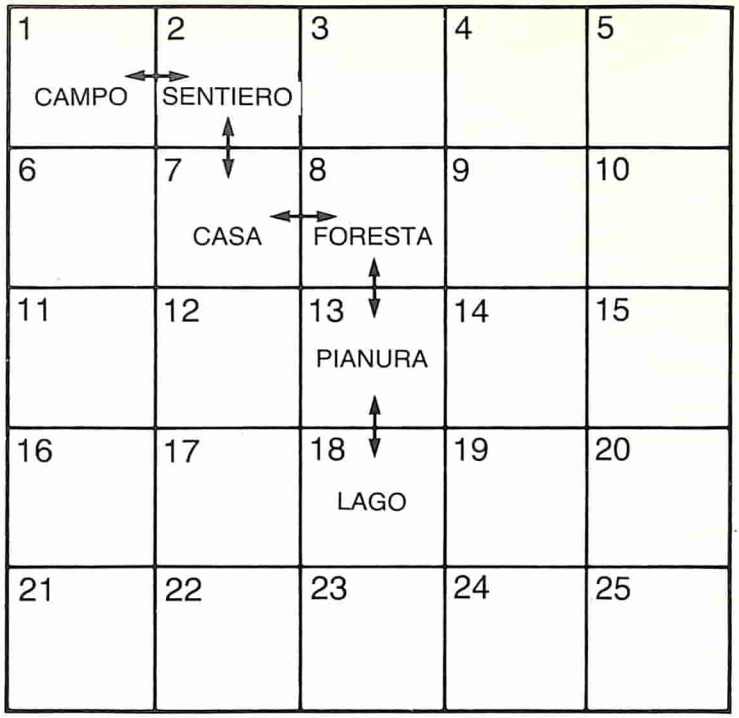


Figura 2.4b

proseguono lungo un sentiero fino ad una casa. Ad est della casa c'è una foresta, quindi ci spostiamo verso sud in una pianura ed infine ad un lago.
La **Figura 2.4 (b)** ha riportato questa mappa in una griglia di 5 x 5 (potrei aver usato una 4 x 4, per lo scopo di questo esempio, ma volendo concedervi delle stanze per i vostri esperimenti, ho deciso di usare la 5 x 5). La conversione è proprio il piazzare i quadrati – lasciandone 19 bianchi per il movimento). Uno dei vantaggi di questo metodo è che diventa facile espandersi o cambiare la vostra mappa mentre proseguite il cammino.

Il movimento all'interno della griglia

Potete aver notato nella **Figura 2.4 (b)** che le varie locazioni ora hanno un diverso numero di riferimento. Per esempio la casa (numero 3 nella **Figura 2.4 (a)** è ora il numero 7). Questo ci consente di fornire al computer una formula semplice per 'convincerlo' di permetterci di 'muovere'.
Se siamo nella locazione 7 e volessimo spostarci verso sud, allora dovremmo spostarci nella locazione 13. In maniera simile se siamo nella Locazione 13 l'ordine sud ci porta

nella Locazione 18. La formula x=x+5 ci porta verso sud – se x=8 allora il sud sarà 8+5 che dà 13. Usando la stessa idea l'andare a nord sarà corrispondente ad una sottrazione – x=x-5 vi sposterà indietro da 18 a 13. L'est sarà ottenuto con x=x+1 mentre l'ovest col contrario, x=x-1.
In una griglia 4 x 4 la formula sarà x=x+4 per il sud e x=x-4 per il nord. Ovviamente l'est e l'ovest sono indifferenti alla dimensione della griglia.
L'ultima considerazione sul movimento è come evitare che il computer muova in una locazione inesistente – per esempio l'est o l'ovest della pianura (LOCAZIONE 13). La soluzione sta nell'ingannare il computer dicendogli che questa locazione non esiste – assegnando a quel quadrato il valore 0.
Ciò diventerà più chiaro dopo che avremo fatto partire l'attuale programma – ma, per adesso, la **Figura 2.5** dovrebbe bastare a dimostrare come noi sappiamo esattamente cosa dire al computer, per realizzare il processo del movimento da una locazione all'altra.

1. CAMPO	2. SENTIERO	0	0	0
0	7. CASA	8. FORESTA	0	0
0	0	13. PIANURA	0	0
0	0	18. LAGO	0	0
0	0	0	0	0

Figura 2.5 La griglia di base per un gioco di Adventure

L'inizio della programmazione della vostra adventure

Ora possiamo finalmente iniziare la programmazione. Prima di descrivere il primo modulo preferirei fornirvi alcuni consigli sulla programmazione in generale, che giudico prezioso menzionare in questa fase. I programmatori esperti possono tralasciarli, non essendovi niente di rivoluzionario qui, ma so dalla mia esperienza che alcuni dei trucchi più semplici del mestiere si dimentica spesso di menzionarli, poiché si presume che siano così semplici che tutti devono conoscerli.

1. *Salvate spesso i programmi*

È un trucco vecchissimo, che ho esitato a menzionare qui, ma che ho poi riportato pensando che quando si è presi dal vivo della programmazione è facile dimenticarselo. Una volta, durante la stesura de "*Il Pianeta dell'Incubo*" (il gioco di Adventure della seconda parte del libro) un temporale provocò il blocco del 64, ed il lavoro di circa un'ora andò perduto. Un'altra volta il disk driver rovinò all'improvviso il mio programma ed anche il mio backup (proprio un bello scherzo questo!), avendo fatto il salvataggio senza il successivo controllo. Ora invece prendo il backup della versione precedente come misura per evitare che ciò possa succedere.
Se il pensiero di tutti questi SAVE e VERIFY vi infastidisce, tenente in considerazione il suggerimento 2.

2. *Un elegante risparmia-tempo*

Se avete un'unità a disco continui SAVE e VERIFY del vostro programma diventano noiosi se non conoscete questo piccolo trucco. Prima di conoscere questo programma non ero consapevole che si potessero sovrapporre i programmi su disco, così come accade automaticamente su nastro, e quindi salvavo la mia nuova versione sotto un diverso nome, e quindi ricambiavo il nome dopo aver cancellato la versione precedente. All'inizio di ogni programma digitate questo piccolo modulo:

1. GOTO 3
2. SAVE "@Ø:NOME",8:VERIFY"NOME",8:STOP
3. REM

Se inserite il nome del vostro programma tra gli apici, ogni qualvolta volete fare il SAVE della versione, dovrete solo inserire GOTO 2 ed il 64 fa il resto. Di tutti i suggerimenti rimanenti che vi ho dato questo è uno dei migliori, ed io sono per questo grato a David Lawrence.

3. *Testate i vostri moduli.*

Quando cominciai la prima volta a copiare programmi da libri e riviste, mi sedevo di fronte al mio computer e mi impegnavo al massimo – tentando di inserire nella macchina tante più istruzioni BASIC che potevo, come se mi fossi prefisso un immaginario fine da raggiungere, senza poter perdere un secondo.
Naturalmente commettevo degli errori ed il rintracciarli diventava una perdita di tempo senza eguali! Comunque, se testate ciascuna sezione del programma più spesso che potete, (avendolo salvato su nastro o su disco) per vedere se funziona, potete spulciare vari errori lungo il cammino, la qual cosa è molto meno frustrante e risparmia tempo.

4. *Usate lo schermo per editare e copiare.*

Quando cominciate a programmare da libri e riviste, è spesso più facile copiare linea per linea, e l'idea di usare lo schermo per risparmiar tempo può spesso sfuggirvi di mente.
Per capire come usare al meglio quest'idea, osservate l'inizio del modulo 1.1, e vedrete che ho usato un sacco di dichiarazioni REM e ":" per spaziare il programma e dargli un aspetto gradevole.
Se digitate la linea 100 prima e la 101 poi, allora è facile cambiare la 101 in 99 e schiacciare RETURN (ricordate di aggiungere o togliere un "*" se il numero di linea cambia da 2 a 3 o da 3 a 4 cifre).
La nuova linea è posta nella memoria senza dover digitare ancora tutte quelle istruzioni con asterischi o ":".
Il sistema può adattarsi benissimo per le linee da 800 a 830 nel modulo 1.3.

Il mio programma esempio contiene solo poche variabili, elencate sotto.

P: Posizione della locazione attuale.
P2: Posizione della nuova locazione.

OB%: Valore intero dell'Oggetto nella Matrice Oggetti.
OB$: Descrizione di una sola parola dell'Oggetto.
SI$: Descrizione estesa dell'Oggetto della Locazione.

I$: Comando aspettato dal Giocatore.
VE$: Verbo atteso in un comando di due parole inserito dal Giocatore.
NO$: Nome atteso nello stesso comando.

SP: Variabile locale usata per dividere I$.
I: Variabile locale per contare gli Oggetti nella Matrice Oggetti.

N: Direzione NORD.
E: Direzione EST.
S: Direzione SUD.
W: Direzione OVEST (West).

SW%: Contiene 2 quando la palude è stata superata.
MW%: Contiene 2 quando il mostro è stato ucciso.

IV: Contatore dell'Inventario.

Descrizioni della locazione

Dopo aver tracciato le locazione sulla nostra mappa, abbiamo bisogno di un messaggio da far apparire sullo schermo quando ci spostiamo in quella posizione, in modo che il giocatore sappia dove è.

Ho usato anche per questo semplice esempio la più concisa spiegazione possibile, proprio per permettere di vedere come funziona il gioco.
Nella seconda sezione del libro questa parte del Programma è forse la più grande divoratrice di byte di tutte, poiché i vari passaggi descrittivi usati per dare atmosfera all'intera storia, occupano veramente una larga fetta della memoria. Tuttavia, dal momento che il fascino del nostro gioco sarà affidato alla nostra abilità nel rendere coinvolto il giocatore – in ciascuna locazione – è ovvio che ciò consumerà quasi tutta la memoria, e secondo me un buon gioco di Adventure ne occuperà gran parte proprio per fornire quella atmosfera, a costo di tralasciare particolari come oggetti aggiuntivi o grafici.

MODULO 1.1

```
1 goto3
2 save"@0:modulo 1.1",8:verify"modulo 1.
1",8:stop
3 rem
99 rem*********************************
**********
100 rem   gioco d' avventura
101 rem*********************************
**********
5991 :
5992 :
5993 rem*********************************
**********
5994 rem   descrizione delle varie posiz
ioni
5995 rem*********************************
**********
5996 :
5997 rem******************
5998 rem   luogo 1
5999 rem******************
6000 print"Siete in un campo."
6010 n=0:e=0:s=0:w=0:goto 500
6017 rem******************
6018 rem   luogo 2
6019 rem******************
6020 print"Siete su un sentiero."
6030 n=0:e=0:s=7:w=1:goto 500
6037 rem******************
6038 rem   luogo 7
```

```
6039 rem*****************
6040 Print"Siete in una casa."
6050 n=2:e=8:s=0:w=0:goto 500
6057 rem*****************
6058 rem    luogo 8
6059 rem*****************
6060 Print"Siete in una foresta."
6070 n=0:e=0:s=13:w=7:goto 500
6077 rem*****************
6078 rem    luogo 13
6079 rem*****************
6080 Print"Siete in una Pianura."
6090 n=8:e=0:s=18:w=0:goto 500
6097 rem*****************
6098 rem    luogo 18
6099 rem*****************
6100 Print"Siete Presso un lago."
6110 n=13:e=0:s=0:w=0:goto 500
```

Linee 99-101: Forniscono il titolo al programma. Ridate un'occhiata al numero 4 della sezione di suggerimenti prima di inserirle, così come tutte le altre linee REM, per risparmiare tempo e fatica.

Linee 5991-5992: I due punti agiscono come un'istruzione REM; è semplicemente un modo per separare aree nella stampa, per eleganza.

Linee 6000-6010: La prima linea stampa la descrizione della locazione.

La linea 6010 è la chiave per correggere il movimento, in modo da chiarire ciò che sta succedendo.

Se ritornate ad osservare la **Figura 2.5** vedrete che la Locazione 1 è un campo. L'unica uscita possibile è verso EST, alla Locazione 2 (il sentiero).

Così la linea 6010 stabilisce che N = 0 (non potete andare a NORD), E = 2 (potete andare ad EST, verso la Locazione 2) e sia S che W = 0 (non potete andare a NORD od OVEST). Infine questa linea vi dice GO TO 500, istruzione nella quale vi verranno mostrate le possibili uscite da questa locazione (vedi modulo 1.3).

Linee 6020-6110: Il resto del Programma completa le diverse dichiarazioni di PRINT per le differenti locazioni, e le appropriate istruzione per permettere il movimento verso Locazioni poste a NORD, EST, SUD, ed OVEST.

MODULO 1.2

```
1 goto3
2 save"@0:modulo 1.2",8:verify"modulo 1.
2",8:stop
3 rem
207 :
208 :
209 rem ***********************************
*********************
210 rem   setta la posizione
211 rem ***********************************
*********************
220 p=7
230 goto 250
235 :
236 :
237 rem ***********************************
*********************
238 rem   setta la posizione
239 rem ***********************************
*********************
240 p=p2
250 if p<11 then on p goto 6000,6020,0,0
,0,0,6040,6060,0,0
260 if p<21 then on p-10 goto 0,0,6080,0
,0,0,0,6100,0,0
270 if p<26 then on p-20 goto 0,0,0,0,0
```

Linee 220-230: Qui si fissa la posizione a cui arrivate dopo aver dato il RUN al programma. P qui è la variabile POSIZIONE, se assegnate ad essa il valore 7 allora il primo messaggio che appare sullo schermo sarà riferito alla Locazione 7 – la casa. Naturalmente potremmo decidere di iniziare l'Adventure in un qualsiasi posto – non necessariamente la casa.

Linea 240: Questa linea è in verità un po' difficile da spiegare in questa fase, sicché inseritela e dimenticatela del tutto per ora. Vi serve solo sapere che essa introduce un'altra variabile – P2 – e che la pone immediatamente uguale a P.

Linee 250-270: Sono le linee che istruiscono il 64 su come mostrare il messaggio appropriato sullo schermo. Usando l'istruzione ON .. GO TO come metodo per selezionare il corretto numero di linea, il programma prende il valore di P (che abbiamo fissato in 7 alla linea 220), e va alla linea 250.

Se P è minore di 11 (come è) allora viene scandita la lista degli indirizzi fino alla settima posizione (corrispondente a 6040).
Scorrete quindi il programma finché non arrivate alla linea 6040; troverete l'istruzione PRINT "SIETE IN UNA CASA".
Quindi, per P = 7, sullo schermo appare il corretto messaggio.
Se P fosse un numero maggiore di 11 (diciamo 13) allora il programma va alla linea 260, e se P è minore di 21, in conseguenza dell'istruzione ON P-10 (cioè 13-10, ovvero 3) GOTO, giunge all'indirizzo corrispondente al terzo numero della lista (che nel caso è 6080).
Se inserite i Moduli 1.1 e 1.2 e quindi date il RUN, potete ricevere il messaggio "SIETE IN UNA CASA", seguito dal messaggio di errore "UNDEF'D STATEMENT ERROR IN 6050".
Questo perché avete detto al Computer di andare (GOTO) alla linea 500 che ancora non esiste.
Fate di nuovo il LIST del programma e cambiate la linea 220 ponendo P = 18, e ridate il RUN. Questa volta riceverete il messaggio "SIETE PRESSO UN LAGO" e di nuovo il messaggio di errore. Prima di inserire il prossimo Modulo rimettete P a 7.
Ora che abbiamo creato le Locazioni e informato il Computer su come trattarle, abbiamo bisogno di aggiungere l'elemento movimento, e di permettere al giocatore di usare l'INPUT per dire al 64 in quale direzione muovere.

MODULO 1.3

```
1 goto3
2 save"@0:modulo 1.3",8:verify"modulo 1.
3",8:stop
3 rem
487 :
488 :
489 rem*********************************
*********************
490 rem mostra le opzioni di direzione
491 rem*********************************
*********************
500 print"Puoi andare a";
510 if n>0 then print" nord";
520 if e>0 then print" est";
530 if s>0 then print" sud";
540 if n>0 then print" ovest";
545 :
546 :
547 rem*********************************
*********************
548 rem   subroutine di istruzioni
```

```
549 rem*********************************
*********************
550 print chr$(13)
560 input"istruzioni:";i$
787 :
788 :
789 rem*********************************
*********************
790 rem  subroutine del movimento
791 rem*********************************
*********************
800 if i$="n" and n>0 then p2=p-5:goto 2
40
810 if i$="e" and e>0 then p2=p+1:goto 2
40
820 if i$="s" and s>0 then p2=p+5:goto 2
40
830 if i$="o" and w>0 then p2=p-1:goto 2
40
837 :
838 :
839 rem*********************************
*********************
840 rem  se non ci sono posizioni possib
ili nella direzione
841 rem*********************************
*********************
850 print:print"spiacente - non puoi seg
uire questa via !!":goto 250
```

Linee 500-540: Mostrano le possibili uscite. Così se N è maggiore di zero, il 64 stamperà NORD, e così via. Si noti sia l'uso dei punti e virgola alla fine delle linee 510-540 – questo impedisce che le direzioni siano stampate su linee separate – sia lo spazio tra virgolette all'inizio di ogni dichiarazione, il che permette di separare con uno spazio ciascuna parola.

Linee 550-560: La linea 560 è un vero INPUT. Se comunque volete inserire prima la linea 550 che stampa il Ritorno Carrello, allora la parola "ISTRUZIONI:" apparirà alla fine della stessa riga contenente le uscite (a causa dei punti e virgola nelle linee 510-54 O). Se volete vedere cosa succede inserite la linea 560 e tralasciate la 550 la prima volta che provate.

Linee 800-830: Qui sta il vero nocciolo dell'intero programma – il movimento reale. Usando la formula che ho descritto prima, queste linee controllano I e quindi se il

movimento è permesso (in altre parole se digitate "N" ed N = O in quella locazione, allora il Computer rifiuterà questa linea di INPUT).
Se il movimento è permesso, segue il rapido cambio di valore di P in P2, la nuova locazione.
Andiamo fino in fondo. Se P = 7 (Locazione 7 – la casa) e digitiamo N per NORD, allora nella linea 800 il 64 controlla se N = O. Se osservate il Modulo 1.1 vedrete che la variabile N è stata posta a 2, nella linea 6050 – sicché il movimento è legale. Ora la formula assegna alla variabile P2 il valore di P – 5 (7 – 5) ovvero 2. Infine la linea contiene un'istruzione GOTO alla linea 240.

Ora possiamo capire perché la linea 240 è posta qui; perché prima che possiamo muovere dobbiamo cambiare P2 riponendola in P, in modo che le linee che seguono possano "comprendere" dove ci si sposta. Abbiamo ora la variabile P = 2 – e osservando la linea 250 possiamo vedere che ora c'è un GOTO alla 6020 – dove c'è il messaggio per la Locazione 2 – il sentiero.

Linea 850: L'ultima del Modulo. Supponiamo di essere nella Locazione 7 (la casa) e chiediamo al Computer di andare a OVEST. Questo è un movimento non concesso, essendoci alla linea 830 W = O. Il Computer passa alla linea seguente, la quale produce una risposta standard "NON PUOI ANDARE PER QUELLA VIA" seguita dal ritorno all'input "ISTRUZIONI" che ripete la richiesta.
Se avete inserito tutto il programma in modo corretto potete verificare che vi è concesso spostarvi dappertutto in completa libertà. Non è propriamente una procedura emozionante — ma illustra il principio del Movimento da un posto all'altro del vostro gioco di Adventure.

Il Flow-Chart per questa parte del programma è mostrato nel **Flow-Chart 2.1.**

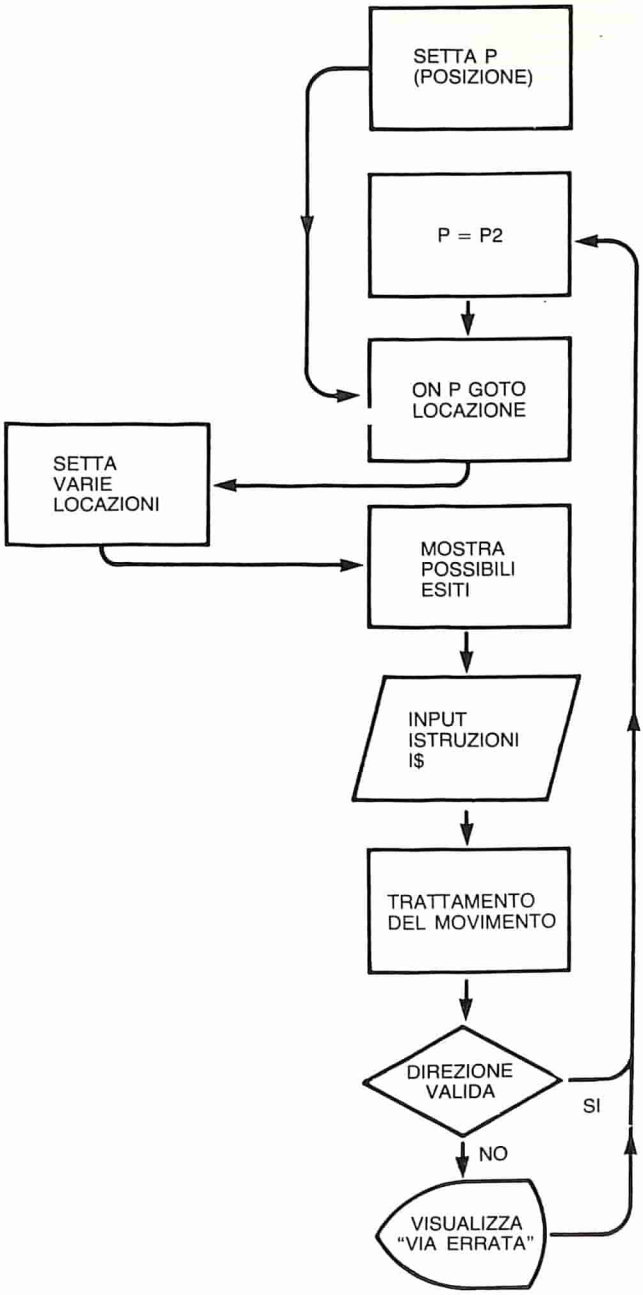
Altri metodi di movimento che possono essere usati, comprendono dichiarazioni DATA e matrici. Sebbene risparmino memoria, io preferisco usare il metodo che ho illustrato, a causa della facilità di modifica, in un programma complesso. Infatti mentre stavo scrivendo "Il PIANETA DELL'INCUBO", l'Adventure della seconda parte del libro, ho modificato parecchie Locazioni ed anche la direzione tra esse, per accrescere la trama. Un secondo vantaggio di questo metodo è la sua abilità nel prevenire facilmente i movimenti illegali – qualche volta mi dilungherò in dettagli più avanti. Per il momento considerate ancora la **Figura 2.5** e immaginate di poter muovere dalla Locazione 2 alla Locazione 1 solamente attraverso una direzione – da EST a OVEST. È una questione semplice modificare le linee 6010 e 6030 nella maniera seguente:

6010 N=O:E=O:S=O:W=O:GOTO 500.

(Si noti che era E = 2 nel programma originale)

6030 N=O:E=O:S=7:W=1:GOTO 500.

Sebbene questa azione possa intrappolare il giocatore nella Locazione 1 (ogni direzione è ora posta a zero) la questione è che in una situazione in cui ci sia un'altra uscita sarebbe possibile passare dal sentiero al campo – ma non tornare di nuovo indietro. Questa flessibilità diventa possibile anche usando la dichiarazione DATA – ma è ben più difficile da realizzare.



Flowchart 2.1 Movimento nelle locazioni

La fase successiva della nostra Adventure è cominciare a sistemare alcuni oggetti nelle varie locazioni – e quindi essere capaci di spostarli intorno. Una annotazione finale: non dimenticate di fare il SAVE su nastro o su disco del programma fino a quando non avrete costruito la struttura, nei seguenti due capitoli.

CAPITOLO 3

OGGETTI E COMANDI

Nell'ultimo capitolo abbiamo creato 5 Locazioni per la nostra Adventure ed abbiamo scoperto come spostarci tra di esse. Dovreste aver notato che dopo aver inserito e usato il programma vi siete un po' stufati. Rendiamolo perciò un poco animato ponendo alcuni oggetti nelle locazioni, ed impariamo come raccoglierli, spostarli attorno, e lasciarli ancora. Descriverò anche come raccogliere un Inventario (una registrazione degli oggetti che state portando) ed inizierò ad analizzare i vari comandi che un giocatore può inserire.

Oggetti e matrici

Come ho detto prima, le matrici possono causare disorientamento, e la comprensione di ciò che trattano non è essenziale alla programmazione del nostro gioco di Adventure – è però di aiuto.
Il modo in cui visualizzo le matrici è pensare ad una variabile (diciamo x) che sappiamo cambierà il proprio valore nel corso del programma. Noi vogliamo tuttavia assegnare a X valori in sequenza piuttosto che in maniera casuale, e vogliamo anche sapere in ogni momento cosa rappresenta ciascun valore.
Per far ciò possiamo scrivere X come X(1), ove 1 è il primo valore di X, X(2) ove 2 è il secondo valore, e così via. La cifra tra parentesi è detta indice di X.

La **Figura 3.1** rende chiaro tutto ciò. Possiamo pensare alla matrice come se fosse costituita da un numero di scatole tutte etichettate con X, dove X(1) è la prima scatola, X(2) la seconda e così via. (Un punto su cui riflettere qui è che il 64 chiamerà la prima scatola con O, se non gli si ordina diversamente – p.e. la scatola 1 è X(O), la scatola

X(1)	
X(2)	
X(3)	
X(4)	
X(5)	

Figura 3.1 Una semplice matrice

2 è X(1) ecc. Trovo che questo possa confondere, sicché tendo per semplicità a far iniziare le mie matrici con X(1).
Dopo la creazione delle matrici passiamo allora ad assegnare dei valori alla variabile X; il risultato può essere visto nella **Figura 3.2.**

X(1)	3
X(2)	14
X(3)	22
X(4)	38
X(5)	73

Figura 3.2 Una semplice matrice contenente valori per X

Torniamo al nostro programma del gioco di Adventure. Dobbiamo decidere quali oggetti sistemare nel gioco, ed in quali Locazioni.
La **Figura 3.3** È di nuovo la griglia che abbiamo usato per la nostra mappa. Ho messo un "COLTELLO" nella Locazione 1 (il campo); un "FUCILE" nella Locazione 7 (la casa) ed un "GIOIELLO" nella Locazione 13 (la pianura).

1. CAMPO COLTELLO	2. SENTIERO	0	0	0
0	7. CASA FUCILE	8. FORESTA	0	0
0	0	13. PIANURA GIOIELLO	0	0
0	0	18. LAGO	0	0
0	0	0	0	0

Figura 3.3 La griglia per il gioco di Adventure contenente oggetti

Possiamo fissare le matrici per questi 3 oggetti come in **Figura 3.4**.

1.	OB% (1)	COLTELLO
2.	OB% (2)	FUCILE
3.	OB% (3)	GIOIELLO

Figura 3.4 La matrice degli oggetti nel gioco di Adventure

Notate che ho assegnato agli oggetti la variabile nome OB% (il % significa valore o numero intero). Così quando si parla di OB%(1) vogliamo indicare il COLTELLO, mentre OB%(2) è il FUCILE e OB%(3) è il GIOIELLO. Ora possiamo "parlare" al 64 di questi oggetti, e manipolarli attorno alle nostre locazioni – in modo che quando digitiamo la parola COLTELLO questa viene intesa dal Computer con OB%(1).

MODULO 1.4

```
1 goto3
2 save"@0:modulo 1.4",8:verify"modulo 1.
4",8:stop
3 rem
147 :
148 :
149 rem*********************************
*********************
150 rem  predispone le matrici per gli o
ggetti
151 rem*********************************
*********************
160 dim ob%(5),ob$(5),si$(5)
170 for i=1 to 3:read ob%(i),ob$(i),si$(
i):next
180 data 1,coltello,qui c'e' un coltello
190 data 7,fucile,il vostro fucile e' qu
i
200 data 13,gioiello,per terra c'e' un g
ioiello
287 :
288 :
289 rem*********************************
*********************
```

```
290 rem Per stampare l'oggetto nella giu
sta posizione
291 rem*****************************
*********************
300 for i=1 to 3:if ob%(i)=p then print
si$(i)
310 next
```

Prima di ripartire a programmare notate che i moduli in questo capitolo si presume siano aggiunti ai moduli dell'ultimo, e non funzionano se usati da soli.
Il Modulo 1.4 definisce le matrici per il nostro esempio e poi stampa i messaggi appropriati sullo schermo.

Linea 160: questa è la linea che definisce le matrici in questo programma. L'ordine DIM significa DIMensionamento ed è l'istruzione per dire al computer che state per definire le vostre matrici. In senso stretto non avremmo bisogno di una DIM in questo esempio dal momento che il 64 tratta matrici fino a 11 posizioni (ricordate che ciò significa un DIM (10), dato che si inizia da O) – sicché dovreste aver bisogno di usare tale dichiarazione solo per DIM(100), diciamo, o DIM(30). Potrei però aver bisogno, senza dubbio, di aggiungere degli altri oggetti più tardi, quando l'Adventure si allargherà, sicché è meglio usare la dichiarazione DIM.
OB%(5) ci offre 5 possibili valori per gli oggetti – OB$(5) ci da 5 nomi – e SI$(5) ci da 5 descrizioni. Dovremo solo usarne per il momento tre, e le linee 180-200 sono delle istruzioni DATA che devono essere sistemate in queste linee.

Linea 170: una linea standard che dice al programma "READ i dati" uno alla volta, e lega ciascuna stringa con un valore intero. Sicché quando il 64 incontra OB%(1) allora stamperà o fl—COLTELLO" o "QUI C'È UN COLTELLO".

Linee 170-200: Spesso i dati sono sistemati alla fine o all'inizio del programma, ma per chiarezza li ho messi qui. Si devono notare due aspetti qui:

1. Il valore intero è lo stesso che per la Locazione.
2. L'attuale parola in SI$ è molto importante, e deve essere usata con riguardo.

Posso illustrare questo ultimo punto cambiando il valore di SI$ per OB%(2) – il 'fucile' – da "IL VOSTRO FUCILE È QUI" a "IL VOSTRO FUCILE È SUL TAVOLO". Ora questa seconda formulazione è più carina, la prima volta che si gioca; e quello che appare sullo schermo entrando nella Locazione 7 sarà:

SIETE IN UNA CASA

IL VOSTRO FUCILE È SUL TAVOLO

Supponete ora di prendere il fucile e portarlo nella foresta, cioè lo dovete lasciare là per una qualche ragione. Tornando alla foresta più tardi voi vedreste il messaggio:

SIETE IN UNA FORESTA

IL VOSTRO FUCILE È SUL TAVOLO

Questo è evidentemente ridicolo, è farebbe sembrare dilettantesco il vostro gioco. È per questo che ho lasciato il più vago ma coerente "IL VOSTRO FUCILE È QUI".

Linee 300-310: queste linee mostreranno in effetti la stringa SI$ sotto la descrizione della Locazione, che riporta l'oggetto presente in essa. Quando un oggetto è in una particolare Locazione allora OB%(I) dà il valore di quella Locazione. Se osservate le Linee 180-200 ve ne renderete conto.
Perciò ogni volta che la Locazione cambia, il 64 scandisce il vettore OB% partendo da OB%(1) e arriva fino a OB%(3), e se ognuno di questi numeri è lo stesso del numero della Locazione (che settiamo a P) allora stamperà SI$ per quel numero. Questo dà l'impressione che quell'oggetto sia in quella Locazione.
Il NEXT alla linea 310 manda il programma indietro, ad esaminare OB%(2) e OB%(3), e se essi pure sono nella Locazione, allora anch'essi saranno mostrati sullo schermo.

Nota importante

Quando battete per la prima volta il gioco, nel Capitolo 2, tutte le linee che descrivono le uscite dalle Locazione (Linee 6010, 6030, 6050, 6070, 6090, e 6110) terminano con GOTO 500. Dovete ora cambiarle da GOTO 500 a GOTO 300 perché ora indirizziamo il programma a questa Sezione del Modulo 1.4 per permettere di mostrare sullo schermo le appropriate stringhe che riguardano gli oggetti. Nell'ultimo capitolo non avevamo ancora scritto questa sezione, e per questo avevamo dovuto indirizzare il Programma alla Locazione 500.
Vi raccomando di cambiare il valore dopo il GOTO da 500 a 300 (ora) per permettere al programma di funzionare quando lo provate.
Se avete inserito il vostro programma finora in modo corretto, dovrebbe apparire il **Programma 1.1**, e se date il RUN dovreste essere ora capaci di muovervi tra le Locazione e vedere il coltello, il fucile e il gioiello mostrati nelle posizioni giuste.

Programma 1.1

```
1 goto3
2 save"00:programma 1.1",8:verify"progra
mma 1.1",8:stop
3 rem
99 rem*****************************************
*********************
100 rem   gioco d'avventura
101 rem*****************************************
*********************
207 :
208 :
209 rem*****************************************
*********************
210 rem   setta la posizione
211 rem*****************************************
*********************
```

```
220 p=7
230 goto 250
235 :
236 :
237 rem*************************************
*********************
238 rem   setta la posizione
239 rem*************************************
*********************
240 p=p2
250 if p<11 then on p goto 6000,6020,0,0
,0,0,6040,6060,0,0
260 if p<21 then on p-10 goto 0,0,6080,0
,0,0,0,6100,0,0
270 if p<26 then on p-20 goto 0,0,0,0,0
487 :
488 :
489 rem*************************************
*********************
490 rem  mostra le opzioni sulla direzio
ne
491 rem*************************************
*********************
500 print"puoi andare";
510 if n>0 then print" nord";
520 if e>0 then print" est";
530 if s>0 then print" sud";
540 if w>0 then print" ovest";
545 :
546 :
547 rem*************************************
*********************
548 rem   subroutine di istruzioni
549 rem*************************************
*********************
550 print chr$(13)
560 input"istruzioni:";i$
787 :
788 :
789 rem*************************************
*********************
790 rem  subroutine del movimento
```

```
791 rem*****************************
*******************
800 if i$="n" and n>0 then p2=p-5:goto 2
40
810 if i$="e" and e>0 then p2=p+1:goto 2
40
820 if i$="s" and s>0 then p2=p+5:goto 2
40
830 if i$="o" and w>0 then p2=p-1:goto 2
40
837 :
838 :
839 rem*****************************
*******************
840 rem  se non ci sono posizioni possib
ili nella direzione
841 rem*****************************
*******************
850 print:print"spiacente - non puoi pre
ndere quella via !!":goto 250
5991 :
5992 :
5993 rem*****************************
*******************
5994 rem  descrizione delle varie posizi
oni
5995 rem*****************************
*******************
5996 :
5997 rem*****************
5998 rem   luogo 1
5999 rem*****************
6000 print"Siete in un campo."
6010 n=0:e=2:s=0:w=0:goto 500
6017 rem*****************
6018 rem   luogo 2
6019 rem*****************
6020 print"Siete su un sentiero."
6030 n=0:e=0:s=7:w=1:goto 500
6037 rem*****************
6038 rem   luogo 7
6039 rem*****************
```

```
6040 print"▒siete in una casa."
6050 n=2:e=8:s=0:w=0:goto 500
6057 rem*****************
6058 rem    luogo 8
6059 rem*****************
6060 print"▒siete in una foresta."
6070 n=0:e=0:s=13:w=7:goto 500
6077 rem*****************
6078 rem    luogo 13
6079 rem*****************
6080 print"▒siete in una pianura."
6090 n=8:e=0:s=18:w=0:goto 500
6097 rem*****************
6098 rem    luogo 18
6099 rem*****************
6100 print"▒siete presso un lago."
6110 n=13:e=0:s=0:w=0:goto 500
```

Manipolazione elementare di stringhe

In questa fase abbiamo bisogno di iniziare a capire un po' come il 64 può manipolare le stringhe di testo, perché dovremo trattare con frasi e parole – e più importante di tutto dobbiamo sapere quando il giocatore ci dà una risposta carente o illegale.
La manipolazione può ingenerare abbastanza confusione ai principianti – sicché mi limiterò ai comandi che si riferiscono in maniera specifica al gioco d'Adventure. Se volete saperne di più dovrebbero esservi di aiuto per allargare la vostra conoscenza sia "*Introduzione al BASIC*", della stessa Commodore, o qualcuno dei libri o articoli sullo stesso tema.

MODULO 1.5

```
1 goto3
2 save"@0:modulo 1.5",8:verify"modulo 1.
5",8:stop
3 rem
687 :
688 :
689 rem*************************************
*******************
690 rem testa le direzioni di una sola l
ettera per i$
691 rem*************************************
*******************
700 if i$="n"ori$="e"ori$="s"ori$="o" th
```

```
en 800
715 goto 950
937 :
938 :
939 rem*****************************************
*********************
940 rem testa i comandi di una sola lett
era per i$
941 rem*****************************************
*********************
950 ifi$="i" then 2000
960 ifi$="a" then 2100
987 :
988 :
989 rem*****************************************
*********************
990 rem  subroutine di controllo per due
 parole
991 rem*****************************************
*********************
1000 for i=1 to len(i$)
1010 if mid$(i$,i,1)=" " then 1100
1020 next
1030 print"prego, puoi usare due parole"
:goto 550
```

Linee 700-710: Abbiamo già stabilito che il 64 risponderà ai comandi N, E, S o O (linee 800-830 nel Modulo. 1.3). Queste due linee indirizzeranno i Comandi di Direzione (nord, est, ecc.) alla Routine del Movimento, alle Locazione 800-830 e tutti gli altri comandi nella sezione seguente di questo modulo (Linee 950-960), dove si controllerà se sono comandi validi.

Linee 950-960: queste vi mandano proprio alle sezioni INVENTARIO o AIUTO del programma che si trovano più avanti.

Linee 1000-1030: Abbiamo esaurito i nostri comandi possibili di una sola parola o lettera – e ora possiamo controllare se il giocatore ha inserito 2 parole o no. Questa piccola routine ricerca nell'input uno spazio bianco, e se non ne trova uno stampa il messaggio "USATE DUE PAROLE PREGO".
La Linea 1000 scandisce l'intero input I$ nel modo seguente: LEN(I$) è la lunghezza totale della stringa dato come numero (per. es. VAI in numero vale 3 mentre VAI NORD vale 8, poiché si conta lo spazio come uno). Così se l'input è VAI NORD questa linea diventa FOR I=1 TO 8, ed il 64 partirà da 1 (la prima lettera, che è "V" in questo caso) e avanza lungo la stringa fino a 8 (l'ultima lettera, che è "D").

La Linea 1010 controlla "b" (spazio) quando si scandisce la stringa. MID$ ha la funzione di "guardare in mezzo" alla stringa – partendo dalla posizione I (che parte da 1 e arriva a 8) e poi estrarre un singolo carattere (a causa del numero 1 nel comando MID$ (I$,SP,1)). Se trova uno spazio singolo allora fa avanzare il programma alla Linea 1010. Se ha scandito tutta la stringa senza trovare uno spazio, allora la Linea 1030 stampa l'apposito messaggio "USATE DUE PAROLE, PREGO", ed il programma torna indietro alla Locazione 550 per attendere un altro input.
Ciò può essere illustrato da un semplice diagramma, nel **Flow-Chart 3.1.**

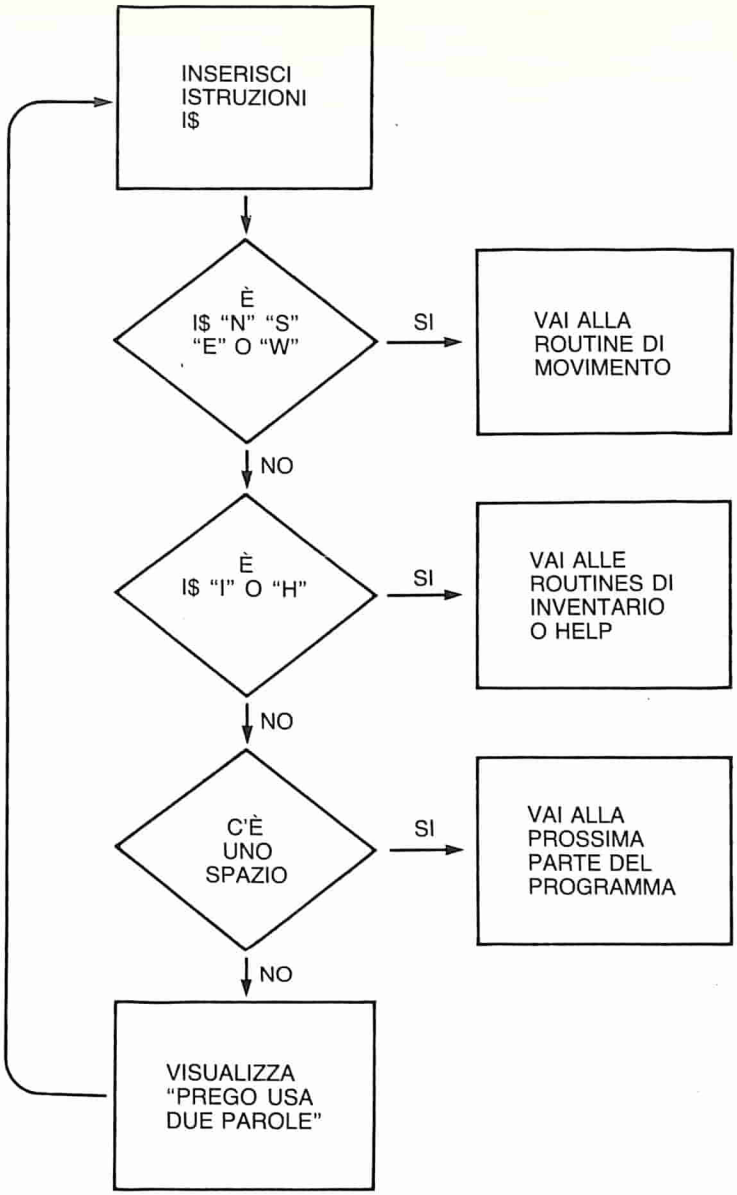
Il prossimo modulo contiene i comandi di manipolazione-stringhe più difficili da controllare – ma come con altre sezioni non è così duro come sembra.

MODULO 1.6

```
1 goto3
2 save"@0:modulo 1.6",8:verify"modulo 1.
6",8:stop
3 rem
1087 :
1088 :
1089 rem*********************************
**********************
1090 rem subroutine per convertire i$ in
 ve$ e no$
1091 rem*********************************
**********************
1100 for sp=1 to len(i$)
1110 if mid$(i$,sp,1)=" " then 1130
1120 next
1130 ve$=left$(i$,sp-1)
1140 no$=right$(i$,(len(i$)-sp))
```

Linee 1100-1120: Queste linee controllano la lunghezza di I$, ancora una volta (come faceva la Linea 1000 nel Modulo 1.5) ogni volta che la variabile SP (per SPAZIO) è usata. Sappiamo che dovrà esserci uno spazio se il loop nelle Linee 1000-1020 ne troverà uno, o in caso contrario si ritornerà ancora all'input ISTRUZIONI. Ora vogliamo trovare dove sia lo spazio, e quindi chiameremo la parte di I$ prima dello spazio VE$ (VERBO), e la parte di I$ dopo lo spazio NO$ (NOME).

Linea 1130: Questa è la linea che individua la parte di I$ prima dello spazio. È abbastanza facile ragionarci quando un esempio. Se il comando I$ è VAI NORD allora lo spazio corrisponderà al numero 4 (essendo al quarto posto). Perciò VE$ è la parte sinistra di I$ che comprende le prime tre lettere (SP - 1 è 4 - 1 = 3), cioè VAI.



Flowchart 3.1 Trattamento di base delle stringhe

Linea 1140: Trattiamo la parte rimanente di I$. Usiamo la funzione RIGHT$ per I$ (la parte destra) e la memorizzazione – ma questa volta usiamo tutta la lunghezza della stringa (LEN (I$)) e sottraiamo lo spazio. Così se usiamo VAI NORD ancora, LEN(I$) sarà 8 (sette lettere più uno spazio) e NO$ sarà la parte destra di I$ partendo dalla quarta lettera (8 - 4 = 4) che è la N di NORD.
Quindi abbiamo diviso VAI NORD in VE$ e NO$, contenenti rispettivamente VAI e NORD. Può sembrare un po' complicato – ma dobbiamo essere in grado di riconoscere sia VE$ che NO$ nel prosieguo del programma.

Prova del Programma
Se date il RUN al programma dovreste trovarlo ora abbastanza più piacevole di quanto era prima, avendo aggiunto gli oggetti nelle Locazioni, all'inizio di questo Capitolo.
Possiamo provare i Moduli di manipolazione delle stringhe battendo qualsiasi istruzione che il programma non sia in grado di riconoscere. Tentate inserendo VOLA o SALTA, dovreste ricevere la risposta "USARE DUE PAROLE PREGO". Per testare il Modulo 1.6 dobbiamo invece battere due parole (provate con MANGIA FORMAGGIO), e poi uscire dal programma battendo RUN/STOP e RESTORE. Battete ora, in modo diretto, PRINT VE$, ed il computer dovrebbe stampare "MANGIA". Ora, ancora in modo diretto, inserite PRINT NO$; questa volta dovrebbe apparire sullo schermo "FORMAGGIO". Se questo non dovesse succedere, esaminate attentamente se avete fatto degli errori, prima di giungere al prossimo Modulo.
Inserite I o A, avrete un messaggio di errore, ma correggeremo fra poco tutto ciò aggiungendo la routine di INVENTARIO e di AIUTO. È più divertente pensare alle routine PRENDI e BUTTA, che ci consentiranno infine di raccogliere quel coltello e quel fucile, e di portarli con noi.

Aggiungere e togliere

MODULO 1.7

```
1 goto3
2 save"@0:modulo 1.7",8:verify"modulo 1.
7",8:stop
3 rem
1187 :
1188 :
1189 rem*********************************
**********************
1190 rem subroutine per scandire vari co
mandi ve$
1191 rem*********************************
**********************
1200 ifve$="prendi"orve$="acchiappa"orve
$="afferra"orve$="porta"then 2200
1210 ifve$="butta"orve$="perdi"orve$="la
```

```
scia"then 2300
1220 ifve$="uccidi"then 2500
1230 Print:Print"non so come fare '";ve$
"' ":goto 550
1987 :
1988 :
1989 rem*********************************
**********************
1990 rem   sezione dell' inventario
1991 rem*********************************
**********************
2000 Print"il tuo inventario e':":iv=0
2010 for i=1 to 3
2020 if ob%(i)=-1 then Print ob$(i):iv=i
v+1
2030 next
2040 if iv=0 then Print"niente"
2050 goto 500
2087 :
2088 :
2089 rem*********************************
**********************
2090 rem   sezione di aiuto
2091 rem*********************************
**********************
2100 if P=18 then Print"Puoi tentare di
ucciderlo":goto 550
2110 Print"non c'e' molto aiuto qui. ho
Paura":goto 550
2187 :
2188 :
2189 rem*********************************
**********************
2190 rem  subroutine di aquisizione
2191 rem*********************************
**********************
2200 for i=1 to 3
2210 if ob$(i)=no$ then 2230
2220 next
2230 ifob%(i)=-1 thenPrint"ce l'hai":got
o 500
2250 ifob%(i)<>P thenPrint"non e' qui":g
```

```
oto 500
2260 print"ok":ob%(i)=-1
2270 goto 500
2287 :
2288 :
2289 rem*********************************
**********************
2290 rem   subroutine del trabocchetto
2291 rem*********************************
**********************
2300 for i=1 to 3
2310 if ob$(i)=no$ then 2330
2320 next
2330 ifob%(i)<>-1 thenprint"non ce l'hai
":goto 500
2340 print"ok":ob%(i)=p
2350 goto 500
```

Linee 1200-1230: Avendo suddiviso I dobbiamo ora esser in grado di riconoscere con una certa facilità i comandi e agire in conformità. Per questo programma-esempio ne ho selezionati in realtà pochi, ma ne "*Il Pianeta Dell'Incubo*" il numero dei verbi concessi diventa così alto che non mi hanno concesso spazio abbastanza, obbligandomi ad una rinumerazione.

La linea 1230 non è altro che la risposta standard, quando il programma esamina un comando che non riconosce, passando poi il controllo all'input Istruzioni – in modo che se battete SCALA la risposta dovrebbe essere "NON SO COME FARE SCALA".

Linea 2000: Se avete inserito I come risposta, allora il programma dovrebbe andare a questa linea e stampare "IL VOSTRO INVENTARIO È:", e la variabile IV viene posta a zero.

Linee 2010-2030: Questo breve loop spazzola le matrici dei dati relativi agli oggetti. Se OB%(I) è vero (in altre parole è uguale a -1) allora il 64 stamperà OB$ (il nome dell'oggetto).
Ogni volta che un nuovo oggetto è aggiunto all'Inventario, IV è incrementata di 1. È di grande significato, in un programma in cui si dovrebbe restringere il numero degli oggetti, che il giocatore possa portarne uno in qualsiasi momento (ho scritto proprio questa routine in "*Il Pianeta Dell'Incubo*").

Linee 2040-2050: La linea 2020 usa una dichiarazione IF per controllare se OB% è vero (= -1). Se non lo è il programma arriva alla linea 2040 e stampa "NIENTE" se IV = O. Poi ritorna alla stampa delle uscite alla linea 500.

Linea 2110: In questa fase non c'è bisogno di aiuto dato che la linea stessa fornisce, al comando "A", una locazione dove andare.

Linee 2200-2260: La routine GET segue la stessa procedura della routine INVENTORY,

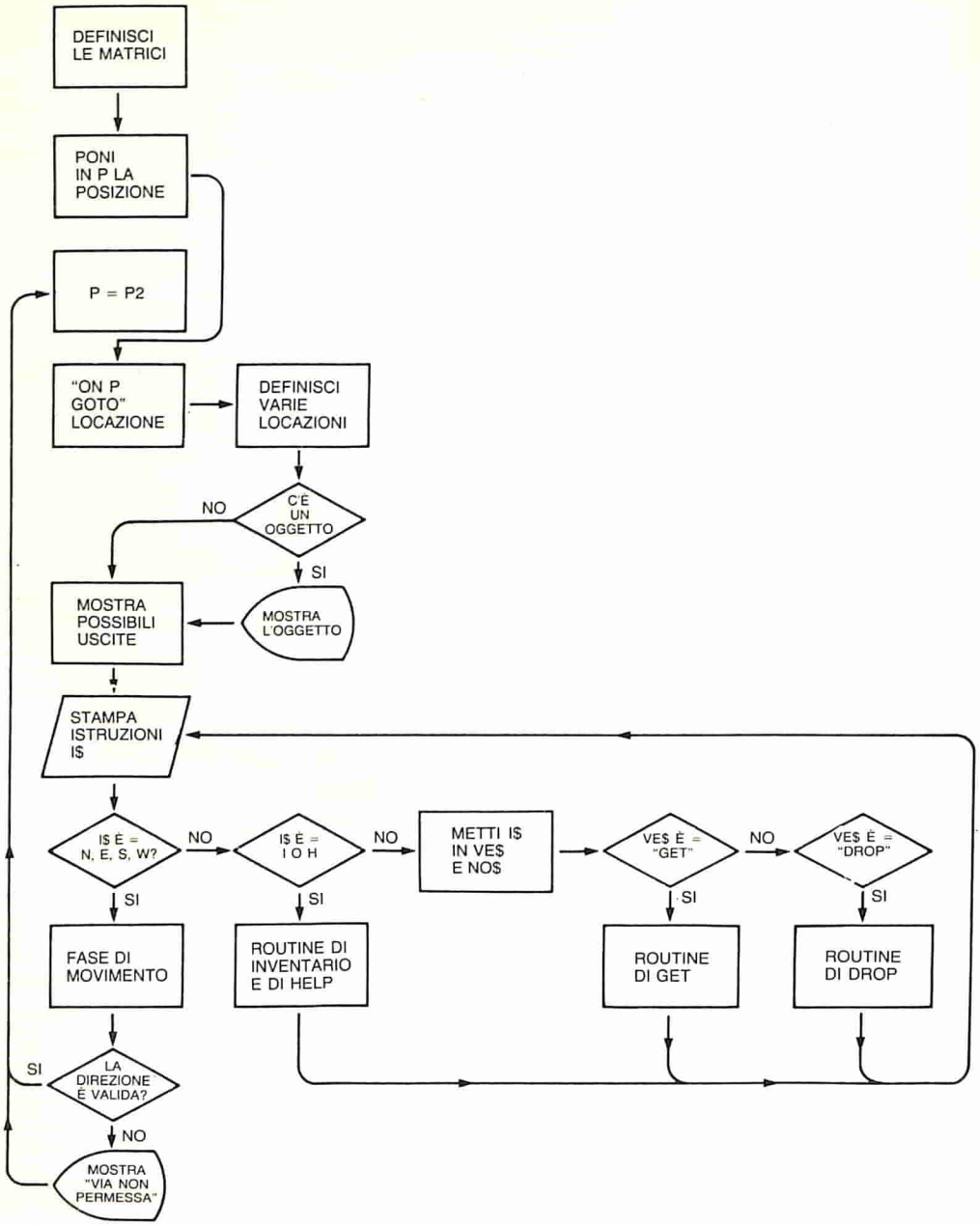
eccetto il fatto che controlla se NO$ è la stessa parola dell'oggetto nella Locazione (OB$) e quindi vi dice che avete già questo, se OB$ è nell'Inventario, o che l'oggetto non è qui, se OB% è diverso da P. Nella Linea 2250 la condizione OB%(I)-1 mette nella condizione di leggere l'oggetto in Inventario, nella Linea 2020.

Linee 2300-2350: Ancora la stessa routine, eccetto il fatto, questa volta, che la Linea 2340 risistema l'oggetto nella nuova Locazione P.
Se ora fate partire il programma (non dimenticate prima di salvarlo su nastro) dovreste trovare che potete prendere gli oggetti, e se poi tornate in quella Locazione, questo non dovrebbe più esserci. Se ora lo lasciate in qualche altro posto allora lasciando la Locazione e ritornando ad essa, l'oggetto dovrebbe essere mostrato. Infine se controllate il vostro inventario dopo che avete prelevato qualcosa, la dovreste trovare.

C'è una sensazione di meta raggiunta, per esser giunti così lontano – perché ora alcune delle possibilità inseribili nel programma sono rese operative. Vi raccomando di passare un po' di tempo trafficando con i listati, aggiungendo altri oggetti in altre Locazione, cambiando le dichiarazioni DATA nelle linee 180-200. Potete anche aggiungere oggetti cambiando l'istruzione FOR I = 1 TO 3, e se voi avete più di cinque oggetti, anche le istruzioni DIM.

Per quelli di voi a cui piacciono i Flow-chart, ne ho incluso uno in questa fase, **Diagramma di Flusso 3.2** (pag. 42). Sembra un po' complicato ora, ma è lo stesso diagramma della **Figura 2.6** con in più il contenuto di questo capitolo.

Nel prossimo capitolo cominceremo ad aggiungere una nuova dimensione introducendo un paio di "pericoli" per il nostro giocatore.



Flowchart 3.2 Il gioco di Adventure con inclusi gli oggetti

CAPITOLO 4

TRABOCCHETTI E PERICOLI

La storia di Romeo e Giulietta è certamente non nuova – il giovane incontra la ragazza, si innamora di lei, i genitori non le concedono di sposarlo, fuggono insieme, e quindi a causa delle incomprensioni scoppia la tragedia. Quello che dà fascino universale a questa storia ed ha portato a così tante variazioni, è la quantità di ostacoli e difficoltà che l'eroe e l'eroina incontrano durante la loro breve relazione.
È lo stesso in ogni aspetto della vita – noi troviamo più gusto nelle cose che cominciano in maniera negativa – specialmente nella nostra immaginazione e fantasia. E i giochi di Adventure non fanno eccezione!

Dobbiamo cominciare a pensare alle varie difficoltà e pericoli che il nostro Intrepido sta per incontrare nel suo viaggio attraverso la nostra Adventure. Dopo tutto il raccogliere oggetti e poi lasciarli cadere, vagando tra le locazioni, non è propriamente un'azione così eccezionale; è invece la risoluzione degli enigmi e la fuga dai pericoli che rende il giocatore Adventure così attraente e frustrante allo stesso tempo.

Incontriamo pericoli di ogni genere e grandezza (e anche varietà di colore). Alcuni sono abbastanza prevedibili – come un mostro che vi si avvicina – una Strega che vi lancia un incantesimo – cadute in buche – e così via. Alcuni sono un po' più subdoli – un indovinello che dovete risolvere per raggiungere il vostro premio – un labirinto in cui siete intrappolati e di cui non potete trovare l'uscita – e innumerevoli altri.
Teniamoci ancorati al prevedibile e vediamo di scoprire come inserirne un po' nel nostro programma esempio.

Pensando alle nostre difficoltà

Come nella maggior parte della programmazione, il più difficile è pensare ai problemi della prima fase – e questo è il motivo per cui mi limito per ora al prevedibile. Quando stavo creando il mio programma esempio decisi di animarlo introducendo una palude (o sabbie mobili) nella Locazione 2, (il sentiero) e un mostro nella locazione 18 (il lago). Questi possono essere aggiunti alla nostra mappa come in **Figura 4.1** riportata a pagina seguente.

Avendo messo i pericoli nella pianura, avevo bisogno di una soluzione per essi, e così decisi che per sfuggire al mostro avremmo avuto bisogno di raccogliere il fucile (in altre parole esso dovrebbe essere nell'INVENTARIO), e se aveste il gioiello nella locazione 13 allora il suo potere magico vi proteggerebbe dal pericolo della palude. Quasi una banalità – ma è solo un esempio per dimostrare il principio.

Torniamo alla tastiera per vedere come trattare l'inserimento della fuga dai pericoli.

1. CAMPO COLTELLO	2. SENTIERO PALUDE	0	0	0
0	7. CASA FUCILE	8. FORESTA	0	0
0	0	13. PIANURA GIOIELLO	0	0
0	0	18. LAGO MOSTRO	0	0
0	0	0	0	0

Figura 4.1 Griglia del gioco di Adventure con i pericoli

MODULO 1.8

```
1 goto3
2 save"@0:modulo 1.8",8:verify"modulo 1.
8",8:stop
3 rem
107 :
108 :
109 rem*************************************
**********************
110 rem  inizializza le variabili
111 rem*************************************
**********************
120 sw%=0:mw%=0
387 :
388 :
389 rem*************************************
**********************
```

```
390 rem  ricerca le posizioni coi perico
li
391 rem*********************************
*********************
400 if p=2 and sw%<>2 then 7000
410 if p=18 and mw%<>2 then 7020
587 :
588 :
589 rem*********************************
*********************
590 rem condizioni speciali per evitare
la stampa delle direzioni
591 rem*********************************
*********************
600 if p=2 and sw%<>2 then 900
610 if p=18 and mw%<>2 then 900
887 :
888 :
889 rem*********************************
*********************
890 rem subroutine per evitare movimenti
 per sfuggire al pericolo
891 rem*********************************
*********************
900 ifi$="n"ori$="e"ori$="s"ori$="o"then
print"non puoi farlo...ora!!":goto 550
6987 :
6988 :
6989 rem*********************************
*********************
6990 rem mostra i pericoli
6991 rem*********************************
*********************
7000 print"siete caduti ................
7010 goto 550
7020 print"un mostro e' apparso sul lago
"
7030 goto 550
```

Questo modulo è in realtà un insieme di piccoli moduli legati tra di loro con le altre parti del programma che avete già inserito e salvato su nastro.

Linea 120: queste due variabili sono la chiave per dire al 64 cosa fare quando arriva ad una condizione speciale (in altre parole un pericolo o una simile insolita situazione).

Dando a SW% e MW% (che stanno per Palude e Mostro) un arbitrario valore di 2 una volta che il pericolo è stato superato, posso evitare o no le parti del programma che si riferiscono a quelle speciali condizioni.
Perciò partiamo dando ad esse un valore O (potrebbe essere un altro qualsiasi, naturalmente) in modo che se il computer incontra il pericolo affronterà la parte del programma che vi si riferisce. (Se trovate un po' di confusione nel leggere ciò, non preoccupatevi: dovrebbe diventare tutto più chiaro poi).

Linee 400-410: qui incontriamo effettivamente il pericolo. Se il programma è nella locazione 2 o locazione 18 esso controlla se SW% nella locazione 2, MW% nella locazione 18, sono settati a 2. Noi tuttavia li abbiamo messi proprio a zero alla linea 120 – sicché il programma salta alle linee 7000 e 7020 rispettivamente.

Linee 600-610: poiché abbiamo posto il giocatore di fronte ad un pericolo, dobbiamo evitarlo uscendo dalla locazione, battendo proprio N o S (sennò potrebbe risultare piuttosto facile fuggire). La linea 600 è per la locazione 2, e sottolinea che se SW% non è uguale a 2 (e sarà uguale a 0 finché non avrete superato la difficoltà) allora saltate la routine del movimento e andate (GOTO) alla linea 900. La linea 610 fa esattamente la stessa cosa per la locazione 18. Finché non avete sconfitto il pericolo la variabile SW% sarà posta a 2 – il resto in questa parte di programma è ignorato, e l'usuale procedura di movimento è ancora una volta permessa.

Linea 900: il programma controlla se il giocatore ha tentato di fuggire dalla locazione battendo N, S, E, o O. Se lo ha fatto allora questa linea stamperà "NON POTETE FARLO... ORA!!" e torna al messaggio di richiesta di input ISTRUZIONI?. Per quante volte il comando per spostarsi sia dato, sarà impossibile per il giocatore fuggire dal nostro banale pericolo.

Linee 7000-7030: queste linee mostrano proprio il pericolo sullo schermo e rimandano il controllo al modulo di istruzioni alla linea 550.

MODULO 1.9

```
1 goto3
2 save"@0:modulo 1.9",8:verify"modulo 1.
9",8:stop
3 rem
970 ifi$="nuota"or i$="galleggia" and p=
2 then 2400
1220 ifve$="uccidi"then2500
2100 if p=18 thenprint"Puoi tentare di u
ccciderlo":goto 550
```

Questo è un modulo di tre linee di riempimento, che si inserisce tra le parti del programma, ma non era applicabile prima.

Linea 970: una delle parti più difficili nella programmazione di Adventure è tentare di

pensare a tutte le varie risposte che il giocatore può dare per cercare di scappare dalla sua situazione imbarazzante, una volta che abbia incontrato un pericolo. Svilupperò questo in grande profondità nel programma "*Il Pianeta Dell'Incubo*", nella seconda sezione del libro – ma questa linea è sistemata qui per illustrare la questione.
Ho presunto che, essendo caduti nella palude, due delle più verosimili risposte potrebbero essere NUOTA o GALLEGGIA. Sicché in anticipo questa linea stabilisce che se la risposta è NUOTA o GALLEGGIA (e la locazione è la 2, poiché noi non vogliamo dare una risposta di fuga essendo arrivati alla locazione sbagliata – potrebbe sembrare stupido!!) allora il programma è indirizzato alla linea 2400, la quale è una speciale sezione per i pericoli della Palude (vedi Modulo 1.10).

MODULO 1.10

```
1 goto3
2 save"@0:modulo 1.10",8:verify"modulo 1
.10",8:stop
3 rem
2387 :
2388 :
2389 rem*********************************
**********************
2390 rem subroutine della palude
2391 rem*********************************
**********************
2400 if ob%(3)=-1then print"ok - sei sop
ravvissuto":sw%=2:goto 500
2410 print"sei troppo leggero senza il m
agico      gioiello.sei affogato!":stop
2487 :
2488 :
2489 rem*********************************
**********************
2490 rem subroutine dell' uccisione
2491 rem*********************************
**********************
2500 if p<>18 then 2540
2510 if ob%(2)=-1 then print"ok":mw%=2:g
oto 500
2520 if ob%(1)=-1 then print"proprio con
 un coltello? stai scherzando!":goto 550
2530 print"non ho niente con cui uccider
lo!":goto 550
2540 print"uccidere che cosa?":goto 500
```

Linea 1220: è lo stesso principio della linea 970, ma in questa sezione del programma, osservando il comando VERBO (VE$) si vede che possiamo quasi certamente scrivere UCCIDI qualcosa. Infatti se voi battete UCCIDI semplicemente, il programma risponderà "USATE DUE PAROLE, PREGO". Tentare per vedere. Come prima questa linea indirizza il programma alla linea 2500 se la vostra risposta all'apparizione del mostro nel lago è di tentare di ucciderlo.

Linea 2100: questa linea aggiunge la sezione AIUTO del programma. Sembra sensato mettere alla prova e aiutare allo stesso tempo il giocatore, sicché se arriva alla locazione 18 e batte A questa linea mostrerà il messaggio "POTETE TENTARE DI UCCIDERLO", prima di ritornare al comando "ISTRUZIONI".

Le routines di help tradizionali sono vaghe e confondono, in giochi di Adventure, ma ci sono periodi in cui il giocatore sarà eternamente grato per un suggerimento, in una parte difficile della lotta. Naturalmente potete dare informazioni false o anche condurre deliberatamente il giocatore verso la sua condanna – sembra così carino nelle Adventure!!

Abbiamo raggiunto il modulo finale del mio programma esempio – due piccole sezioni che trattano dei due pericoli che abbiamo introdotto. Se osservate le linee 2400 e 2520 noterete che ho posto le variabili SW% e MW% e 2 finché il pericolo non sia superato con successo – il che permette al giocatore di uscire da questa locazione e assicurerà che quando incontrerà di nuovo il pericolo, questo non sarà mostrato sullo schermo.

Linee 2400-2410: se avete battuto la risposta giusta (dopo essere caduti nella palude) alla linea 2400 il programma per prima cosa controlla se avete il gioiello magico. OB%(3) è il gioiello, se ricordate, e quando OB%(3)=1 allora dovete averlo in inventario. Se è così, allora il programma mostrerà il messaggio "OKAY – siete sopravvissuti" – pone SW% a 2 – e torna indietro alla linea 500 (per mostrare ora le possibili uscite attraverso cui possiamo lasciare la locazione). Se stiamo scrivendo un programma più complicato e vogliamo controllare un numero maggiore di pericoli, avremo bisogno di tornare indietro alla linea 400 per controllare a questo punto altre condizioni particolari.

La linea 2410 è la risposta standard nel caso il giocatore non abbia il gioiello nel suo Inventario, e dopo aver ricevuto l'istruzione NUOTA o GALLEGGIA, mostrerà l'oscura risposta.
"SIETE TROPPO PESANTI SENZA IL GIOIELLO MAGICO. SIETE ANNEGATI!"
e quindi provoca l'END del programma. Questo messaggio darà almeno al giocatore un'idea di quello che dovrà fare la prossima volta. Notate il numero di spazi tra le parole MAGICO e GIOIELLO. Perché questo ingrandisce il messaggio sullo schermo, e non spezza le parole, il che farebbe apparire GIOIELLO parte in una linea e parte in un'altra.

Linee 2500-2540: queste ingannevolmente semplici linee nascondono un bel po' di concetti, e la maggior parte dei principi della programmazione dei pericoli che ho usato ne "*Il Pianeta Dell'Incubo*".

Per prima cosa abbiamo bisogno di trattare la risposta UCCIDI che non riguardasse il mostro del lago. La linea 2500 dirotta semplicemente il programma oltre tutte le risposte

fino alla linea 2540, che è una risposta standard al comando UCCIDI usato in ogni altra locazione.
Ora siamo al momento in cui giocatore ordina UCCIDI IL MOSTRO, alla locazione 18. Ho delineato solamente tre possibilità in questo programma:

1. Il giocatore ha il fucile – e può avere il coltello.
2. Il giocatore ha il solo coltello.
3. Il giocatore non ha nessuno dei due.

Nel Caso 1 (il giocatore ha il fucile), la linea 2510 rileva che OB%(2) è nell'INVENTARIO e dà la risposta "OKAY", pone MW% a 2 e riporta il giocatore in gioco.
Ora egli può fuggire dalla locazione (notate che abbiamo stabilito GOTO 500 e non GOTO 550) e tornare salvo ad un altro stadio. Se il giocatore ha il coltello assieme al fucile, non ci sarebbe nessuna differenza, naturalmente. Finché OB%(2) è vera questa linea è eseguita.

Nel Caso 2 la Linea 2520 riconosce questo fatto, ma non permette al giocatore di fuggire. Così scriviamo una risposta adatta e poi torniamo al comando ISTRUZIONI.
Infine, nel Caso 3 (né fucile né coltello) c'è una risposta standard "NON HO NULLA CON CUI UCCIDERLO", prima di tornare indietro.

È importante aver chiaro che la Linea 2500 o la Linea 2530 saranno stampate se scriverete UCCIDI in qualsiasi altra Linea.

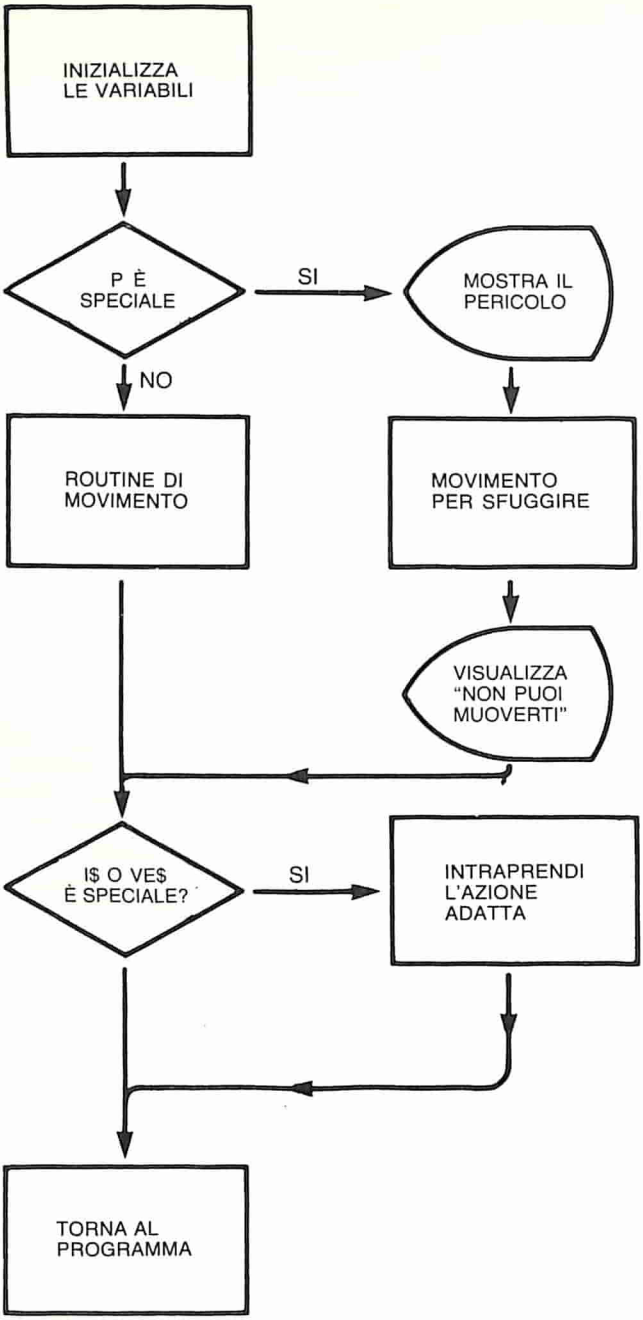
Alcune idee o tecniche di programmazione che ho introdotto in questo Capitolo scalfiscono solamente la superficie della costruzione di una buona Adventure. Infatti questa parte del vostro Programma certamente occuperà il minimo spazio, ma richiede la maggior parte del tempo per essere preparato e corretto in seguito. Il pensare a veramente intricate e frustranti difficoltà adatte al vostro Gioco e quindi vederle funzionare, quando qualcun altro tenta di percorrere i passaggi per uscire, è certamente il più gratificante aspetto della stesura di una Adventure.

Per aiutarvi a capire i problemi di base della programmazione, ho aggiunto un diagramma di flusso che dovrebbe essere applicato in ogni caso, per delineare il problema. (**Vedere Diagramma di Flusso 4.1**) (pag. 50).

Se avete inserito correttamente il Gioco d'Adventure lungo il cammino, dovrebbe essere possibile spostarsi liberamente da una Locazione all'altra finché non si incontra un pericolo – e quindi essere impediti, nel tentativo di uscire, se non possedete l'oggetto appropriato nel vostro inventario. Dovreste essere capaci di raccogliere oggetti, e, quando tornate a quella Locazione, l'oggetto non dovrebbe più essere mostrato. Al contrario, se lasciate cadere un oggetto, allora tornando in quella Locazione esso dovrebbe essere ora mostrato, dopo la descrizione della stessa.

Per verificare la vostra conoscenza dei principi implicati o no, provate ad aggiungere una nuove Locazioni, ed un nuovo pericolo in esse. Potete anzi tentare di aggiungerne parecchie, e anche nuovi pericoli.

Non abbiate paura di far esperimenti – sia sulla carta che sul 64. Il tempo speso ora in questo, vi faciliterà grandemente sia per la comprensione della seconda parte del libro, sia per la vostra abilità nello scrivere più tardi la vostra personale Adventure.



Flowchart 4.1 Pericoli nel gioco di Adventure

PARTE II

CAPITOLO 5

LA STESURA DELLA TRAMA

La maggior parte dei libri e articoli sulla programmazione vi dirà che la stesura di qualsiasi tipo di software dovrebbe seguire alcune direttive ben definite, ed i giochi di Adventure non dovrebbero fare eccezione. Ho trovato che a dispetto dell'ovvio desiderio di sederci davanti alla tastiera e cominciare a programmare direttamente, questo è uno dei momenti in cui il consiglio – *il principale*, io penso – è vitale. So che ciò è noioso – infatti anche il mio stesso stile di programmazione risente in modo fatale dell'attrazione esercitata dall'hardware – ma se non vi sforzate prima di tutto a lavorare sulla carta, senza farvi prendere dall'intreccio della vostra stessa storia, otterrete che (oltre a crearvi problemi futuri) dovrete riscrivere, in un po' di settimane extra, il vostro programma ripetutamente. La trama (o forse si dovrebbe usare un termine migliore), è cruciale, sia per il successo del gioco sia per quanto riguarda la struttura del programma.

In molte fasi il processo di creazione di un'Adventure qualificante è simile ai metodi che i registri usano per costruire un film (un concetto su cui tornerò più avanti nel capitolo) e trovo che sia la fase più emozionante della mia programmazione.

È necessario sia che definiate il tema di base (scrivere la storia – in altre parole) sia che poi sappiate visualizzarla, come se la steste osservando attraverso gli occhi dei vostri spettatori.

Parte dell'emozione in molti dei giochi di Adventure che ho provato, è dovuta alla sensazione di partecipare realmente allo scenario. Se il gioco avrà successo ciò dovrà essere imputato alla combinazione dei temi che spiegherò in questo capitolo, ma che comincio a riassumere in questa fase, per fissarli già da ora nella mente – nel mentre cominciate a stendere l'inizio della vostra trama.
Il successo e la soddisfazione del vostro gioco dipenderanno da:

1. La vostra abilità nell'usare la terminologia adatta per creare immagini nel racconto.
2. La profondità e plausibilità della trama.
3. L'immaginazione della persona che userà il gioco.

A proposito dell'ultimo punto: voi non avete nessun controllo sull'abilità e sull'immaginazione dei potenziali giocatori, ma poiché succede che i giocatori di Adventure sono spesso appassionati di fantascienza e di "fantasy", allora sembra ragionevole supporre che essi abbiano immaginazione ben sviluppata, sicché penso che possiate considerare il punto 3 come assodato. Le altre due caratteristiche giocano perciò un ruolo più importante.

Cominciamo il racconto

Come ho prima accennato, i giochi di Adventure tendono ancora, in parecchi casi, a seguire lo stile delle versioni originali create per grossi computer, o i ben noti giochi di Scott Adams – un viaggio attraverso i temi di "dungeons and dragons", fantascienza, "fantasy" e altri ancora. Una veloce scorsa ad una qualsiasi rivista farà saltar fuori una varietà di titoli come "Il Castello Misterioso", "Il Covo di Dracula", "L'Isola Del Destino", "Il Racconto Del Drago" e via di seguito.
Naturalmente non dovreste aver bisogno di seguire questa tendenza, e ci sono parecchi giochi con una trama totalmente diversa (fuga dal riformatorio, alla ricerca dell'ideale compagno della vita, l'avventura di una notte passata fuori città) che aggiungono un gradito tocco di originalità per giocatori smaliziati. Dal momento però che il tema "fantasy" è così popolare (ed uno di quelli che più mi divertono) ho basato il mio gioco sul genere di trama che si riferisce a questo familiare filone.
Ho letto una volta che non ci sono trame originali per i racconti – solo differenti variazioni. Ciò è naturalmente vero, ma "Guerre Stellari" è una perfetta dimostrazione di abilità nel prendere una trama semplice e trasformarla in un successo strepitoso! In tutte le nostre storie abbiamo bisogno di alcuni generi di esplorazioni o di scopi da raggiungere (trovare tesori, liberare la principessa, sfuggire da una situazione pericolosa, scoprire il significato di un enigma, ecc.). Abbiamo bisogno di un'eroina o di un eroe ben riconoscibili (nell'Adventure il giocatore acquisisce tale ruolo) e, di solito, o di un malvagio o di qualche altro conflitto per i nostri personaggi principali.
Quando scrissi la mia storia usai i seguenti passaggi:

1. Scegliere l'ambientazione (per es. fantasy, orrore, fantascienza).
2. Scegliere la ricerca o lo scopo (per es. trovare tesori, sfuggire ad un mago).
3. Decidere tra il ruolo di eroe/eroina.
4. Scegliere i personaggi principali (per es. mago, vampiro, contessa).
5. Scrivere un brogliaccio del racconto.
6. Tracciare una mappa semplificata con un po' di Locazioni base.
7. Tradurre la storia in tavole.

Può sembrare che ci sia molto duro lavoro da fare prima di por mano alla tastiera, ma molti dei passi della stesura della vostra trama seguiranno in modo così naturale che tutto diventerà affascinante e tenderà da solo alla meta. E, come ho già lungamente detto, il tralasciare questa fase o causeràpiù tardi un lavoro aggiuntivo, oppure alla fine farà risultare un prodotto non soddisfacente.

La scelta dell'ambientazione

Il tipo tradizionale di gioco d'Adventure che più pesantemente prende spunto dal mondo della fantasy di Tolkien e degli scrittori che a lui si ispirano, abbonda di elfi, draghi, streghe, castelli trasudanti umidità, magia, misteri e cose simili. Forse è in realtà la nostalgia dei racconti di fate della nostra infanzia che spiega in parte questa popolarità, forse una ragione più profonda, ma per il crescente numero di appassionanti di Adventure, la gamma di possibilità aperte dalla scelta di questa ambientazione aggiunge motivi e originalità che sarebbero meno realistici nella più prosaica realtà quotidiana. Dopo tutto ogni cosa è possibile nel nostro racconto!
"Spade e streghe" è un filone di questo genere di ambientazione che sostituisce con

violenza gli aspetti del magico. In questo genere (esemplificato dalle opere di Robert E. Howard e dal suo splendido eroe Conam) il mondo è uno scuro e selvaggio posto dove incantesimi e streghe sono reali, e la vostra abilità con la spada è il solo efficace vantaggio che avete.
Pur essendo un'eccellente miniera per il rifacimento di giochi di Adventure, la mancanza di malia presente nel mondo di Tolkien è la principale ragione per la mia preferenza per le Adventure fiabesco-fantastiche.
Anche i temi dell'orrore tradizionale sono estremamente popolari. Dracula gioca il ruolo della star, in parecchie di queste Adventure. Di nuovo l'immaginazione del programmatore è scatenata con una varietà di stimolanti possibilità – persone che all'improvviso diventano vampiri, cripte buie ed umide, cunicoli dove l'inaspettato può balzare fuori appena voltate l'angolo, limiti di tempo concessivi per fuggire prima che vi ritrasformiate in vampiri ecc. Quest'ambiente si presta a situazioni con case vastissime, con un gran numero di stanze in cui ci si può muovere, discese agli inferi per affrontare il Diavolo e riscattare la vostra anima, fughe dalle isole dei Voodoo, dove gli zombi attaccano ad ogni angolo, lupi mannari, adoratori di Satana ed altri ancora. Secondo me questo tipo di ambientazione è il migliore per varietà, quando si deve pensare ad un intreccio.
Un'altro mondo zeppo di possibilità (e così relativamente ignorato dagli scrittori) è quello degli eroi delle "comic strip". Superman, Batman ed altri hanno dimostrato da decenni di poter suggerire una varietà di temi differenti (ancora gli stessi schemi di base, arrangiati in modo differente).
Alcune rapide idee che vengono alla mente sono: un A-man (ovvero un Adventureman, se non l'avete già intuito!) che insegue il Super-malvagio Bug-man sopra l'intera Commodore City, prima che egli riveli la sua identità segreta, un A-man che salva "Luigi Left" dal diavolo "Interfaccia" che sta tramando di conquistare il mondo. Sono sicuro che questo sia un genere ricco di spunti.
Ci sono molte altre situazioni che potete utilizzare, come ho detto, l'investigatore che scopre il crimine, l'innocente coinvolto nel mondo dello spionaggio, il naufrago sull'isola deserta. Dovete solo guardare i programmi della TV, i libri di fantascienza della vostra libreria o i film che si proiettano al vostro cinema, per trovare immediate idee per la trama. E infine c'è la fantascienza. Ho lasciato per ultima la fantascienza poiché è l'ambito che ho scelto per la mia avventura – "*Il Pianeta Dell'Incubo*". I fans puristi della fantascienza potrebbero probabilmente argomentare che *Il Pianeta Dell'Incubo* è più una "space opera" che fantascienza, ma per onor di semplicità io considero tutte le storie con problematiche di spazio e tempo, come fantascienza. Di nuovo, le possibilità sono piuttosto vaste: viaggio nel tempo per salvare il buon Dottore risucchiato nel futuro dall'azionamento prematuro della sua macchina del tempo, battaglie contro gli alieni che mirano ad invadere la terra, ricerca di un pianeta post-nucleare adatto alla vita dopo l'olocausto. È in questo genere di ambientazione generale che ho deciso di situare la mia storia.

Scegliete una ricerca o una meta

Può sembrare strano rivolgerci a ciò prima di ogni considerazione circa la stessa trama, ma poiché l'intera idea di un gioco di Adventure è risolvere un enigma, trovare una risposta, raggiungere una meta – così la principale considerazione veramente prioritaria deve essere il decidere quale sarà la vostra meta.

Quando cominciai la prima volta a lavorare a *"Il Pianeta Dell'Incubo"*, avevo una sola meta – liberare la Principessa Aurora. Dal momento che la trama si andava espandendo durante la fase programmatoria, ho aggiunto un secondo obiettivo – trovare il cristallo energetico, e portarlo all'astronave. Questo ha aggiunto difficoltà al gioco ed ha esteso l'ampiezza dell'Adventure in modo abbastanza considerevole, ma non era in verità una parte essenziale della storia originale. Perciò a dispetto dell'importanza dello scegliere la vostra meta, è possibile fare revisioni più tardi o, come ho detto, delle aggiunte.
Non tentate di iniziare, in questa fase, ad aggiungere dettagli troppo particolareggiati. Per esempio supponete di aver fissato come meta TROVARE IL TESORO, nel castello del Destino. Quando la vostra immaginazione comincia a delineare la trama, potreste anche pensare a come aggiungere un punteggio al gioco, sommando 10 punti per ogni informazione sul tesoro. Mentre la vostra mente sta pensando a ciò *potreste* orientarvi ad aggiungere il principio di sottrarre punti a causa di varie difficoltà incontrate, e che il vostro giocatore non ha superato in un determinato limite di tempo. Da ciò voi *potreste* decidere di aver bisogno di evidenziare (per tutto il tempo) il punteggio sullo schermo – in modo da sedervi davanti al 64 ed iniziare a trattare la parte grafica della vostra impaginazione del punteggio...
L'essenza di una buona stesura della trama in questo stadio è la semplicità. I dettagli arriveranno più tardi, una volta che avrete iniziato il programma.

Decidete sul ruolo dell'eroe

Questo – molto semplicemente – coinvolge il giocatore nello scenario. Dal momento che la partecipazione nell'avventura è la chiave per giocare in modo avvincente, avete bisogno di far sentire al vostro potenziale giocatore di essere lui l'eroe.
Avete due opzioni principali ora:

1. Il vostro giocatore agisce come se esso stesso fosse proiettato nel mondo della fantasia.

2. Il vostro giocatore è rivestito del ruolo di un eroe di fantasia.

Non penso che importi troppo quale delle due scegliate – mentre dovrete averlo proprio chiaro all'inizio del gioco.
Nel mio caso personale ho voluto che l'eroe fosse il pilota di una ammaccata ma affidabile astronave (l'ombra di Han il Solitario) che lo ha ridotto a fare la vita del trasportatore.

Scegliete gli altri personaggi

Naturalmente gli altri personaggi della trama dipenderanno dalla storia che state scrivendo, sicché questa fase dovrebbe in realtà essere vista allo stesso tempo come creazione vera e propria della storia.
Ma dato che queste fasi sono un po' artificiose... (ciò che realmente accade è che mentre state pensando alla vostra storia automaticamente penserete anche all'Eroe, alla Locazione, al Malvagio ecc.) è più facile se avete alcuni generi di struttura che assicurano che non vi siete dimenticati di nulla.
I personaggi principali devono essere: complici, persone da liberare, marrani, e vari tipi per aggiungere colore locale o che agiscano come "pesci pilota" o suggeritori di tracce.

Il Pianeta dell'Incubo è piuttosto carente nei personaggi a causa del suo ambiente – un pianeta alieno che contiene svariati pericoli piuttosto che personaggi malvagi – così che l'unico altro vero personaggio è la Principessa Aurora.
Un aspetto della stesura di questo tipo di gioco è l'ovvio problema del sesso del giocatore. In ogni racconto dove il giocatore rivesta il ruolo di un uomo che tenta di liberare una donna, si potrebbe argomentare che il gioco attrarrà solo i maschi, specialmente in alcune fasi dell'intreccio, (ciò apparirà ovvio a chiunque lo abbia già giocato!). Nonostante io non abbia fatto ciò, dati gli scopi di questo libro – dovrebbe essere questione semplice inserire una domanda all'inizio del gioco per chiedere se il giocatore è maschio o femmina. Non appena si riceve l'apposita risposta il gioco potrebbe quindi fissare alcune variabili, di modo che la Principessa potrebbe diventare un Principe ed il giocatore diventare una Pilota spaziale femmina. Secondo me questo aggiunge un elemento di flessibilità ai giochi di Adventure che manca negli altri "media", come libri e film.

Scrivete una sinossi della storia

Possono venire idee nei momenti più strani, spesso in bagno, a notte fonda o, nel mio caso, sul treno. Mentre sferragliavo da Manchester a Londra, una bigia mattina di maggio, l'intera trama mi si presentò in modo pressoché inaspettato e repentino, e la scribacchiai giù proprio lì. Questo costituì la sinossi – che è rimasta la stessa nei contenuti di base, durante il processo di creazione e programmazione dell'intero gioco. La mia sinossi era del tipo seguente:

Siete il pilota di una astronave in missione per liberare la bella Principessa Aurora sul pianeta Zen dove sta per essere maritata al tiranno Ruler. Vi siete innamorati di Aurora ma non osate dirglielo.
Un'improvvisa caduta di potenza o uno stormo di meteore ha provocato la vostra caduta su uno strano, inesplorato pianeta. Siete atterrati con successo con la vostra navicella, ma siete svenuti durante la caduta.
Quando rinvenite scoprite che la nave è stata violata dall'esterno e la Principessa catturata. Dovete porvi in cammino e liberarla.
L'atmosfera del pianeta è venefica, sicché siete obbligati ad indossare per prima cosa la vostra tuta spaziale, ma lungo la via ve la levate non appena trovate una pianta aliena, con strani frutti che (se mangiati) vi permettono di respirare l'aria senza pericolo.
Attraversate un vasto deserto fino ad una città in rovina, nel vostro girovagare, ma tutto ciò che vi trovate è un serpente gigante che vi attacca.

Arrivate finalmente ad una smisurata foresta e continuate la ricerca. Vi perdete, e dopo un po' di tempo scalate un albero per vedere dove siete. Verso sud vedete salire del fumo – vita, o qualcosa del genere.
Sulla via verso quel posto venite attaccati da un dinosauro (ovviamente il pianeta non si è sviluppato ancora molto lungo il cammino dell'evoluzione) e infine trovate un villaggio di capanne di fango appartenenti agli indigeni che abitano questa parte del pianeta.
Avete scoperto che Aurora è considerata dagli indigeni come una Dea. Essi vi minacciano quando tentate di portarla via – ed è solo quando la baciate, e dimostrate loro che siete il suo compagno destinato, che essi concedono ad entrambi di fuggire.
Lungo la via del ritorno alla vostra astronave, scoprite che in realtà Aurora vi ha amato

per tutto il tempo (ovviamente c'era qualcosa di magico nel vostro bacio!!) e volate insieme verso un felice futuro, ai margini della Galassia.
Può non essere elevata letteratura – ma l'essenza della mia storia è tutta qui. Quando cominciai a sviluppare il racconto aggiunsi rifiniture e miglioramenti lungo la via – tutto

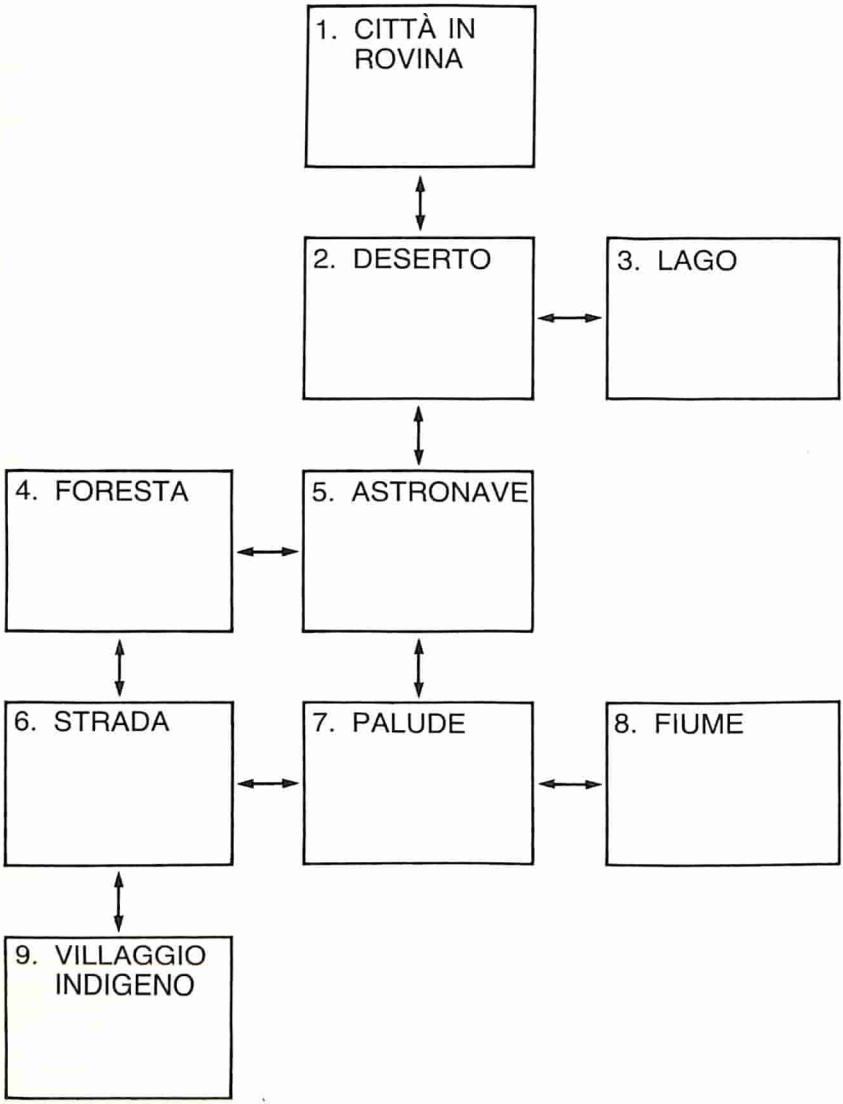


Figura 5.1 Mappa iniziale per il pianeta dell'incubo

sommato pochi però. Molta parte del racconto è piuttosto prevedibile ma in un certo senso queste aggiungono interesse al gioco. Spesso ci piace "sentire" che il film che stiamo vedendo o il libro che stiamo leggendo è "giusto" – in altre parole ci aspettiamo che accadano certi eventi in una particolare situazione – e questo accresce la soddisfazione quando scopriamo che ciò infatti è esatto. È questo il motivo per cui ho cercato di non essere troppo strano o fuori dal comune.

Tracciate la mappa iniziale

Ho già descritto le basi delle mappe in un precedente capitolo – ma in questa fase della creazione di una vera Adventure è necessario descrivere ancora un altro tipo di mappa, che sarà convertita poi nella forma della griglia descritta nel Capitolo 2.
Avete bisogno di alcune idee sul legame geografico esistente fra le varie Locazioni nella vostra Adventure, che vi aiutino ad evitare di commettere errori nel vostro futuro progetto. Trovo che è più semplice se tracciate direttamente una vera mappa di base (vedi **Figura 5.1**) che situi le varie Locazioni in uno "spazio reale", piuttosto che tentare di adattarle all'interno di una griglia.
A questo stadio considerai un numero di Locazioni abbastanza piccolo (infatti variava da 7 a 9 all'inizio) perché sapevo che avrei dovuto aumentare alcune sezioni in modo abbastanza considerevole, più tardi. Nella versione finale, sia la città in rovina che la foresta, contengono circa 11 Locazioni diverse ciascuna – che mi permettono libertà di piazzare oggetti e pericoli dovunque io voglia.
La cura nel tracciare questa mappa è stata di grande aiuto nello svolgimento della storia, quando cominciai a pensare agli oggetti e pericoli che avrei voluto iniziare a sistemare in Locazioni specifiche, e quindi mi accorsi che la struttura della mia trama si stava espandendo a macchia d'olio. La **Figura 5.2** (riportata a pagina seguente) mostra l'inizio di ciò e un'analisi della mia storia potrebbe illustrare come questo mi è stato di aiuto.
Questa mappa può essere ora espressa in forma di tavola (**Tavola 5.1**) che è un eccellente modo di raccogliere le vostre idee in questa fase della progettazione della vostra Adventure.

Tavola 5.1

LOCAZIONE 1 CITTÀ IN ROVINA	Pericolo: Serpente Oggetto: Fune (da usarsi nella palude)
LOCAZIONE 2 DESERTO	Oggetto: Frutta su un arbusto (bisogna mangiarli)
LOCAZIONE 3 LAGO	Pericolo: Anguilla
LOCAZIONE 5 ASTRONAVE	Pericolo: Atmosfera velenos Oggetto: Tuta spaziale (da indossare all'esterno) Oggetto: Mitra (per uccidere il serpente e il dinosauro)
LOCAZIONE 6 STRADA	Pericolo: Dinosauro
LOCAZIONE 7 PALUDE	Pericolo: Sabbie mobili
LOCAZIONE 8 FIUME	Oggetto: Coltello (per uccidere l'anguilla)
LOCAZIONE 9 VILLAGGIO	Pericolo: Indigeni ostili Oggetto: Principessa Aurora

La tavola riportata a pagina precedente ha bisogno di alcune spiegazioni perché nel progettare la mappa iniziale avevo anche previsto le difficoltà presenti dell'Adventure. Vi do un semplice esempio:
nella LOCAZIONE 1 (la città in rovina) c'è una fune necessaria per sfuggire alle sabbie mobili nella palude (LOCAZIONE 7) più avanti. Ma per prendere la fune dovete prima uccidere il serpente – e non potete farlo prima di ricordarvi di prendere la mitra-

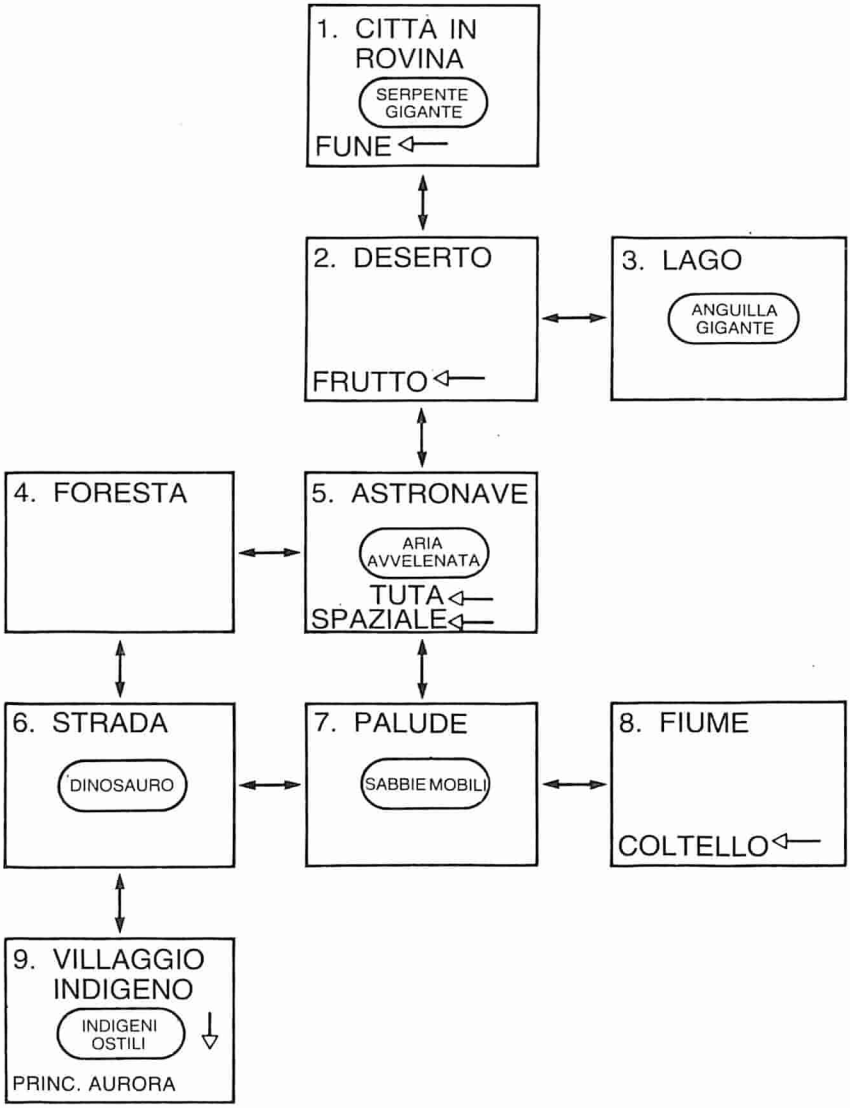


Figura 5.2 Mappa iniziale per il Pianeta dell'incubo con oggetti e pericoli

glietta prima di lasciare la LOCAZIONE 5. Una volta raggiunta la LOCAZIONE 9 (villaggio indigeno) non potete liberare Aurora senza baciarla. Non potete baciare la Principessa senza togliervi la tuta, e se lasciate la vostra tuta prima di aver mangiato la frutta nella LOCAZIONE 2 (il deserto) allora morirete nell'atmosfera velenosa.
Dopo aver tracciato la mappa iniziale e aver pensato qualcosa circa la storia, il prossimo passo è iniziare a disegnare su tabelle la vostra trama.

Mettete in tabella la trama

Io sono anzitutto una persona che ama gli spettacoli visivi. Mi piacciono i film, preferisco le storie illustrate alla prosa (ciò significa fumetti!) e quando comincio a scrivere tendo a vedere con gli occhi dell'immaginazione il prodotto finito, prima di mettere mano al "word processor".
Perciò è stato naturale per me cominciare a scrivere la mia trama usando la tecnica delle tavole illustrate simile al procedimento di un regista che le usa spesso, nelle fasi iniziali, per progettare un film.
Una tavola è proprio una raccolta di immagini visive che ritraggono la *storia* in un insieme di *strisce* – un "strip-cartoon" del film in altre parole. Ovviamente lo scrivere un programma per computer che fa pesantemente riferimento alla parte scritta non è la stessa cosa che riprendere un film d'avventura. Se ho usato un simile principio, allora ciò di cui ho bisogno è "vedere" prima quello che il giocatore dovrebbe vedere poi sullo schermo del computer.
Ho bisogno di immaginarmi il tracciato del testo sullo schermo – e di figurarmi alcune possibili risposte del mio potenziale giocatore in modo da poter cominciare a pensare alla struttura della trama.
All'inizio questa tecnica può sembrare una perdita di tempo – dopo tutto chi vuole occuparsi di scrivere le immaginarie risposte a "COSA POSSO FARE ORA?" su un pezzo di carta, quando c'è la tastiera da usare – ma nel mio caso trovo che questa parte della costruzione è estremamente valida.
Presumo che ciò sia dovuto al fatto che sono capace di iniziare a "strizzar-pensieri" (una frase che ho preso in prestito da un mio caro amico perché è così adatta a descrivere il procedimento). Questo significa che sono obbligato (dal fatto di scrivere su carta varie idee e risposte potenziali) a concentrarmi molto più profondamente sulla trama e sulle Locazioni. Cominciai a raccogliere idee che potessero essere di aiuto più avanti. In breve – stavo realmente pensando alla storia. La **Figura 5.3** può spiegare quello che intendo.
La fraseologia della mia prima tavola è un po' troppo vasta per l'ampiezza della memoria del 64, ma riassume il senso del gioco. E sebbene l'effettivo vocabolario che verrà usato più avanti nel programma sia differente, in questa fase una o due idee che avrei potuto usare più tardi, si sono imposte – idee che avrei già potuto usare, ma non ho fatto in questa prima fase.
Un semplice esempio è il "personal robot Proteus" che è sopravvissuto alla caduta e che dice sullo schermo "...posso aiutarvi nella ricerca di Aurora."

LOCAZIONE: dentro la vostra astronave.
Vi svegliate con la testa che scoppia, in mezzo ai rottami dell'arredamento della vostra cabina. Non appena vi rialzate malfermo sui piedi vi torna improvviso alla mente il ricordo della vostra Astronave che, perduto il controllo, vi precipita giù.
La vostra missione – liberare la bella Principessa Aurora dal pianeta Thoth, dove sta per essere maritata al tiranno Zorn-Ramok, un essere crudele che vede l'unione solo in termini politici.
Il vostro problema – vi siete innamorati della Principessa e, a sua insaputa, avete guidato la vostra navicella nei solitari spazi esterni della Galassia, nel tentativo di persuaderla a dimenticare la sua promessa di matrimonio a Zorn-Ramok, e a fuggire con voi.
Ma il Fato ha messo il suo zampino nei vostri piani, per cui la vostra astronave è stata danneggiata da un improvviso fascio di ioni, e tutto quello che avete potuto fare è pilotarla su un pianeta inesplorato, agli estremi limiti della Galassia, e tentare di atterrare.
Ora vi riavete in mezzo alle macerie della vostra malridotta navicella. Vicino a voi giace il contenuto del vostro baule, la tuta spaziale, carte galattiche, la vostra mitraglietta, ed i segni di un'evidente lotta. Ma Aurora se n'è andata.
La vostra cabina di guida mostra di essere stata aperta dall'esterno e quindi richiusa. La vostra sola fortuna è che Proteus, il vostro personal-Robot, è intatto, e può aiutarvi nella ricerca di Aurora.

COSA POSSIAMO FARE ORA?

Figura 5.3 Dentro l'astronave

All'inizio abbozzai l'idea di introdurre un piccolo robot che avrebbe potuto gironzolare intorno, dicendo battute comiche o lazzi sullo schermo – la mia prima tavola illustrata. Questa idea potrebbe essere modificata e migliorata, per far diventare Victor il robot (uno "sprite" nel 64), e farlo scendere giù dall'alto dello schermo in vari momenti del gioco. Mi chiedo se avrei mai pensato a Victor se non avessi passato il tempo rimuginando e riportando, all'inizio, il contenuto dello schermo.

LOCAZIONE: Vicino ad un fiume immenso.
La giungla si dirada e all'improvviso vi trovate sulle rive di un immenso fiume, un vasto tratto d'acqua più simile ad un mare, tuttavia rapido di corso oltre i vostri piedi, con la velocità di un rovinoso vortice.
Troppo largo da attraversarsi a nuoto, vi sentite scoraggiati. Essere arrivati così lontano solo per sentirvi ora impotenti!
Attorno a voi vi sono delle rocce e i resti di un'antica era, che sembrano essere stati abbandonati da migliaia d'anni. Vi siedete per un istante, guardando fisso oltre questo rapido fiume, prima di decidervi a tornare indietro.

Figura 5.4 Di fronte ad un immenso fiume

LOCAZIONE: Foresta.
Vagate in una vasta foresta, piena di alberi alti come grattacieli, con tronchi grossi come case.
Dovunque l'aria pullula d'insetti, sfreccianti e balenanti nella luce solare che filtra dall'alto. Ogni tanto un piccolo animale fugge dal nascondiglio e attraversa la vostra strada, sparendo nella foresta.
Mentre camminate vi rendete conto di andare sempre più in profondità; per l'aria si diffonde un gelo, e la luce svanisce mentre gli alberi si fanno più fitti e vicini tra loro. Presto ogni traccia del sentiero sparisce, e dovete lottare attraverso rovi e cespugli.
Vi siete persi. Con un brivido vi rendete conto che non c'è modo di poter ricordare per quale via avanzare o tornare indietro. E dietro di voi ci sono strani rumori, come se qualcosa di immane vi seguisse lentamente e con sicurezza.

Figura 5.5 Dentro la foresta

Questi due quadri, nella **Figura 5.4** e **Figura 5.5**, descrivono proprio delle altre locazioni, ancora per mostrarvi come io pianifico il mio racconto. Spero che l'atmosfera del gioco possa essere recepita, attraverso la fraseologia di queste tavole, e sebbene non sia stato capace di mantenere la stessa lunghezza delle frasi sul 64, penso di aver afferrato l'essenza del fascino de "*Il Pianeta Dell'Incubo*".

LOCAZIONE: Sulle rive di un lago
COSA POSSIAMO FARE ORA?
Tuffo
NON POTETE FARLO... ANCORA!
COSA POSSIAMO FARE ORA?
Nuotare
O.K.
Avanzate silenziosamente sul filo dell'acqua, vedendo in basso uno scintillante oggetto di metallo, sul fondo del lago. È un'altra mitraglietta dell'astronave.
COSA POSSIAMO FARE ORA?
Tuffo
O.K.
Una mostruosa anguilla gigantesca vi attacca, come apparendo dal nulla.
COSA POSSIAMO FARE ORA?
Sparare con la mitraglietta
IMPOSSIBILE – È troppo bagnata.
COSA POSSIAMO FARE ORA?
Uccidere il mostro.
NON SO COSA SIA 'MOSTRO'
COSA POSSIAMO FARE ORA?
Uccidere l'anguilla

Figura 5.6 Alla riva del lago

La **Figura 5.6** è una delle numerose tavole che ho composto, avanzando nel mio racconto – tentando di immaginare di star digitando le risposte appropriate ai messaggi del computer. Quando arrivai all'effettiva programmazione di questa sequenza (Capitolo 12) ero già abbastanza sicuro di quello che volevo – e sebbene abbia cambiato dappertutto un po' di idee, l'essenza è rimasta nella versione finale de "*Il Pianeta Dell'Incubo*".

Sommario

Questo Capitolo potrebbe aver rappresentato la parte più dura della stesura della vostra personale Adventure, e la più frustrante per programmatori smaliziati – non abbiamo ancora acceso il computer!
Ma ho la sensazione che gli appassionati siano persone dotate di immaginazione e creatività e possano supporre che una volta che l'idea di un filo conduttore è sviluppata nella mente, sia molto difficile sradicarla. Se però ciò non accade allora la sola via di salvezza è di riportare i concetti su carta, e solo poi sul computer.
Spero di aver ben introdotto il reale procedimento di come si traduce quel filo conduttore, ed il concetto che ciò che renderà unica e funzionale la vostra Adventure contribuirà anche a risparmiare tempo ed evitare stress più tardi.
Ho speso un po' di tempo su questo perché tutti i libri ed articoli che ho letto finora sulla programmazione di Adventure, si concentrano sull'abilità nella programmazione, e tendono a sorvolare sulla parte difficile – l'ideazione.
Un ultimo suggerimento prima che passiate al Capitolo 6 e cominciate a battere sui tasti – prendete tempo, nello sviluppare la vostra storia direttamente. Non c'è fretta di finire. Il tempo speso ora nel pensare sarà fruttifero per il vostro sforzo finale.

CAPITOLO 6

DEFINIAMO LA SCENA

I giochi su computer, come ogni altra cosa nella vita, possono aver successo o fallire nei primissimi istanti. Noi tutti prendiamo rapide decisioni su persone, luoghi, libri, lavoro, ecc. basate sulle prime impressioni – giuste o sbagliate – e così questi primi pochi secondi del nostro gioco diventano molto importanti.
Un buon Gioco dovrà partire con un buon titolo – uno che richiami l'attenzione e la mantenga desta. Quando mettete una cassetta nel vostro lettore, aspettate per lunghissimi minuti che la carichi, e se vi si presenta un titolo carente, siete giustamente dispiaciuti. E anche se non lo siete – potreste esserlo! Titoli buoni devono avere una parte grafica ben scritta (e affascinare al modo giusto) e parole che diano il sapore ed il gusto del gioco che sta per iniziare.
Non c'è nessun bisogno di ammattire per questo, (ho visto alcuni giochi che occupavano virtualmente tutta la memoria e la maggior parte dell'abilità del programmatore concentrata sul titolo, a scapito del programma stesso). Ma ho pure visto titoli sconcertanti, con parole scentrate, battute in modo scorretto, e nessun tentativo di cambiare colori o di sistemare qualche grafico qui e là.

I concetti di partenza

Nell'ultimo capitolo ho descritto la trama di base della mia storia – ma prima di partire a sviluppare ulteriormente l'intreccio sul 64 ho voluto sensibilizzarmi bene sul gioco, progettando e programmando un titolo che ho trovato apprezzabile.
Dato che è possibile creare il proprio set di caratteri mi sono divertito con l'idea di disegnare mie proprie lettere per le parole "*Il Pianeta Dell'Incubo*", ma il tempo e la capacità di memoria alla fine me lo ha impedito, sebbene mi fossi molto sforzato nel provare varie alternative. Alla fine decisi che avrei preferito un titolo più semplice con un'astronave in qualche posto dello schermo, e per valorizzare tutto ciò avrei dovuto spostare lo sprite attraverso le parole, preferibilmente con un rumore sibilante.
Dovrò trattare con lo stesso sprite più avanti, nel Capitolo 8 – per cui ora partiamo con il solo titolo.

N.B. Ho suddiviso il programma in pezzi chiamati moduli, in questa sezione del libro, come ho già fatto nella sezione 1, tuttavia qui ci sono pochi posti dove si devono tralasciare delle linee nei primo moduli, in modo da rendervi possibile inserire e far girare il vostro programma per provarlo. Così dato che queste linee mancanti saranno aggiunte più tardi, vi raccomando di esser sicuri di seguire attentamente la numerazione, altrimenti scriverete delle linee sopra le altre – con un sacco di problemi in fase di prova. Ho avuto anche problemi con la memoria, di cui ho raggiunto i limiti in parecchie occasioni. Per riguadagnare memoria sono stato obbligato a rinumerare il programma

e a limitare il numero di asterischi nelle REM a non più di due. Pensai anche di eliminare tutte le REM – ma decisi ciononostante di mantenerle, poiché rendono più facile seguire il programma e comprenderlo – specialmente dopo parecchi mesi.

MODULO 6.1

```
1 goto3
2 save"@0:modulo 6.1(a)",8:verify"modulo
 6.1(a)",8:stop
3 rem
100 rem**
110 rem Pre-Punteggio
120 rem**
130 Print"■":Poke53280,0:Poke53281,0
140 Print"■■■■■scagliati in lontano futu
ro ....."
150 goto280
250 rem**
260 rem variabili
270 rem**
280 clr:v=53248:sc=54272:bs=53280:bc=532
81
290 pr=0:cv=0:sv=0:ev=0:qv=0:dv=0:es=0:n
v=0:na=0:np=0:fv=0:rc=0:yy=0:hh=0:hr=0
300 vo=54296:w1=54276:a1=54277:s1=54278:
w2=54283:a2=54284:s2=54285
310 w3=54290:a3=54291:s3=54292
320 h1=54273:l1=54272:h2=54280:l2=54279:
h3=54287:l3=54286
970 rem**
980 rem titolo
990 rem**
1000 Pokebs,0:Pokebc,0:Print"■":ss=1024:
cs=55296
1010 Poke53272,21
1020 fori=1to23
1030 Pokecs+40*i,4:Pokess+40*i,102
1040 Pokecs+40*i+38,4:Pokess+40*i+38,102
1050 nexti
1060 Print"■■■■■
```

```
e"
1090 print"▚▚▚▚▚▚▚▚H"
1100 print"▚▚▚▚▚▚ ▚"
1110 print"▚▚▚▚▚B+B▚
     Q"
1120 print"▚▚▚▚▚▚BQB"
1130 print"▚▚▚▚▚▚BQB"
1140 print"▚▚▚▚▚▚BQB▚▚    ▚*"
1150 print"▚▚▚▚▚B+B ▚"
1160 print"▚▚▚▚BB▚B ▚▚
          *"
1170 print"▚▚▚▚▚BB▚B ▚▚"
1180 print"▚▚▚▚▚   ▚▚"
1190 print"▚▚▚▚▚▚▚▚
```

Modulo 6.1 che è stata convertita a Standard CT.
Per quelli di voi meno familiari con tutto ciò, sarà più semplice che lo prendano così com'è (sembra proprio che io voglia spaventarvi!!).
Perciò CLS sta per "CLEAR SCREEN" (pulire lo schermo) e CU e CD stanno per "CURSOR UP" (cursore su) e "CURSOR DOWN" (cursore giù) rispettivamente. G‹ e G› si riferiscono a "GRAPHIC LEFT" (simbolo grafico a sinistra) e "GRAPHIC RIGHT" (simbolo grafico a destra) sicché G< + significa tasto "+" quando il tasto Commodore è azionato – e ciò fa risultare l'appropriato simbolo grafico.
Se osservate la linea 1180 nel Modulo 6.1 (b) ve lo dimostrerò più chiaramente possibile:

1180 PRINT" 3CR – cursore a destra di tre posizioni –
GRN – tasto CTRL e tasto del colore verde –
G<K – grafico sulla sinistra del tasto K –
3SPE – tre spazi –
REV – tasto CTLR con REVERSE ON –
G>K – grafico sulla destra del tasto K –
OFF – tasto CTRL con REVERSE OFF"

Quindi se fate riferimento al Modulo 6.1 (b) così come al Modulo 6.1 (a) non dovreste avere difficoltà nel battere questa parte.

MODULO 6.1(b)

```
970 REM**
980 REM TITOLO
990 REM**
1000 POKEBS,0:POKEBC,0:PRINT"[CLR]":SS=1024:C
S=55296
1010 POKE53272,21
1020 FORI=1TO23
1030 POKECS+40*I,4:POKESS+40*I,102
1040 POKECS+40*I+38,4:POKESS+40*I+38,102
1050 NEXTI
1060 PRINT"[CR][PURPLE][CYAN][CYAN][CYAN][CYA
N][CYAN][CYAN][CYAN][CYAN][CYAN][CYAN][CYAN][
CYAN][CYAN][CYAN][CYAN][CYAN][CYAN][CYAN][CYA
N][CYAN][CYAN][CYAN][CYAN][CYAN][CYAN][CYAN][
CYAN][CYAN][CYAN][CYAN][CYAN][CYAN][CYA
N][CYAN]"
1070 PRINTTAB(11)"[CD][CD][CD][WHITE]IL PIANE
TA DELL'INCUBO"
1080 PRINT"[CR][CR][CD][CD]AVVENTURA SPAZIO-T
EMPORALE".
1090 PRINT"[CD][CD][CR][CR][CR][CR][CR][GREEN
]|"
1100 PRINT"[CR][CR][CR][CR][RVS ON][CYAN] [CY
AN]"
1110 PRINT"[CR][CR][CR][CR][RVS ON]|+|[RVS OF
F]                          [RED]●"
1120 PRINT"[CR][CR][CR][CR][RVS ON][GREEN]|●|
"
```

```
1130 PRINT"[CR][CR][CR][CR][RVS ON][GREEN]|●|
"
1140 PRINT"[CR][CR][CR][CR][RVS ON][CYAN]|●|[
CYAN][RVS OFF]   [PURPLE]*"
1150 PRINT"[CR][CR][CR][GREEN][RVS ON]|+| "
1160 PRINT"[CR][CR][CR][RVS ON]|[CYAN]| [RVS
OFF]                                [YELLOW]*"
1170 PRINT"[CR][CR][CR][GREEN][RVS ON]|[CYAN]
| [RVS OFF]"
1180 PRINT"[CR][CR][CR][CR][GREEN]   [RVS ON]
[RVS OFF]"
1190 PRINT"[CR][CD][PURPLE][CYAN][CYAN][CYAN]
[CYAN][CYAN][CYAN][CYAN][CYAN][CYAN][CYAN][CY
AN][CYAN][CYAN][CYAN][CYAN][CYAN][CYAN][CYAN]
[CYAN][CYAN][CYAN][CYAN][CYAN][CYAN][CYAN][CY
AN][CYAN][CYAN][CYAN][CYAN][CYAN][CYAN][CYAN]
[CYAN]"
1210 FORM=1TO1500
1220 NEXTM
50000 REM***
```

Linea 1000-1010: prima puliamo ancora lo schermo, e rimettiamo il bordo ed il centro a nero con POKE BS e BC. Il motivo per cui lo facciamo ancora è perché nella fase successiva del gioco dovremo aver bisogno di ritornare a questa sezione, e così non dovremo reinserire il programma tutto dall'inizio.
Qui sono inizializzate due variabili SS e CS. Esse riguardano la costruzione del modello sullo schermo e poiché si riferiscono proprio a questa sezione del titolo, le ho lasciate qui per semplicità. SS è l'indirizzo di partenza dello schermo e CS della memoria colore.

La linea 1010 converte il 64 al modo UPPER-CASE (maiuscole) – non indispensabilmente al primo RUN, ma poiché il testo dell'Adventure è in lettere minuscole (trovo che ciò è molto più carino) abbiamo bisogno di questa linea di riconversione nel caso in cui dovrete ridare il RUN una seconda o una terza volta.

Linee 1020-1050: Questa routine sistema con una "POKE" due linee verticali all'estrema destra e sinistra dello schermo usando il simbolo grafico con il tasto Commodore ed un '+' (POKE 102) in color porpora (POKE 4). Ho usato l'istruzione POKE qui perché è possibile sistemare il simbolo nell'ultima posizione a destra senza provocare un Carriage Return, che causerebbe lo spostamento in su dello schermo di una linea.

Linee 1060-1090: Semplici istruzioni PRINT che riempiono i bordi orizzontali, superiore e inferiore, dello schermo con gli stessi simboli grafici.

Linee 1070-1180: L'immagine dell'astronave nella parte a sinistra dello schermo con tre stelle e le parole del titolo appariranno, una volta che avrete inserito queste linee.

Linee 1210-1220: infine un breve loop di tempo per mantenere sullo schermo il titolo per un po'. Questo loop è un po' corto in questa fase, in modo che quando avremo aggiunto lo sprite a questa sequenza, ciò allungherà il tempo in cui il titolo rimane sullo schermo. Naturalmente potete facilmente allungarlo (FOR M = 1 TO 3000 per esempio) se volete mantenerlo più a lungo.

Prova del modulo 6.1

Se battete RUN lo schermo dovrebbe annerirsi completamente e dopo la sequenza del titoli di testa, dovrebbe comparire una scatola attorno al bordo, seguita dalle parole "IL PIANETA DELL'INCUBO – AVVENTURA SPAZIO-TEMPORALE" nel centro dello schermo, un po' sopra la metà. Alla fine un'astronave grigia dovrebbe apparire per poi scendere nell'angolo in basso a sinistra.

Iniziano nuovi argomenti

Se guardate la **Figura 5.3** nel Capitolo 5 vedrete come ho progettato il mio schermo di apertura, nella sezione in cui disegno la trama. Questo secondo Modulo è il risultato di questo disegno.
Ho spesso avuto la sensazione che gli appassionati di Adventure hanno bisogno di un tocco di drammaticità, per sentirsi veramente dentro al mondo dell'immaginaria storia (i giochi di Scott Adams che ho visto mancano completamente di questo) quindi, dopo aver impostato il mio titolo, ho voluto una pagina di testo per spiegare quale sarebbe stato lo scenario e cosa sarebbe seguito.
Scrivere il testo per il Computer non è facile. Ho dovuto scrivere tutto direttamente, innanzittutto per aver un buon equilibrio, se mi fosse capitato di cambiare le parole sullo schermo. Mi sono alla fine deciso per uno sfondo marrone con lettere nere, per dare una sensazione di dramma. Infine ho dovuto riarrangiare il testo in modo che le parole fossero posizionate esattamente come volevo.
Il problema dei colori è uno di quelli che risolverete solamente giocando con diversi bordi e lettere colorate, provando le varie combinazioni finché non siete soddisfatti. Di nuovo credo che il tempo speso qui nell'aggiungere qualcosa di extra sia buona cosa, perché dà una sensazione globale di professionalità.

MODULO 6.2

```
1 goto 3
2 save"@0modulo6.2",8:verify"modulo6.2",
8:stop
3 rem
1270 rem**
1280 rem scrittura
1290 rem**
1300 pokebs,9:pokebc,9
1310 printchr$(147):rem Pulisce schermo
1320 form=1to200
1330 nextm
1340 poke53272,23
1350 printtab(6)"■Siete soli nella sala
di controllo dell'astronave ";
1360 print"Pensando alla ragazza che ave
```

```
te trasportato a pagamento"
1370 print"■sul pianeta Zen: la Principe
ssa che ha segretamente rubato ";
1380 print"il vostro cuore, ma che e' pr
omessa al tiranno di Zen";
1390 print" in matrimonio politico"
1400 printtab(5)"■Ma voi siete solo preo
ccupati di Aurora, non della vostra ";
1410 print"abilita' di pilota. All'impro
vviso e senza nessun segnale ";
1420 print"vi dirigete verso lo spazio R
eale troppo vicini ad un misterioso ";
1430 print"pianeta - e siete risucchiati
 dalla gravita'."
1440 printtab(5)"■Forse la vostra mente
non e' concentrata - ";
1450 print"Forse il pianeta e' troppo fo
rte. Non c'e' modo di controllare ";
1460 print"la vostra astronave. Mentre l
'ultimo ricordi e' la terra ";
1470 print"che s'avvicina velocissima...
.."
2100 printtab(11)"■Premi 's' per inizian
e"
2110 geta$:ifa$<>"s"then2110
```

Linee 1300-1340: Prima che il testo appaia è necessario cambiare il colore in marrone, pulire lo schermo (PRINTCHAR (147) è un modo alternativo a PRINT CIS) e impostare un piccolo loop. Infine la linea 1340 imposta il testo a LOWER CASE (minuscolo).

Linee 1350-1470: Banali linee di testo – introducetele con cura, con un occhio alle spaziature. I noiosi simboli grafici che appaiono sono dovuti al fatto che le frasi iniziano tutte con lettere maiuscole.

Linee 2100-2110: Avendo impostato il testo abbiamo bisogno di mantenerlo sullo schermo abbastanza a lungo perché chiunque sia in grado di leggerlo – allora che cosa c'è di meglio che avere una linea con un messaggio "BATTI S PER INIZIARE"? La linea 2110 è un metodo standard di mantenere un'istruzione PRINT sullo schermo finché l'utilizzatore vuol continuare.

Prova del modulo 6.2

Se ora date il RUN ai due Moduli assieme, troverete che lo schermo cambia ad un delizioso marrone, ed il testo sarà spaziato con regolarità sull'intera area. Non ci

dovrebbero essere parole spezzate, se avete ben controllato le spaziature sul vostro listato.
Adesso premete S: il programma si interrompe con il messaggio READY.

Consigli generali sulla scrittura del testo

Dato che molto dello spazio disponibile per un gioco di Adventure viene preso dal testo, spenderò un po' di tempo per darvi qualche suggerimento che ho sviluppato scrivendo "*Il Pianeta Dell'Incubo*".
Il problema più grosso (come avrete già scoperto) è la spaziatura corretta delle parole sullo schermo. Il modo che ho usato per farlo è stato di scrivere due o tre istruzioni PRINT allo stesso tempo, e quindi vedere come apparivano sullo schermo.
Per esempio io scriverei le Linee 1350-1370 e quindi in maniera diretta batterei GOTO 1350. Le poche linee apparirebbero, dandomi un'idea di dove ho bisogno di aggiungere o togliere spazi per rendere il testo più nitido. Dopo di ciò, aggiungendo LIST 1350-1370 (ancora in modo diretto) posso giocare con le linee mentre sono ancora evidenziate sullo schermo. Infatti se voi battete l'istruzione GOTO nella linea più alta dovete battere solo HOME per avere il cursore in alto a sinistra, e quindi battere RETURN per avere l'immediato messaggio.
Più avanti nel programma troverete delle istruzioni PRINT con GOTO e GOSUB insieme sulla stessa linea, e ho trovato più facile aggiungere un simile messaggio di errore:

es... HOME."; GOTO 3300 (notate il punto e virgola)

invece di
...HOME.": GOTO 3300

Usando il punto e virgola invece dei due punti, faccio in modo che il 64 blocchi il programma con il messaggio SINTAX ERROR, e non esegua le istruzioni GOTO e GOSUB – che era esattamente ciò che volevo quando ho scritto il testo.

Sommario

Questo Capitolo ci ha messo sulla strada del programma. È abbastanza facile finora, ma spero che siate d'accordo con me che l'atmosfera della storia può già essere recepita nella pagina del titolo e nel testo.
Naturalmente la vostra Adventure può essere semplice o complessa, a vostro piacere. Qualcuno potrebbe risparmiare sul titolo a vantaggio del programma stesso. Altri potrebbero indulgere in una grafica migliore. Ogni programmatore è diverso. La mia intenzione in questo capitolo è di introdurvi nella programmazione di testi e darvi alcune idee di come si presentano. Il resto è, come sempre, compito vostro.

CAPITOLO 7

SERVE AIUTO?

Proprio per completare alcune parti essenziali del gioco prima di muoverci tra le righe della storia attuale, mi occuperò in questo capitolo di "Istruzioni Generali" e dei moduli per fare SAVE e LOAD su nastro, e dell'aggiunta di sprite e suono nel prossimo.
La ragione per cui faccio ciò è perché questi moduli costituiranno delle subroutines di notevole valore che potrete mettere nel vostro programma man mano che proseguirete – il che è molto più veloce che il tornare indietro dopo aver programmato la storia principale, per poi continuare ad aggiungerli nei posti opportuni.
Inoltre la possibilità di salvare e di caricare vi renderà la vita molto più facile, quando inizierete la storia principale, poiché potrete salvare il gioco e poi ricaricarlo nuovamente molte volte, dopo che avrete testato delle particolari linee. Ciò risulta molto più veloce che il dover provare il gioco per ogni Locazione ogni volta che volete testare una particolare routine.

Se proprio vi occorre aiuto!

Quando ho programmato il "*Pianeta dell'Incubo*", per circa un mese ho chiesto ad alcuni amici di fare delle prove – il modo migliore per scoprire gli errori! Li ho fatti sedere di fronte al computer, ed ho osservato, foglio e penna a portata di mano per annotarmi in fretta gli errori. Ciò che più di tutto mi ha sorpreso, comunque, è stata la loro mancanza di conoscenza elementare nel condurre un gioco di Adventure. Ora, mentre voi ed io conosciamo le regole e le consuetudini di un gioco di Adventure – nessun altro è in grado di saperle. Quindi il buon programmatore fornirà una o due pagine di istruzioni per favorire il novizio nei giochi di Adventure.

MODULO 7.1

```
1 goto 3
2 save"@0modulo7.1",8:verify"modulo7.1",
8:stop
3 rem
2270 print"■■■■Volete istruzioni (s o n
):"
2280 geta$:ifa$<>"s"anda$<>"n"then2280
2290 ifa$="s"thenpokevo,0:gosub15000
14970 rem**
14980 istruzioni
```

```
14990 rem**
15000 Print"■":Pokebs,11:Pokebc,11:Print
tab(13)"■■■Istruzioni■■■
15010 Print"■■■■Entrate nel mondo della
 fantasia e avventura";
15020 Print" col vostro computer come gu
ida. Per superare";
15030 Print" questa avventura dovete dar
e i comandi - in italiano - :"
15040 Print"sempre un ■verbo■ e qualche
volta un nome.
15050 Print"Per es: - ■prendi mitra■ o ■
scala albero
15060 Print"■■■Vi verra' data una descri
zione del luogo in cui siete ";
15070 Print"e delle direzioni in cui pot
ete andare. Potete usare solo"
15080 Print" vai ■nord■, ■sud■, ■est■ o
ovest.
15090 Print"■■A volte potete incontrare
un oggetto che potreste usare ";
15100 Print"piu' tardi durante il gioco;
 potete raccoglierlo con ";
15110 Print"la parola ■'prendi'■ o ■'aff
erra'. Potete usarlo per ";
15120 Print"■'prendi'■ la vostra tuta o
'portare' la principessa indietro.
15130 Printtab(8)"■■■Premi un tasto per
cominciare
15140 geta$:ifa$=""then15140
15150 Print"■■■■Potete anche ■'butta'■ o
 'togli' un oggetto in ogni momento ";
15160 Print"ed esso rimarra' nel luogo f
inche' non lo riprenderete ";
15170 Print"quando ne avrete bisogno anc
ora.
15180 Print"■■■Tasti speciali risparmiera
nno tempo: per es:-
15190 Print"■■■■n■ ■e■ ■s■ ■o ■o■ ■sig
nificano ";
15200 Print"■nord■, ■est■, ■sud■ o ■oves
t.
```

```
15210 print"▶▶▶▶i▶ ▶▶vi dara´ un ▶▶inventa
rio  degli oggetti che raccogliete.
15220 print"▶▶▶▶l▶ significa ▶guarda▶▶ in
torno."
15230 print"▶▶▶▶h▶ ▶▶significa ▶▶che volet
e aiuto!!
15240 print"▶▶▶▶battete ▶istruzioni▶▶ Per
 avere informazioni generali.
15250 print"▶▶Se volete fare salvare il g
ioco battete salva
15260 print"▶▶Se volete fare finire il gi
oco battete esci
15270 print"▶▶o ▶fine▶▶."
15280 printtab(8)"▶▶▶Premi un tasto
15290 geta$:ifa$=""then 15290
15300 print"▶▶▶Usate solo lettere senza ▶
shitf:
15310 print"in altre parole battete ▶▶▶a▶
urora non ▶A▶urora.
15320 print"(notate la ▶a▶▶ ,non ▶A▶▶).
15330 print"▶▶Suggerimenti:":print"▶▶1. Fe
rmatevi spesso per fare salva"
15340 print"2. Tracciate una ▶mappa▶▶ men
tre esplorate il pianeta.
15350 print"3. Siate precisi nell´ ▶inve
ntario potendo prendere solo";
15360 print" pochi oggetti alla volta.
15370 print"4. Se volete aiuto battete ▶
esamina per vedere gli oggetti.
15380 print"5.Ricordate di battere ▶istr
uzioni se rivolete questi messaggi."
15400 print"▶▶▶▶▶Buona fortuna per la ric
erca di Aurora";
15410 print"e fate attenzione agli alien
i!!
15420 print"▶▶▶▶▶▶▶▶Batti ´L´ per tornar
e al gioco
15430 geti$:ifi$<>"l"then15430
15440 ifi$="l"thenprint"▶":return
```

Linee 2270-2290: Non appena le pagine di titolo sono terminate ed il giocatore ha premuto S per iniziare il gioco, seguiranno due ulteriori suggerimenti prima di entrare nel merito del gioco. Il primo di questi concerne il caricamento da nastro ed il secondo è mostrato sullo schermo dalle Linee 2270-2290. In poche parole si chiede al giocatore se desidera avere "istruzioni". La Linea 2290 è resa operativa se la risposta è S, mandando il programma alla subroutine a 15000. (La istruzione POKE VO,0 in questa linea sopprime il suono come riferirò nel Capitolo 8). Se la risposta è N il programma passa automaticamente alla sessione successiva.

Linee 14970-15410: Questa è l'intera subroutine. Sullo schermo appaiono tre pagine separate di istruzioni, ciascuna di esse scorrerà premendo un tasto qualsiasi del 64 (Linee 15140 e 15280). Non c'è molto da dire qui. L'unico problema che potete avere è quello di interpretare i vari colori ed i tasti reverse on/off, dato che ho illuminato parole e frasi cambiando il colore in tutto il testo. Ciò inoltre mette in risalto l'aspetto delle pagine, conferendo loro una piacevole presentazione e composizione.

I colori base che ho usato sono il verde per il testo principale, il blu (il tasto Commodore più il tasto 7, non il tasto Control più 7) per le parole chiave, bianco per le linee "PREMI IL TASTO" e nero per l'intestazione "ISTRUZIONI".

Una volta ancora: quando battete il testo scrivete solo un paio di linee per volta, poi battete GOTO al numero di linea per controllare la spaziatura.

Linee 15420-15440: Notate che questa volta ho dichiarato "PREMI L PER RITORNARE AL GIOCO" invece di semplicemente PREMI UN TASTO. La ragione di ciò sta nel fatto che se il giocatore è nel mezzo della trama e chiama questa subroutine quando fa RETURN al gioco, ha bisogno di avere il programma evidenziato nella posizione in corso, per ricordargli a che punto è.

Più tardi programmeremo "G" o "GUARDA" che sarà il corretto suggerimento per permettere alla Locazione corrente di essere evidenziata. Perciò ci occorre lo stesso suggerimento qui.

(Se non capite ciò ora – non preoccupatevi – Vi sarà più chiaro in seguito quando programmerete i comandi I).

prova del Modulo 7.1

Se ora fate il RUN del programma, dovreste trovare che dopo l'introduzione alla storia su uno schermo marrone – lo schermo diventa grigio non appena premete S, e appare quindi la prima pagina di istruzioni. Se non premete S per iniziare, ma un qualsiasi altro tasto, allora lo schermo rimarrà marrone. (Provate per controllare – non prendete le mie parole alla lettera!) Dovreste riuscire a esaminare tre pagine di istruzioni prima che il programma finisca. Se fate scorrere queste tre pagine e poi premete un altro tasto esattamente alla fine, il programma tornerà all'inizio delle tre pagine e le ripercorrerà di nuovo. Se continuate a premere un tasto alla seconda esecuzione, il programma abortirà con un messaggio di errore RETURN WITHOUT GOSUB – che potrebbe confondervi. La spiegazione è che alla prima esecuzione il programma disporrà il RETURN nella Linea 15440 e tornerà alla 14970 perché è da qui che è partita. Alla seconda esecuzione, poiché non ha avuto il comando GOSUB nella Linea 2290, abortisce.

Caricamento e salvataggio

Non c'è dubbio che i giochi di Adventure devono avere la possibilità di fare il SAVE di una sessione su nastro. A parte il fatto dell'uso ovvio di immagazzinare il gioco se non avete più tempo, è anche comodo salvare il gioco quando raggiungete un punto critico dove potreste soccombere – e fare SAVE a questo punto vi metterà in grado di ricominciare senza dover ripercorrere tutta la Adventure.
Il modulo seguente vi permetterà di fare sia il SAVE che il LOAD.

MODULO 7.2

```
1 goto 3
2 save"@0modulo7.2",8:verify"modulo7.2",
8:stop
3 rem
2240 print"■■■■■Volete caricare un gioc
o da nastro ? (s o n):"
2250 geta$:ifa$<>"s"anda$<>"n"then2250
2260 ifa$="s"thenpokevo,0:gosub16000
15970 rem**
15980 rem carica
15990 rem**
16000 print"■■■Inserisci il nastro e ria
vvolgi:
16010 input"Batti ■RETURN■:";g$
16020 open1,1,0,"planet"
16030 input#1,pr,cv,sv,ev,es,gv,dv,nv,fv
16040 input#1,z,n$,m,i,p,n,e,s,w,i$,pi,s
p
16050 input#1,ve$,no$,g,p2,np,na,rc,yy,h
h,hr
16060 fori=1to9:input#1,ob%(i):nexti
16070 fori=1to9:input#1,ob$(i):nexti
16080 fori=1to9:input#1,si$(i):nexti
16090 close1
16100 gosub35420
16110 print"■":pokebs,11:pokebc,11:goto4
020
16970 rem**
16980 rem salva
16990 rem**
17000 print"■■■Siete sicuri di voler sa
lvare":printtab(15)"il vostro gioco?
17010 print"■Premete ■s■ o ■n■:"
```

```
17020 geta$:ifa$<>"s"anda$<>"n"then17020
17030 ifa$="n"thenprint"■":goto4020
17040 input"■Metti il nastro e premi ■RE
TURN■:"q$:r$=chr$(13)
17050 open1,1,1,"Planet"
17060 print#1,pr;r$;cv;r$;sv;r$;ev;r$;es
;r$;qv;r$;dv;r$;nv;r$;fv
17070 print#1,z;r$;n$;r$;m;r$;i;r$;p;r$;
n;r$;e;r$;s;r$;w;r$;i$;r$;pi;r$;sp
17080 print#1,ve$;r$;no$;r$;g;r$;p2;r$;n
p;r$;na;r$;rc;r$;yy;r$;hh;r$;hr
17090 fori=1to9:print#1,ob%(i):nexti
17100 fori=1to9:print#1,ob$(i):nexti
17110 fori=1to9:print#1,si$(i):nexti
17120 close1
17130 gosub35420
17140 print"■":goto4020
```

Linee 2240-2260: Una pura e semplice routine che permette al giocatore di andare alla subroutine di caricamento.

Linee 1600-1610: Se premete SAVE per attivare questa sobroutine, allora lo schermo sparirà e apparirà l'istruzione "METTI IL NASTRO NEL REGISTRATORE E RIAVVOLGI:" seguito da "PREMI RETURN". Queste due linee fanno ciò. Lo strano geroglifico nella Linea 1610 è un RETURN in modo shiftato.

Linea 16020: Questa Linea fa una OPEN di un file su una cassetta (il che significa che permetterà ai dati che avete creato nel programma di essere registrati su una cassetta). La terza cifra (0) in questa linea è chiamata "indirizzo secondario" e tutto ciò che fa è di dire alla cassetta di "scrivere" i dati. Infine daremo al file il nome di "PIANETA".
Quando ho iniziato a giocare con le Adventure ho pensato che la cassetta avrebbe immagazzinato il mio gioco in modo sequenziale, come se il computer avesse tenuto una registrazione del mio viaggio, e avesse quindi scritto tutte queste informazioni sulla cassetta come parte integrante della storia stessa. Il che significa che se avessi passato due ore a giocare, avrebbe dovuto immagazzinare il doppio di informazioni che se avessi giocato un'ora.
Ciò che effettivamente succede è molto più semplice – e molto più ovvio. Tutto ciò che il 64 deve registrare è il valore di tutte le variabili nel programma, quindi ricaricando questi dati sembrerà che si riprenda dalla stessa posizione che si era lasciata. In tal modo avrete registrato un "file" di dati fatti con i valori di tutte le variabili di cui discuteremo più avanti nel libro.

Linee 16030-16050: Queste tre linee memorizzano semplicemente tutte le variabili che cambiano mentre state adoperando il nastro con il comando INPUT . È molto importante mantenerle nello stesso ordine del mio, poiché occorre fare il SAVE ed il READ dei dati nello stesso esatto ordine, altrimenti il 64 abortirà il processo.

Linee 16060-16090: Infine scriveremo tutti i dati dai vettori relativi agli oggetti (come nel programma "Gioco d'Adventure" della prima sezione del libro). L'ultima linea fa una CLOSE del file.

Linea 16100: Questa linea si riferisce agli sprite che tratterò nel prossimo Capitolo, ma poiché il principio di come stabilite un caricamento e salvate una routine è importante, vi darò dei cenni qui. Quando caricate dati per sprite, viene utilizzato il buffer della cassetta (in altre parole il pezzo di memoria che il 64 usa come store temporaneo, mentre la cassetta registra o fa un caricamento di dati). In tal modo dopo aver eseguito una routine di SAVE o di LOAD, possiamo ricreare i dati dello sprite nella subroutine 35420 prima di tornare al programma principale.

Linea 16110: Se il giocatore ha caricato un programma per un gioco precedente, gli occorrerà caricare il gioco in modo che lo schermo mostri la posizione corrente (che non sarà quella iniziale). Per questa ragione alla subroutine servirà un GOTO e non un RETURN o qualcos'altro, il che la rimanderebbe all'inizio della storia. Questa Linea fornisce la procedura corretta, anche se a questo punto del programma non avrà effetto, poiché non avete immesso ancora la Linea 4020. Essa imposta inoltre il corretto colore dello schermo.

Linee 17000-17040: Prima di permettere alla subroutine di SAVE di cominciare, c'è un messaggio di sicurezza "SEI SICURO" nella Linea 17050, PRINT# è usato invece di INPUT#, e (questo è molto importante) R$ (che è uguale ad un ritorno carrello ed è impostato nella Linea 17040) è messo tra ogni variabile. La ragione di ciò è che il 64 ha scarsa specializzazione al salvataggio di dati e necessita di queste suddivisioni tra ogni variabile.
Notate che l'ordine delle variabili è lo stesso delle Linee 16030-16080.

Linee 17130-17140: Infine cancelliamo il buffer della cassetta e ridefiniamo i dati dello sprite come prima, cancelliamo lo schermo, e poi facciamo un GOTO dell'area del programma che abbiamo deciso di salvare. Ciò è necessario nel caso vogliate fare il SAVE del programma e poi continuare a giocare.

Prova del Modulo 7.2.

Naturalmente a questo punto del programma non c'è niente da caricare e niente da salvare – ma se fate RUN troverete che dopo aver impostato il messaggio "PREMI S PER INIZIARE" otterrete un altro suggerimento – "VOLETE CARICARE UN GIOCO ESISTENTE DAL NASTRO (S o N)". Premendo N dovreste ora ottenere il messaggio "ISTRUZIONI" del Modulo 7.1, mentre invece premendo S entrerete nella routine di caricamento.
Lo schermo diventerà nero, e vi sarà richiesto di caricare un nastro nella cassetta e premere RETURN. Facendo ciò (naturalmente non dovrete avere noie col nastro) dovreste ottenere una risposta PRESS PLAY ON TAPE. L'ordine "Run/Stop" vi porterà fuori dal programma. Se ciò non succede, vuol dire che c'è qualcosa di sbagliato edio suggerisco di controllare attentamente di aver battuto i numeri giusti dopo il comando OPEN nelle Linee 16020 e 17050. Quando farete effettivamente LOAD e SAVE vedrete che lo schermo lampeggerà e le parole che avete battuto appariranno per un breve momento. Questa è una idiosincrasia del 64 ed è meglio ignorarla. Comunque, non preoccupatevi.

Sommario

L'accuratezza e l'offerta adeguata di istruzioni è un elemento importante per ogni software – spesso carente anche nei packages più cari in circolazione. La difficoltà sta nello scrivere abbastanza senza occupare troppa memoria. Nel "*Pianeta dell'Incubo*" ho fornito il massimo, in seguito alle reazioni delle persone che non conoscevano nulla di Adventure – un eccellente pubblico per il vostro gioco. Potreste forse preferire di essere più economi, o almeno di omettere quest'area, nel caso che il gioco venga usato unicamente da voi e qualche altro vostro amico.
Comunque, ritornando al gioco dopo qualche anno, troverete che non potete ricordare così facilmente fatti che sembravano senza ombra di dubbio ovvi a quel tempo: la memoria è labile. Un tipo qualsiasi di guida è in queste circostanze di inestimabile valore.
Il prossimo passo è quello di fornire il suono e grafici – per aggiungere il tocco finale prima di iniziare la storia vera e propria.

CAPITOLO 8

SPRITE ED EFFETTI SONORI

Sebbene abbia affermato che preferisco giocare Adventure basate solo su testi, era abbastanza ovvio che se avessi aggiunto grafici ed effetti sonori si sarebbe ottenuto un miglioramento in finezza. Ovviamente non c'è abbastanza memoria nel 64 per permettermi di creare un'intera Adventure con disegni e colonne sonore, ma un po' di programmazione può portare molto lontano grazie alla disponibilità di due facility del 64:

1 Sprite
– 2 Un chip sonoro con tre 'voci'.

Non c'è spazio in un libro di queste dimensioni per indugiare a lungo su sprite e programmazione del suono, perciò ho voluto descrivere come ho ottenuto gli effetti che ho usato nel '*Pianeta dell'Incubo*', e credo che anche voi farete ciò che ho fatto io – leggetevi come utilizzare queste facilitazioni nel Reference Guide, se non avete familiarità con essi.

Creazione dello sprite

Che cos'è esattamente uno sprite? Io lo intendo come un gruppo di "pixel" programmabili, piuttosto che un maxi-carattere definibile dall'utente. In altre parole usando poche istruzioni DATA potete creare un oggetto di alta risoluzione che si muoverà sullo schermo (senza tutte le ammucchiate di POKE e PEEK che di solito accompagnano i grafici, e senza far uso di POKE di spazi bianchi per muoverli).
Il Reference Guide del Commodore riassume tutto ciò quando dice... (e cito tra virgolette):

"tutto quello che dovete fare è dire 'a cosa deve somigliare' 'di che colore deve essere' e 'come deve apparire' uno sprite. Il chip del Vic-II farà il resto!".

Per creare uno sprite è necessario per prima cosa sapere a cosa deve somigliare. Ci sono diversi modi di arrivare a ciò ma il più facile è di disegnarlo su una griglia di 21 x 24 come nell'User Manual. Nella **Figura 8.1** ho preso una copia di questa griglia e l'ho riportata quattro volte per aver modo di esercitarmi con disegni differenti.
Diamo un'occhiata allo sviluppo di uno dei miei sprite, Victor, dal concepimento al completamento.
Se vi ricordate, desideravo che un Personal Robot (che nasceva con il nome di Proteus ma ad un certo punto del cammino è diventato Victor) apparisse qualche volta nella storia con un messaggio per il giocatore. Uno sprite sembrava la soluzione ideale – ma dovevo disegnarlo. La **Figura 8.2** mostra i vari disegni partendo da un paio di schizzi grossolani che ho tentato in seguito di adattare alla mia griglia 24 x 21. Ciò risultò essere più arduo di quanto pensassi inizialmente – poiché avevo la tendenza a provare con curve e cose del genere – ma al terzo tentativo questi cominciò a prendere forma.

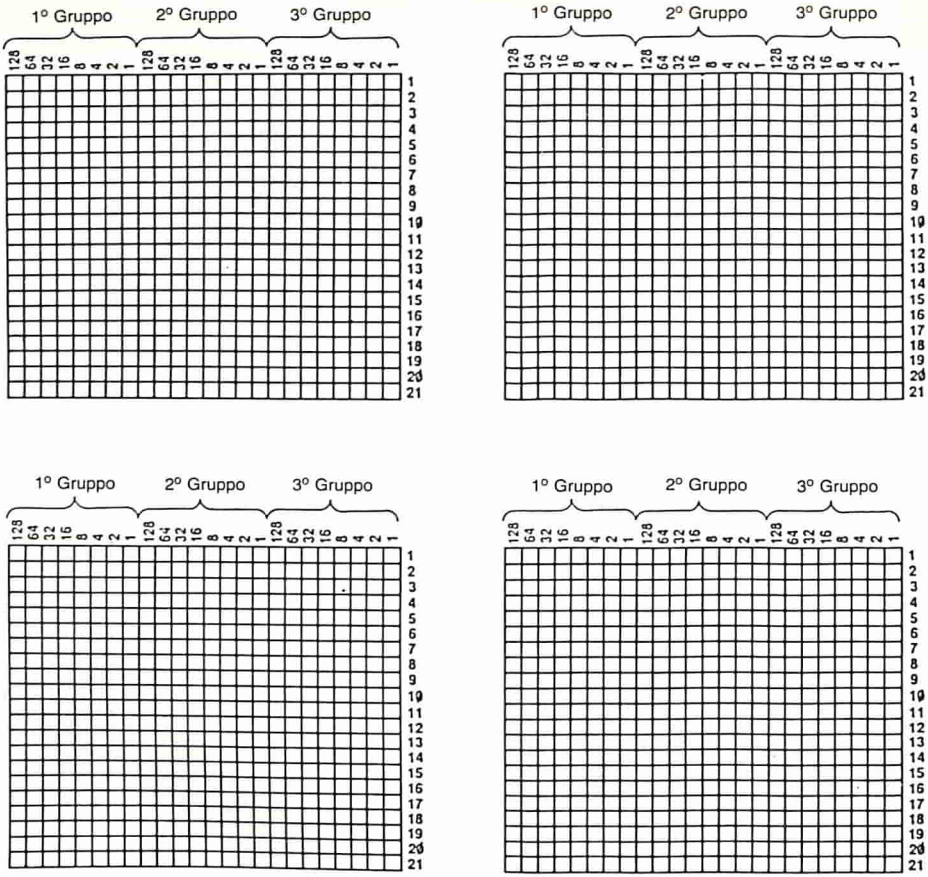
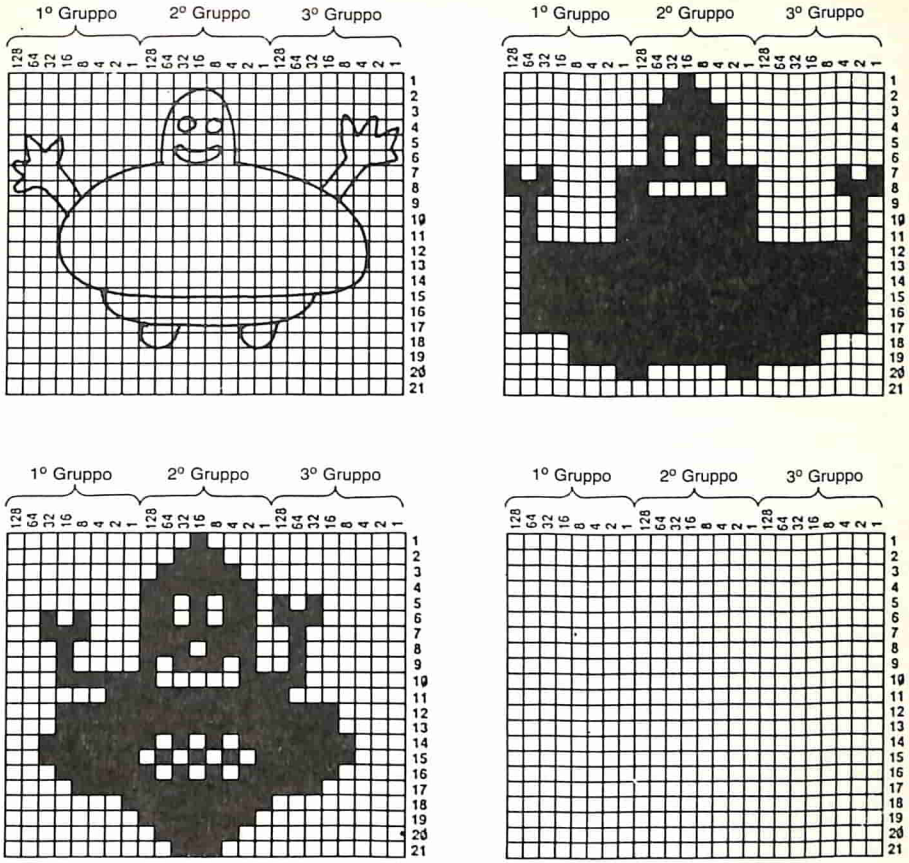


Figura 8.1 Griglia per progettare sprites

La tappa successiva è di convertire la vostra griglia in istruzioni DATA. Potete fare ciò manualmente (questo spesso seccante compito è adeguatamente descritto nel Reference Manual) ma ho pensato che sarebbe stato meglio trovare un programma che facesse ciò per me, ed ho finito per usare un programma di Sprite Editor scritto da A.R. Bennet che ho trovato in una recente copia del *'Personal Computer World'*. Potete anche comprare degli Sprite Editors già pronti (anche il Simon's BASIC ne contiene uno) ma assicuratevi di prenderne uno che vi permetta di stampare i vostri dati una volta che avete disegnato lo sprite sullo schermo.
Sono quasi certo che tutti funzionino con gli stessi principi, permettendovi sia di tener fermi i vostri pixel sullo schermo, sia di cancellarli, così potete manipolarli fino a che ne avete voglia. Lo Sprite Editor che ho usato io aveva pure la possibilità di espandere lo sprite orizzontalmente e verticalmente prima di salvare i dati, così potevo realmente vedere che aspetto avrebbe assunto.



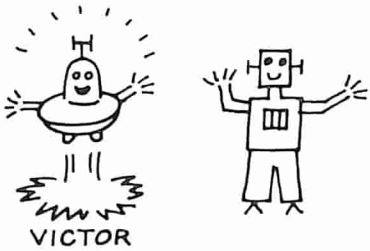


Figura 8.2 Progetto di Victor

Il movimento dello Sprite.

Dopo aver disegnato Victor ed aver determinato i dati per crearlo sullo schermo, ho deciso di eseguire un piccolo programma di test per provarne il movimento prima di inserire tutte le linee nel programma d'Adventure. Ciò ha il vantaggio di vedere se i colori dello sfondo, i colori dello sprite e la grandezza necessitano di correzioni prima di inserire l'intero programma, e dà anche un'idea della manipolazione dello sprite.
Forse il più grosso problema che ho incontrato con gli sprite è stato la loro capacità di attrazione. Dopo aver trovato che potevo creare una figura con relativa facilità e poi iniziare a muoverla su tutto lo spazio (con effetti tridimensionali, collisioni, ecc.) mi è passata rapidamente per la testa una grande quantità di idee, ed è stato straordinariamente duro soffocare la mia immaginazione, e concentrarmi con tenacia solo su un'immagine – l'utile robot servitore che appare nei momenti di tensione, oppure di comicità, lungo tutto il programma. Il 64 rende certamente la vita facile, e giunge a effetti di alto livello professionale.
La prima cosa che vi occorre fare è 'accendere lo sprite' o 'abilitarlo', altrimenti sebbene lo sprite possa essere in memoria voi non potete vederlo. Il **Programma 8.1** è un semplice esempio.

Programma 8.1. Spriteman

```
1 goto3
2 save"@0:spriteman",8:verify"spriteman"
,8:stop
3 rem
7 rem*******************
8 rem inizializzazione
9 rem*******************
10 vc=53248:rem locazione video chip
15 print"⌂"
17 rem*******************
18 rem creazione di sprite
19 rem*******************
20 pokevc+21,128:rem livello minimo di p
riorita' (da 0 (max) a 128 (min))
30 poke2047,13:rem locazione corretta pe
r sprite
40 forn=0to62:readd:poke832+n,d:next:rem
 s@rite creato ora in memoria
45 pokevc+23,128:pokevc+29,128:rem x & y
 di grandezza doppia
47 rem*******************
48 rem movimento dello sprite
```

```
49 rem*********************
50 Pokevc+15,100:rem coordinata 'y'
57 rem*********************
58 rem movimento su x in 2 fasi
59 rem*********************
60 forx=0to255:Pokevc+14,x:forj=1to25:ne
xt:next
70 Pokevc+16,128:forx=0to64:pokevc+14,x:
forj=1to25:next:next
77 rem*********************
79 rem*********************
80 Pokevc+16,0:Pokevc+21,0
97 rem*********************
98 rem data
99 rem*********************
100 data 15,127,240,0,127,0,0,129,0,255,
255
110 data 255,255,255,255,192,231,3,192,2
31,3,255
120 data 231,255,255,231,255,255,231,255
,255,231,255
130 data 192,0,3,192,0,3,192,0,3,255,255
140 data 255,255,255,255,1,128,192,1,128
,192,1
150 data 128,192,1,128,192,7,128,240
```

Questo programma crea un omino (che chiameremo Victor) con le istruzioni DATA nelle Linee 100-150 e poi lo abilita (Linea 20) e prepara un blocco di memoria (Linea 30). La Linea 40 legge i DATA e li mette in memoria (ricordate che questa è il buffer che va sulla cassetta) e la Linea 45 raddoppia lo sprite in grandezza sia verticalmente che orizzontalmente.
La linea 50 mette il nostro sprite in una posizione in basso nello schermo e poi le Linee 60 e 70 lo muovono attraverso lo schermo. Infine la Linea 80 lo disabilita di nuovo.
Sono stato breve all'estremo in questa spiegazione, perché essa è sostanzialmente un doppione di quello che potete leggere sul Reference Manual o sull'User Manual, se lo desiderate.

Victor e il mealy bug

Ho detto nel Capitolo 6 che volevo uno sprite da muovere attraverso il titolo "*Il Pianeta dell'Incubo*", sul genere di Victor (che sarebbe apparso in diversi momenti durante il programma). Avendo disegnato Victor sulla mia griglia, ho lavorato con il mio Sprite

Editor fino a che sono giunto ad un piccolo diabolico oggetto, simile ad un insetto che sembrava ideale per scorrere tra i titoli, e l'ho chiamato Mealy Bug. È giunto il tempo di programmare questi due sprite nel nostro Adventure.

MODULO 8.1

```
1 goto3
2 save"@0modulo8.1",8:verify"modulo8.1",
8:stop
3 rem
3500 pokev+39,2:pokev+27,1:pokev+23,1:po
kev+29,1
34970 rem**
34980 rem  poke puntatori
34990 rem**
35000 pokev+39,2:pokev+27,1:pokev+23,1:p
okev+29,1
35010 pokev+1,100:pokev,0:pokev+21,1
35020 forx=0to255:pokev,x:nextx:pokev+16
,1
35030 forx=0to95:pokev,x:nextx:pokev+16,
0:pokev+21,0
35040 return
35070 rem**
35080 rem victor
35090 rem**
35140 pokev+40,4:pokev+29,2:pokev+23,2
35150 pokev+2,80:pokev+3,0:pokev+21,2
35160 fory=0to190:pokev+3,y:nexty
35180 return
35200 data 99999
35210 rem**
35220 rem data sprite 1
35230 rem**
35240 data 255,128,0,255,128,0,1,128,0,1
,128
35250 data 0,3,252,0,31,255,224,255,252,
32,63
35260 data 252,63,63,255,255,224,0,1,63,
255,240
35270 data 63,255,240,3,1,0,3,25,0,3,31
35280 data 0,227,0,0,255,0,0,0,0,0,0
35290 data 0,0,0,0,0,0,0,0,0
```

```
35300 rem**
35310 rem data sprite 2
35320 rem**
35330 data 0,112,0,0,248,0,1,252,0,1,172
35340 data 0,1,172,0,3,254,0,3,222,20,1
35350 data 116,20,1,4,28,1,252,8,0,112,8
35360 data 7,255,8,7,255,8,127,255,248,6
8,1
35370 data 0,229,253,0,265,253,0,4,1,0,7
35380 data 255,0,7,255,0,1,36,0
35390 rem**
35400 setta 9li sprite
35410 rem**
35420 restore:forll=1to1000:reada$:ifa$=
"99999"thenll=1016
35430 nextll
35440 Poke2041,14:Poke2040,13
35450 forll=0to126:read9:Poke832+ll,9:ne
xtll
35460 restore
35470 return
```

Linee 35000-35040: La variabile V, lungo tutta la programmazione dello sprite, si riferisce all'indirizzo iniziale del Video Chip (53248) ed era definita nella Linea 280 nel Modulo 6.1. In questo modo queste Linee 'abilitano' lo sprite, definiscono il suo colore e grandezza e lo muovono attraverso lo schermo da sinistra finché non scompare. È questa subroutine che mantiene sullo schermo un po' più a lungo la pagina d'inizio riportante il titolo.

Linee 35140-35180: Queste Linee fanno esattamente la stessa cosa per Victor, tranne che lui si muove in una direzione verticale anziché orizzontale, così che sembra cadere dall'alto nello schermo e si ferma a circa due terzi della strada, in basso sul lato sinistro.

Linea 35200: Questa Linea crea la maschera DATA di "99999" per permettere al programma di distinguere tra i data sprite e i data che programmeremo più tardi, riguardanti gli oggetti che saranno sparsi attorno nell'Adventure.

Linee 35240-35380: I DATA per i due sprite.

Linee 35420-35470; La sezione è la subroutine iniziale per settare gli sprite. Per prima cosa il 64 provoca un RESTORE, per iniziare a leggere i DATA daccapo. Poi verifichiamo che la maschera 99999 (nella Linea 35200, ricordate) impedisca alla stringa di data nell'altra parte del programma dall'essere letta per sbaglio, e che compaia quindi un messaggio di syntax error. Infine le istruzioni DATA e gli sprite vengono letti e messi nel buffer di memoria della cassetta. Questa è la subroutine che causa un apparente blocco del computer proprio all'inizio del programma ed anche dopo il LOAD ed il SAVE.

Prova del Modulo 8.1.

Se cercate di testare questo modulo sembrerà che non succeda niente, perché, sebbene noi abbiamo creato i nostri sprite, non c'è niente nel programma per chiamarli. Il prossimo modulo correggerà questa mancanza.

L'introduzione di Victor

MODULO 8.2

```
1 goto3
2 save"@0:m8.2",8:verify"m8.2",8:stop
3 rem
410 gosub35420
1200 gosub35000
2370 rem**
2380 rem victor
2390 rem**
2400 pokebs,11:pokebc,11:print"■":print"
Personal Robot Victor informa:"
2410 print"■Durante questa avventura sar
o' pronto ad aiutarvi..."
2420 print"■Di solito apparirò nei mome
nti di grande tensione e pericolo!"
2430 print"Ma vi sarò sempre appresso,
anche se non mi vedete."
2440 input "■■battete il vostro nome";n$
2450 gosub18000:print"■■grazie ";n$;" Vi
terrò d'occhio !"
2460 form=1to1000:nextm
2470 print"■■■■Inizia l'Avventura■...."
2480 form=1to2000:nextm
2490 pokev+21,0
```

Linea 410: Questa è la GOSUB iniziale che setta gli sprite proprio all'inizio nel programma. È lo stesso metodo della GOBUS delle Linee 16100 e 17130 nel modulo 7.2.

Linea 1200: Con questa Linea facciamo apparire il Mealy Bug nel titolo principale.

Linea 3200: Questa Linea interrompe prima il suono (poiché Victor può essere chiamato in ogni momento del programma, volevo essere sicuro che qualsiasi suono fosse

Prova del Modulo 8.1.

Se cercate di testare questo modulo sembrerà che non succeda niente, perché, sebbene noi abbiamo creato i nostri sprite, non c'è niente nel programma per chiamarli. Il prossimo modulo correggerà questa mancanza.

L'introduzione di Victor

MODULO 8.2

```
1 goto3
2 save"@0:m8.2",8:verify"m8.2",8:stop
3 rem
410 gosub35420
1200 gosub35000
2370 rem**
2380 rem victor
2390 rem**
2400 pokebs,11:pokebc,11:print"■":print"
Personal Robot Victor informa:"
2410 print"■Durante questa avventura sar
o' pronto ad aiutarvi..."
2420 print"■Di solito apparirò nei mome
nti di grande tensione e pericolo!"
2430 print"Ma vi sarò sempre appresso,
anche se non mi vedete."
2440 input "■■battete il vostro nome";n$
2450 gosub18000:print"■■grazie ";n$;" Vi
terrò d'occhio !"
2460 form=1to1000:nextm
2470 print"■■■■Inizia l'Avventura■...."
2480 form=1to2000:nextm
2490 pokev+21,0
```

Linea 410: Questa è la GOSUB iniziale che setta gli sprite proprio all'inizio nel programma. È lo stesso metodo della GOBUS delle Linee 16100 e 17130 nel modulo 7.2.

Linea 1200: Con questa Linea facciamo apparire il Mealy Bug nel titolo principale.

Linea 3200: Questa Linea interrompe prima il suono (poiché Victor può essere chiamato in ogni momento del programma, volevo essere sicuro che qualsiasi suono fosse

disabilitato prima di chiamarlo) con la istruzione POKE VO, 0. Poi la GOSUB 35100 chiamerà Victor.

Linee 2400-2440: Appena Victor entra in scena, questo messaggio appare con lui. Seguendo questo messaggio vi si chiede di immettere il vostro nome (essendo presente la variabile N$).

Linee 2450-2490: Il primo GOSUB nella Linea 2450 verrà onorato dal Modulo 8.4 (che introdurremo tra poco) e si riferisce ad una routine di suono. Ho provato ad aggiungere un beep o un burp (un indecente ruttino!) ogni volta che il 64 risponde all'input. Il beep è per una risposta okay e il burp è per una risposta non okay (ovviamente!).

Linea 2460: È un breve loop di pausa, seguito dalla pulizia delo schermo e dalle parole "INIZIA L'AVVENTURA...", che appariranno in alto sullo schermo. c'è un'altra breve pausa (Linea 2480) poi Victor sparisce (Linea 2490 che lo spegne).

Prova del modulo 8.2

Il titolo comincia finalmente a prendere forma. Se lo fate eseguire ora, ci sarà una pausa significativa mentre la sequenza dei titoli di testa è sullo schermo (mentre gli sprite vengono settati), quindi il Mealy Bug viaggerà tranquillamente attraverso lo schermo oltre il bordo rosso e le parole "IL PIANETA DELL'INCUBO".

Comunque, se proseguite il programma non arriverete ancora a Victor, ma al messaggio UNDEF'D STATEMENT ERROR IN 2300. Ciò accade perché la subroutine di Victor richiede ora il suono e la Linea 3500 non esiste ancora.

Adesso ascoltate...

L'aggiunta del suono al programma è come guardare un film a colori in bianco e nero e poi comperare la televisione a colori – in altre parole fantastico. Naturalmente dopo un po' il suono comincia ad essere considerato scontato, ma dopo aver battuto i moduli seguenti, vi prometto che noterete una grandissima differenza tra il programma prima e dopo questa sessione.

Comincerò affermando che non ho intenzione di imbarcarmi in una spiegazione sul suono e sul chip sonoro. Di nuovo, sia il vostro Manual che il Reference Guide contengono una quantità di informazioni su questo argomento – sicché mi limiterò a presentare il modo in cui ho ottenuto gli effetti che ho usato.
Non ho sperimentato nessuna musica vera, in parte per la mia personale incapacità musicale ed in parte per mancanza di tempo. Se siete musicofili potete inventare un breve motivo introduttivo che sarà anche meglio della mia traccia-sonora.

Ho cominciato a giocare col suono usando alcuni programmi di esempio nell'User Manual, ma sono arrivato ben presto alla conclusione che avevo bisogno di un semplice programma tutto mio, che mi permettesse di sostituire i valori differenti per le varie forme d'onda, frequenze, ecc. di tutte e tre le voci del 64. Così è stato necessario convertire i vari valori delle differenti variabili del chip del suono in numeri semplici, ed ho costruito la mia tavola delle variabili, di carattere piuttosto mnemonico (vedi **Tavola 8.1**). Riportata a pagina seguente.

disabilitato prima di chiamarlo) con la istruzione POKE VO, 0. Poi la GOSUB 35100 chiamerà Victor.

Linee 2400-2440: Appena Victor entra in scena, questo messaggio appare con lui. Seguendo questo messaggio vi si chiede di immettere il vostro nome (essendo presente la variabile N$).

Linee 2450-2490: Il primo GOSUB nella Linea 2450 verrà onorato dal Modulo 8.4 (che introdurremo tra poco) e si riferisce ad una routine di suono. Ho provato ad aggiungere un beep o un burp (un indecente ruttino!) ogni volta che il 64 risponde all'input. Il beep è per una risposta okay e il burp è per una risposta non okay (ovviamente!).

Linea 2460: È un breve loop di pausa, seguito dalla pulizia delo schermo e dalle parole "INIZIA L'AVVENTURA...", che appariranno in alto sullo schermo. c'è un'altra breve pausa (Linea 2480) poi Victor sparisce (Linea 2490 che lo spegne).

Prova del modulo 8.2

Il titolo comincia finalmente a prendere forma. Se lo fate eseguire ora, ci sarà una pausa significativa mentre la sequenza dei titoli di testa è sullo schermo (mentre gli sprite vengono settati), quindi il Mealy Bug viaggerà tranquillamente attraverso lo schermo oltre il bordo rosso e le parole "IL PIANETA DELL'INCUBO".

Comunque, se proseguite il programma non arriverete ancora a Victor, ma al messaggio UNDEF'D STATEMENT ERROR IN 2300. Ciò accade perché la subroutine di Victor richiede ora il suono e la Linea 3500 non esiste ancora.

Adesso ascoltate...

L'aggiunta del suono al programma è come guardare un film a colori in bianco e nero e poi comperare la televisione a colori – in altre parole fantastico. Naturalmente dopo un po' il suono comincia ad essere considerato scontato, ma dopo aver battuto i moduli seguenti, vi prometto che noterete una grandissima differenza tra il programma prima e dopo questa sessione.

Comincerò affermando che non ho intenzione di imbarcarmi in una spiegazione sul suono e sul chip sonoro. Di nuovo, sia il vostro Manual che il Reference Guide contengono una quantità di informazioni su questo argomento – sicché mi limiterò a presentare il modo in cui ho ottenuto gli effetti che ho usato.
Non ho sperimentato nessuna musica vera, in parte per la mia personale incapacità musicale ed in parte per mancanza di tempo. Se siete musicofili potete inventare un breve motivo introduttivo che sarà anche meglio della mia traccia-sonora.

Ho cominciato a giocare col suono usando alcuni programmi di esempio nell'User Manual, ma sono arrivato ben presto alla conclusione che avevo bisogno di un semplice programma tutto mio, che mi permettesse di sostituire i valori differenti per le varie forme d'onda, frequenze, ecc. di tutte e tre le voci del 64. Così è stato necessario convertire i vari valori delle differenti variabili del chip del suono in numeri semplici, ed ho costruito la mia tavola delle variabili, di carattere piuttosto mnemonico (vedi **Tavola 8.1**). Riportata a pagina seguente.

Tavola 8.1: Le Variabili Sonore

VO Volume del suono (VO = 54296)
W1 Forma d'onda della Voce 1 (W1 = 54276).
W2 Forma d'onda della Voce 2 (W2 = 54283).
W3 Forma d'onda della Voce 3 (W3 = 54290).
A1 Attacco/Decadimento per la Voce 1 (A1 = 54277).
A2 Attacco/Decadimento per la Voce 2 (A2 = 54284).
A3 Attacco/Decadimento per la Voce 3 (A3 = 54291).
S1 Mantenimento/Rilascio per la Voce 1 (S1 = 54278).
S2 Mantenimento/Rilascio per la Voce 2 (S2 = 54285).
S3 Mantenimento/Rilascio per la voce 3 (S1 = 54292).
H1 Alta Frequenza per la Voce 1 (H1 = 54273).
H2 Alta Frequenza per la Voce 2 (H2 = 54280).
H3 Alta Frequenza per la Voce 3 (H1 = 54287).
L1 Bassa Frequenza per la Voce 1 (L1 = 54272).
L2 Bassa Frequenza per la Voce 2 (L2 = 54279).
L3 Bassa Frequenza per la Voce 3 (L3 = 54286).

Prima di ritornare al "*Pianeta dell'Incubo*" descriverò brevemente il mio programma, perché ciò non solo vi sarà d'aiuto per capire alcuni dei suoni che ho trovato, ma contribuirà a migliorare il vostro.

```
1 goto3
2 save"@0effetti sonori",8:verify"effett
i sonori",8:stop
3 rem
5 rem*********************************
6 rem programma effetti sonori
7 rem mike grace 24.7.83
8 rem*********************************
10 sc=54272
20 w1=54276:rem forma d'onda per voce 1
30 w2=54283:rem forma d'onda per voce 2
40 w3=54290:rem forma d'onda per voce 3
50 a1=54277:rem attack/decay per voce 1
60 a2=54284:rem attack/decay per voce 2
70 a3=54291:rem attack/decay per voce 3
80 s1=54278:rem sustain/release per voce
 1
90 s2=54285:rem sustain/release per voce
 2
```

```
100 s3=54292:rem sustain/release per voc
e 3
110 h1=54273:rem alta frequenza per voce
 1
120 h2=54280:rem alta frequenza per voce
 2
130 h3=54287:rem alta frequenza per voce
 3
140 l1=54272:rem bassa frequenza per voc
e 1
150 l2=54279:rem bassa frequenza per voc
e 2
160 l3=54286:rem bassa frequenza per voc
e 3
170 vo=54296:rem volume
200 print"premi un tasto"
210 geta$:ifa$=""then210
300 forl=0to24:pokesc+l,0:next:rem azzer
a le variabili
320 pokevo,15:rem fissa il volume
330 pokea1,17:pokea2,64:pokea3,17:rem se
tta attack/decay
340 pokes1,128:pokes2,128:pokes3,128:rem
 setta sustain/release
350 pokew1,33:pokew2,33:pokew3,129:rem s
etta lunghezza d'onda
360 pokeh1,7:pokel1,0:rem frequenza per
voce 1
370 pokeh2,3:pokel2,0:rem frequenza per
voce 2
380 pokeh3,1:pokel3,0:rem frequenza per
voce 3
400 fort=1to800:next:pokevo,0
450 print"batti 2 spazi per continuare"
455 print"'p' per stampare - 'f' per fin
ire"
460 geta$:ifa$<>"p"anda$<>"f"anda$<>" "t
hen460
470 ifa$="p"then600
480 ifa$="f"thenend
490 ifa$=" "then210
600 input"titolo";t$
```

```
610 open4,4
620 print#4,chr$(14):rem comando doppio
630 print#4,t$:rem stampa il titolo
640 print#4,chr$(15):rem torna all'ampie
zza standard
650 cmd4:list 330-380
700 print"ricordate di battere print#4:c
lose4"
900 end
```

Linee 10-170: Settano le variabili sonore.

Linee 200-210: Questa coppia stampa semplicemente il messaggio "BATTI UN TASTO QUALSIASI" e aspetta che voi lo battiate, il che provoca l'emissione del suono. È un semplice espediente per permettervi di suonare la nota che desiderate.

Linea 300: Prima di un rumore qualsiasi questa linea setta un loop per spazzolare tutto il campo delle variabili del chip sonoro, impostandole tutte a 0. È semplicemente un modo per ripartire – ed uso questa linea parecchie volte nel "*Pianeta dell'Incubo*".

Linee 320-380: Come dicono le istruzioni di REM nelle linee stesse queste settano i diversi valori. Tutto quello che dovete fare è LISTare il programma e cambiare i valori, poi fare di nuovo RUN per ascoltare i rumori cambiati. Provando e riprovando potrete eventualmente arrivare ad un suono che vi dia gli effetti desiderati.

Linee 400-900: La Linea 400 dà la lunghezza della nota. Le Linee 450-460 non sono altro che un menu per perttervi di ripetere il suono o finire il programma o stampare un LISTING che vi permetta di tener nota dei valori. Le Linee 600-650 sono istruzioni per la stampante (nel mio caso la 1515) e la linea 700 non ha in effetti alcuna utilità, se non quella di ricordarmi che devo battere "print#4: CLOSE4" alla fine del LISTING.

Il programma non è particolarmente brillante e l'ho scritto in circa 15 minuti. Ma fa il lavoro che volevo facesse – e mentre avrei potuto impiegare tempo e sforzi per farne un superbo pezzo di programmazione, con fantastici display e migliori menu, ecc., in realtà volevo solo che mi fosse di rapido aiuto. Per questo genere di cose mi va bene. La sola ragione per cui l'ho messo qui è per darvi un'idea di come ottenere il suono. La stampa di alcuni degli effetti con cui ho prodotto il suono può essere vista nella **Figura 8.3.**

RONZIO DELL'ASTRONAVE

```
330 POKEA1,17:POKEA2,17:POKEA3,17:REM ATT./DEC.
340 POKES1,64:POKES2,64:POKES3,64:REM SUST./REL.
350 POKEW1,33:POKEW2,33:POKEW3,129:REM FORMA ONDA
360 POKEH1,7:POKEL1,119:REM FREQ. VOCE 1
370 POKEH2,31:POKEL2,165:REM FREQ. VOCE 2
380 POKEH3,2:POKEL3,187:REM FREQ. VOCE 3
```

MOTORI DELL'ASTRONAVE

```
330 POKEA1,17:POKEA2,17:POKEA3,17:REM ATT./DEC.
340 POKES1,64:POKES2,64:POKES3,64:REM SUST./REL.
350 POKEW1,65:POKEW2,129:POKEW3,129:REM FORMA ONDA
360 POKEH1,37:POKEL1,162:REM FREQ. VOCE 1
370 POKEH2,1:POKEL2,12:REM FREQ. VOCE 2
380 POKEH3,2:POKEL3,187:REM FREQ. VOCE 3
```

RONZIO 2 DELL'ASTRONAVE

```
330 POKEA1,225:POKEA2,225:POKEA3,225:REM ATT./DEC.
340 POKES1,128:POKES2,128:POKES3,128:REM SUST./REL.
350 POKEW1,65:POKEW2,129:POKEW3,129:REM FORMA ONDA
360 POKEH1,37:POKEL1,162:REM FREQ. VOCE 1
370 POKEH2,1:POKEL2,12:REM FREQ. VOCE 2
380 POKEH3,2:POKEL3,187:REM FREQ. VOCE 3
```

BLEEP ? DIREZIONE

```
330 POKEA1,17:POKEA2,64:POKEA3,17:REM ATT./DEC.
340 POKES1,128:POKES2,128:POKES3,128:REM SUST./REL.
350 POKEW1,33:POKEW2,33:POKEW3,33:REM FORMA ONDA
360 POKEH1,37:POKEL1,162:REM FREQ. VOCE 1
370 POKEH2,1:POKEL2,12:REM FREQ. VOCE 2
380 POKEH3,2:POKEL3,187:REM FREQ. VOCE 3
```

BUZZ 1

```
330 POKEA1,17:POKEA2,64:POKEA3,17:REM ATT./DEC.
340 POKES1,128:POKES2,128:POKES3,128:REM SUST./REL.
350 POKEW1,33:POKEW2,33:POKEW3,129:REM FORMA ONDA
360 POKEH1,7:POKEL1,0:REM FREQ. VOCE 1
370 POKEH2,3:POKEL2,0:REM FREQ. VOCE 2
380 POKEH3,1:POKEL3,0:REM FREQ. VOCE 3
```

Figura 8.3

```
1 goto3
2 save"@0:modulo8.3",8:verify"modulo8.3"
,8:stop
3 rem
330 rem**
340 rem effetto sonoro
350 rem**
360 forl=0to24:pokesc+l,0:nextl
370 pokevo,15:pokea1,255:pokea2,255:poke
a3,255
380 pokes1,255:pokes2,255:pokes3,255
390 pokew1,17:pokew2,33:pokew3,17
400 pokeh1,37:pokel1,162:pokeh2,1:pokel2
,1:pokeh3,2:pokel3,187
1230 pokevo,0
2000 rem**
2010 rem pre-partita
2020 rem**
2030 forl=0to24:pokesc+l,0:nextl
2040 pokevo,15:pokea1,255:pokea2,255:pok
ea3,255
2050 pokes1,128:pokes2,128:pokes3,128:po
kew1,65:pokew2,129:pokew3,129
2060 pokeh1,37:pokel1,162:pokeh2,1:pokel
2,12:pokeh3,2:pokel3,187
2120 ifa$="s"thenpokevo,0
2130 form=1to150:nextm
2160 rem caduta
2170 forl=0to24:pokesc+l,0:nextl
2180 pokevo,15:pokea1,140:pokea2,140:pok
ea3,140:pokes1,0:pokes2,0:pokes3,0
2190 pokew1,35:pokew2,129:pokew3,129
2200 pokeh1,37:pokel1,162:pokeh2,1:pokel
2,12:pokeh3,2:pokel3,187
2210 print"■":pokebs,0:pokebc,0
2220 print"■■■■■■■penso che vi siate sch
iantati !!!"
2230 form=1to1000:nextm
35100 forl0to24:pokesc+l,0:nextl
35110 pokevo,15:pokea1,190:pokea2,190:po
```

```
kea3,190:Pokew1,33:Pokew2,33:Pokew3,129
35120 Pokeh1,75:Pokel1,69:Pokeh2,212:Pok
el2,230:Pokeh3,1:Pokel3,123
35130 form=1to200step.25:nextm
35170 Pokew1,0:Pokea1,0:Pokew2,0:Pokea2,
0:Pokew3,0:Pokea3,0
```

Linea 360: Come ho già detto, questa linea azzera le variabili del chip sonoro.
Linee 370-400: Questo è il primo suono che sentite nel dare il RUN del programma, ed io volevo qualcosa di alquanto misterioso e futuristico – piuttosto simile al tema con cui iniziava lo splendido film di fantascienza '*Alien*', di alcuni anni fa. I miei sforzi in merito fallirono presto, ma mi piace immaginare che vi suggerisca una profonda inquietitudine, col rumore che accompagna il titolo d'apertura.
Dando il valore 255 a tutte e tre le voci per le variabili di Attacco/Decadimento e di Mantenimento/Rilascio (vedi la Linea 370 e 380) permetto al suono di alzarsi lentamente fino al suo picco. Se il titolo fosse durato più a lungo, il suono sarebbe di nuovo finito nel silenzio, ma siccome il secondo schermo appare prima che ciò succeda, c'è un'interruzione, dal momento che lo schermo passa da nero a marrone, ed esce il primo testo.

Linea 1230: Questa è la linea che taglia i rumori d'apertura mentre lo schermo passa al primo testo, con una POKE VO (il volume), 0.

Linee 2030-2060: Non appena appare il testo inizia un suono come il tambureggiare di una nave spaziale, che rompe il silenzio quando premete S per iniziare il programma.

Queste Linee provocano quel rumore. Di nuovo notate il lento alzarsi verso un picco (nella Linea 2040 le variabili di Inizio A1, A2 e A3 sono settate nuovamente a 255), ma questa volta ho mantenuto il volume, settando il Sustain Level alto, con un bassissimo Release (il valore 128 per le variabili S1, S2 e S3 nella Linea 2050).

Linee 2120-2130: Queste due linee si inseriscono dopo il messaggio "PREMI S PER INIZIARE", interrompendo il suono dopo che avete premuto "S", e quindi provocando un breve loop di pausa.

Linee 2170*2230: Una volta che il giocatore preme S lo schermo diventa di nuovo nero, c'è un rumore di frastuono e appaiono le parole "CREDO CHE VI SIATE SCHIANTATI". Esse durano qualche secondo prima che seguano i suggerimenti circa il montaggio del nastro, e il prosieguo delle istruzioni (che abbiamo trattato nei Moduli 7.1 e 7.2). Sono queste Linee che producono questo breve scenario.

Linee 35100-35170: Le ultime linee di questo modulo riguardano Victor. Volevo che apparisse con un effetto sonoro piuttosto simile a quello di un robot caduto dall'alto. Per caso credo di aver scoperto un suono ideale (sembra più felice di qualsiasi altro). La principale differenza fra questa routine sonora e le altre simili sta nella Linea 35130 dove c'è un piccolo loop FOR... NEXT che abbassa il suono non appena Victor cade giù.
Prima di testare questo modulo finiamo di programmare, poiché il modulo finale mette l'ultimo tocco alla sequenza d'apertura.

```
1 goto3
2 save"@0:m8.4",8:verify"m8.4",8:stop
3 rem
17970 rem**
17980 rem beep
17990 rem**
18000 for l=0to24:pokesc+l,0:next l
18010 pokevo,15:pokea1,17:pokea2,17:poke
s1,64:pokes2,64
18020 pokew1,17:pokew2,17:pokeh1,17:poke
l1,195:pokel2,6
18030 fort=1to100:nextt:pokevo,0
18040 return
18970 rem**
18980 rem burp
18990 rem**
19000 for l=0to24:pokesc+l,0:next l
19010 pokevo,15:pokea1,17:pokea2,64:poke
a3,17:pokes1,128:pokes2,128:pokes3,128
19020 pokew1,33:pokew2,33:pokew3,129
19030 pokeh1,189:pokel1,172:pokeh2,7:pok
el3,233:pokeh3,1:pokel3,19
19040 fort=1to200:nextt:pokevo,0
19050 return
```

Linee 18000-18040: Questo effetto sonoro è un rumore beep. Notate i differenti valori per la variabili A (Attacco/Decadimento) e le variabili S (Mantenimento/Rilascio) che sono concentrate su un tipo di nota che sale rapidamente e poi cade veloce, piuttosto che una lenta ascesa ed un calo dei suoni precedenti. Ho inoltre usato solo due voci per questa nota e l'ho tenuta corta (FOR T=1 TO 100). Lo scopo di questa subroutine è di suonare quel rapido, abbastanza piacevole rumore, per far sapere al giocatore come ha risposto il 64.

Linee 19000-19050: Questa subroutine procura un rumore di burp alquanto più lungo (in effetti è più di un burp musicale) che dice al giocatore di aver fatto un versaccio! Riservo questa nota per i messaggi di errore o per quando avete fatto qualcosa di stupido. Questa volta ho usato nuovamente tutte le voci brevi, ed ho incluso la forma d'onda del rumore bianco (POKE W3, 129 nella Linea 19020) da aggiungere all'effetto.

Prova dei Moduli 8.3 e 8.4

Il momento è arrivato.

Allungatevi comodamente davanti allo schermo e battete RUN.
Dovreste trovare una notevole differenza ora nel vostro programma. Prima di tutto, quel certo inquietante rumore mentre compaiono fuori i titoli, e il Mealy Bug svolazza tranquillamente attraverso lo schermo. Poi il rumore profondo dei motori, mentre l'astronave inizia a seguire la propria rotta attraverso lo spazio. Poi l'improvvisa interruzione non appena fate partire il gioco – seguita dal frastuono della nave che si fracassa sulla superficie del Pianeta dell'Incubo. Una pausa, poi rispondete con N per i consigli sul nastro e le istruzioni – e Victor appare (lo vedete per la prima volta) con un rumore che riconoscerete. Egli cade sul posto, il suono smette, e dopo aver letto il messaggio potete immettere il nome. C'è un gradevole rumore di "beep" – Victor risponde e lo schermo si cancella con il messaggio "—INIZIA L'AVVENTURA...". Poi Victor sparisce. Con il programma allo stato attuale potete inoltre ottenere nuovamente le Pagine di Istruzione, poiché il programma proseguirà naturalmente in quella parte. Spero proprio che voi cominciate ad apprezzare alcuni degli sforzi che ho impiegato nel costruire un po' di suspense all'inizio del gioco.

Sommario

Ho cercato di essere molto breve nella descrizione dei meccanismi di programmazione di questo capitolo, principalmente a causa di mancanza di spazio. Ho trovato che usare gli sprite e il suono non è così difficile come può sembrare nel Reference Guide, e incoraggio quelli di voi che non hanno dimestichezza in PEEK e POKE e sedersi per un paio d'ore con la Guida e il vostro 64 e lavorarci sopra: avrete successo.
Le capacità del computer saranno dispiegate, non appena avrete superato il primo problema: apprendere qualsiasi nuovo trucco – considerate quanto detto come il primo. I titoli sono finiti e voi siete preparati a partire, cioè ad entrare nel programma. Il Capitolo 9 si rivelerà essere la maratona del libro – le Locazioni. Ma coraggio... il meglio sta per arrivare.

CAPITOLO 9

DOVE SI VA DA QUI?

Questo capitolo è probabilmente il più avvincente e il più noioso allo stesso tempo. Avvincente – perché dopo che avete finito di programmare i Moduli 9.1 e 9.2 siete in grado di vedere l'intero orizzonte della Adventure e siete in grado di maneggiare tutte le Locazioni. Noioso perché c'è una grande quantità di cose da battere, poiché si tratta ora del testo stesso.
Non ci sono scappatoie per evitare questa programmazione – ma trovo che battere tutte le linee del testo non è poi così brutto come sembra poiché scrivere parole è più facile che scrivere in BASIC. Questo è tuttora il mio problema!

Creazione della mappa

Abbiamo già trattato i principi delle mappe nel Capitolo 2, e disegnato la mappa base per *Il Pianeta dell'Incubo* nella Figura 5.2. nel Capitolo 5. Ma alla vera Adventure occorre una mappa migliore di questo – una con molte Locazioni, per conferire una dimensione alla storia e dare la sensazione al giocatore di disporre di grandi spazi.
La **Figura 9.1** mostra la mappa alla quale alla fine sono giunto. Essa segue il piano di base del mio precedente tentativo, ma ho introdotto un tempio nella città diroccata e la foresta si è ingrandita considerevolmente. Altri affinamenti alla storia originale includono un messaggio nel tempio (Casella 22) e un tesoro nella foresta (Casella 38) ed il cristallo di energia si trova ora nel lago (Casella 13).
I numeri nelle caselle nella **Figura 9.1** sono puramente fittizi, perché convertendo la mia mappa in una griglia li cambierò – come mostra la **Figura 9.3.**
Per convertire la mappa in forma di griglia ho disegnato una griglia standard 10 x 10 per poter trattare con differenti disegni (vedi **Figura 9.2**).
La mappa originale è stata praticamente presa in blocco e messa così com'è nella griglia – io ho fatto solo qualche cambiamento. Una volta che la mappa finale è stata approntata, allora si tratta semplicemente di piazzare gli oggetti appropriati nelle loro Locazioni e si è ora in grado di riprendere e programmare.

Trucchi per tracciare la mappa

Prima di cominciare a sviluppare la vostra Adventure, vi suggerisco uno o due trucchi che mi sono stati utili:
1. Disegnate sempre a matita – cambierete una quantità di cose man mano che sviluppate idee e cambiate Locazioni.
2. Cercate di incorporare un'area 'labirinto' dove si trovano parecchie Locazioni ed anche quantità di uscite per permettere al giocatore di perdervisi dentro. Il tempio e la foresta sono esempi di ciò nel *Pianeta dell'Incubo.*

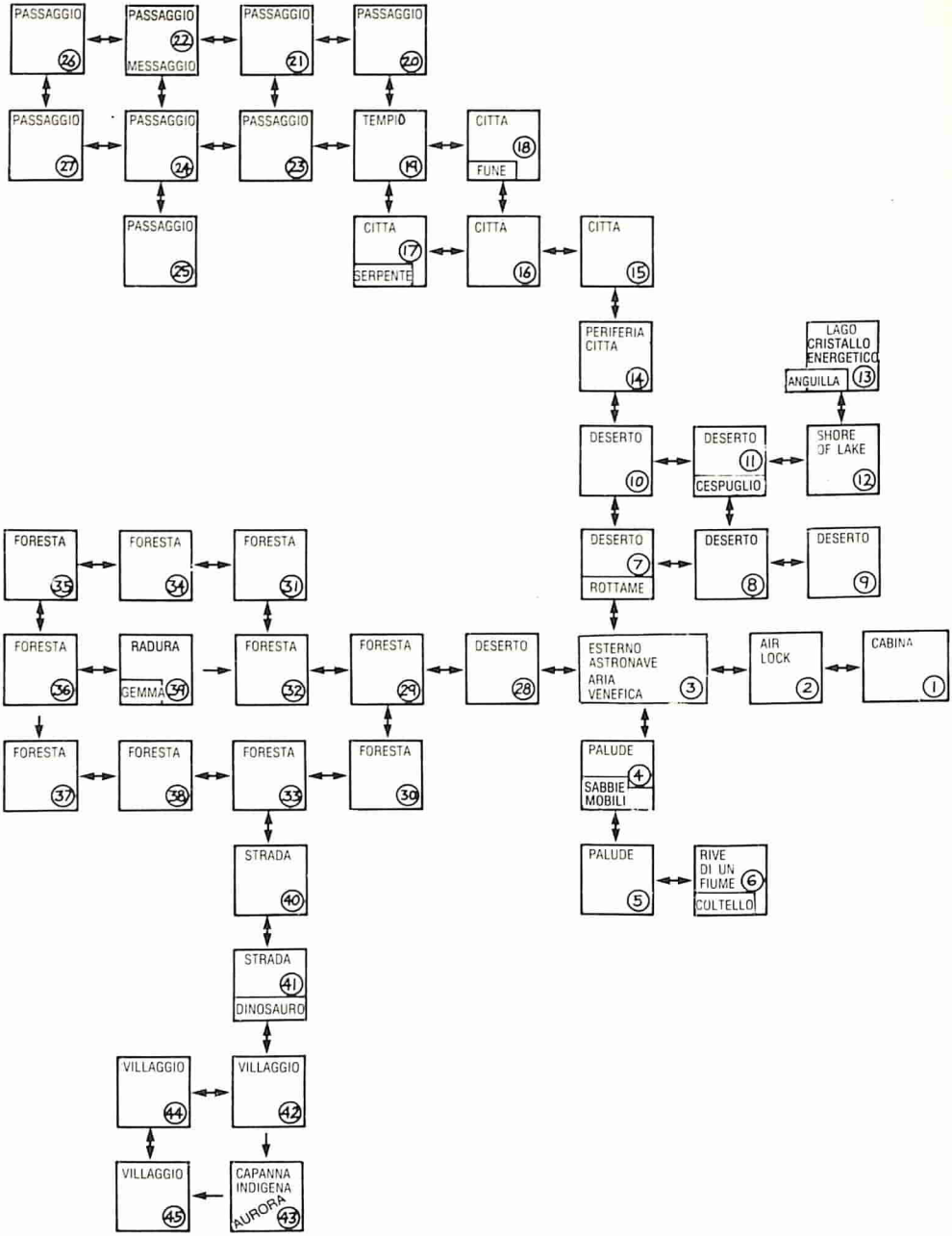
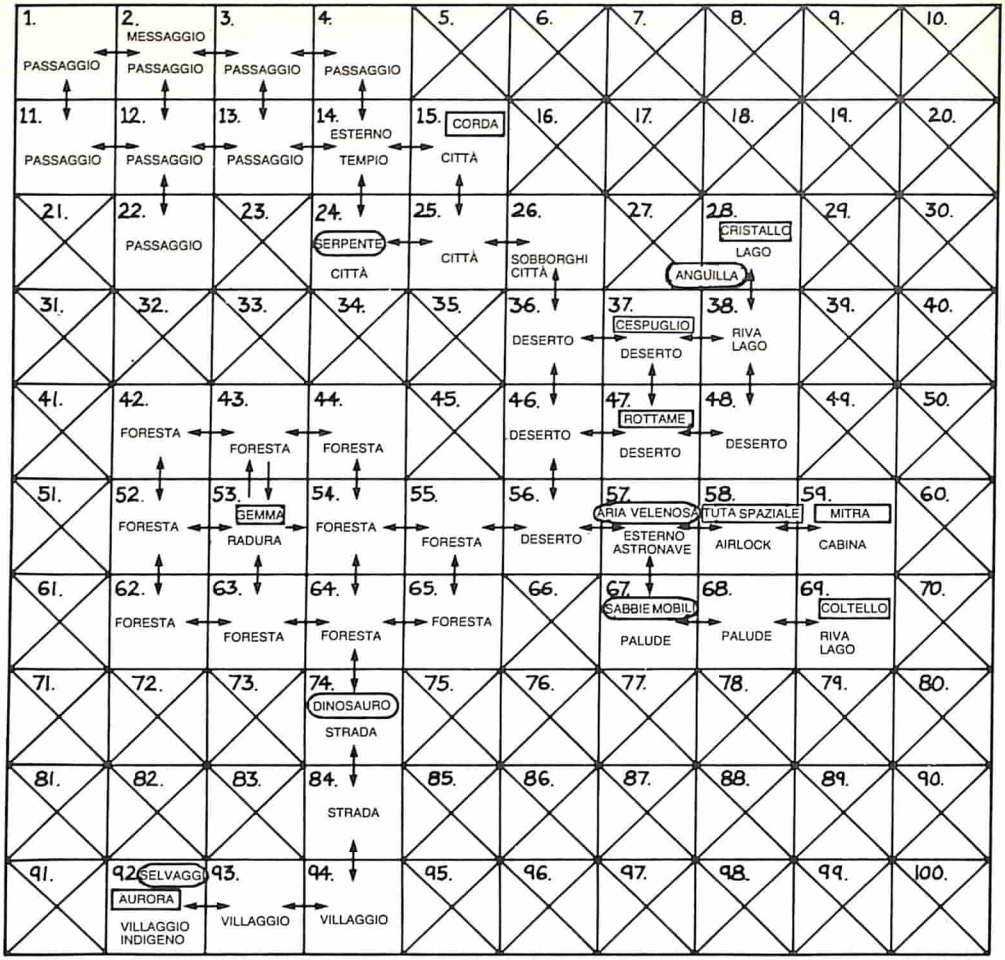


Figura 9.1 Mappa finale per "Il pianeta dell'incubo"

3. Alcune Locazioni hanno solo una direzione a senso unico. Se guardate la radura nella Locazione 53 nella **Figura 9.3** noterete che ci sono frecce che escono dalla radura in tutte le direzioni, ma potete entrare solo da ovest e da nord. Un buon modo di usare questa possibilità è di avere una Locazione riservata a, diciamo, una caverna o una stanza, che vi permetta di entrare ma non di uscire prima che un certo lavoro sia eseguito. Mostrerò ciò nella Locazione 92 più avanti nel libro.
4. Non siate troppo ambiziosi con oggetti e pericoli. L'ammontare della memoria necessaria per mettervi in grado di trattare con essi è fondamentale (come vedrete più tardi) ed il meglio è partire dal piccolo e ingrandire in seguito, se è possibile.

1.	2.	3.	4.	5.	6.	7.	8.	9.	10.
11.	12.	13.	14.	15.	16.	17.	18.	19.	20.
21.	22.	23.	24.	25.	26.	27.	28.	29.	30.
31.	32.	33.	34.	35.	36.	37.	38.	39.	40.
41.	42.	43.	44.	45.	46.	47.	48.	49.	50.
51.	52.	53.	54.	55.	56.	57.	58.	59.	60.
61.	62.	63.	64.	65.	66.	67.	68.	69.	70.
71.	72.	73.	74.	75.	76.	77.	78.	79.	80.
81.	82.	83.	84.	85.	86.	87.	88.	89.	90.
91.	92.	93.	94.	95.	96.	97.	98.	99.	100.

Figura 9.2 Una griglia 10 × 10



GLI ELEMENTI IN ⬭ SONO PERICOLI
GLI ELEMENTI IN ▭ SONO OGGETTI

Le locazioni

MODULO 9.1

```
1 goto 3
2 save"@0:modulo9.1",8:verify"modulo9.1"
,8:stop
3 rem
19970 rem**
```

```
19980 rem luo9hi
19990 rem**
20000 rem 1
20010 9osub 22050:n=0:e=2:s=11:w=0:9oto4
150
20020 rem 2
20030 Print"▮▮▮Mentre avanzate ancora di
 Piu´ dei Pezzetti di carta vi ";
20040 Print"attra99ono. In una fessura d
el muro c´e´ un messa99io con la ";
20050 Print"9rafia di aurora. Nell´incer
ta luce Potete ";
20060 Print"riconoscere ProPrio le Parol
e man9ia noci."
20070 n=0:e=3:s=12:w=1:9oto4150
20080 rem 3
20090 Print"▮▮▮Avanzando sempre Piu´Profo
ndamente nel cuore del";
20100 Print"temPio,diventa buio e umido.
  Le Pareti sono coPerte con ";
20110 Print"una fetida melma mali9na che
 sembra Pulsare con luce ";
20120 Print"malefica.…":n=0:e=4:s=13:w=2
:9oto4150
20130 rem 4
20140 Print"▮▮▮E´ ProPrio buio dentro il
 temPio.Il terreno e´ sconnesso ";
20150 Print"e sPesso cadete. APPare un P
assa99io ";
20160 Print"verso ovest .":n=0:e=0:s=0:w
=3:9oto4150
20170 rem 11
20180 9osub 22050:n=1:e=12:s=0:w=0:9oto4
150
20190 rem 12
20200 9osub 22050:n=2:e=13:s=22:w=11:9ot
o4150
20210 rem 13
20220 Print"▮▮▮Dentro al temPio e´ troPP
o buio Per vedere bene. Passa99i ";
20230 Print"sembrano mandare in Parecchi
e direzioni. Qual´e´ la mi9liore?"
20240 n=3:e=14:s=0:w=12:9oto4150
```

```
20250 rem 14
20260 print"■ Siete in un edificio che p
are un tempio. ";
20270 print"Sul pavimento fuori la stanz
a ci sono segni ";
20280 print"tracciati sul fango del corr
idoio. Sembra come una ";
20290 print"fraccia puntata ad ovest den
tro il tempio."
20300 n=0:e=15:s=24:w=13:goto4150
20310 rem 15
20320 print"■■La ricerca delle tracce v
i rende un turbine - pronti a ";
20330 print" tutto. Ma solo le pietre re
stano insensibili - ovviamente ";
20340 print"alla vostra paura. Forse non
 sono morte, dormono solo..."
20350 n=0:e=0:s=25:w=14:goto4150
20360 rem 22
20370 gosub 22050:n=12:e=0:s=0:w=0:goto4
150
20380 rem 24
20390 print"■■Avete l'impressione che q
ualcuno vi osservi nascosto nelle ";
20400 print"ombre. Nel silenzio volete u
rlare, chiamare aiuto, fare ";
20410 print"qualsiasi cosa per sapere se
 c'e' qualcuno di vivo qui in ";
20420 print"questo posto morto.":n=14:e=
25:s=0:w=0:goto4150
20430 rem 25
20440 print"■■Il vento fischia deridendo
vi, spettrale suono tra le pareti ";
20450 print"rocciose, mentre passate per
 stanze vuote, spalancando finestre. ";
20460 print"Solo questo rimane della pas
sata civilta?"
20470 n=15:e=26:s=0:w=24:goto4150
20480 rem 26
20490 print"■■Vorreste vedere dove siet
e, nella citta'! Se solo ";
20500 print"poteste trovare una via; ma
i muri sbrecciati dell'oscura casa ";
```

```
20250 rem 14
20260 print"■ Siete in un edificio che p
are un tempio. ";
20270 print"Sul pavimento fuori la stanz
a ci sono segni ";
20280 print"tracciati sul fango del corr
idoio. Sembra come una ";
20290 print"fraccia puntata ad ovest den
tro il tempio."
20300 n=0:e=15:s=24:w=13:goto4150
20310 rem 15
20320 print"■■La ricerca delle tracce v
i rende un turbine - pronti a ";
20330 print" tutto. Ma solo le pietre re
stano insensibili - ovviamente ";
20340 print"alla vostra paura. Forse non
 sono morte, dormono solo..."
20350 n=0:e=0:s=25:w=14:goto4150
20360 rem 22
20370 gosub 22050:n=12:e=0:s=0:w=0:goto4
150
20380 rem 24
20390 print"■■Avete l'impressione che q
ualcuno vi osservi nascosto nelle ";
20400 print"ombre. Nel silenzio volete u
rlare, chiamare aiuto, fare ";
20410 print"qualsiasi cosa per sapere se
 c'e' qualcuno di vivo qui in ";
20420 print"questo posto morto.":n=14:e=
25:s=0:w=0:goto4150
20430 rem 25
20440 print"■■Il vento fischia deridendo
vi, spettrale suono tra le pareti ";
20450 print"rocciose, mentre passate per
 stanze vuote, spalancando finestre. ";
20460 print"Solo questo rimane della pas
sata civilta?"
20470 n=15:e=26:s=0:w=24:goto4150
20480 rem 26
20490 print"■■Vorreste vedere dove siet
e, nella citta'! Se solo ";
20500 print"poteste trovare una via; ma
i muri sbrecciati dell'oscura casa ";
```

```
20510 Print"bloccano o9ni via eccetto un
 intrico di cunicoli verso ovest."
20520 n=0:e=0:s=36:w=25:9oto4150
20530 rem 28
20540 Print"■■Siete sulla riva di un la9
o.  Le Pietre nell'ac9ua sembrano ";
20550 Print"riflettere la luce del sole
come se avessero ";
20560 Print"un Potere ma9ico.":n=0:e=0:s
=38:w=0:9oto4150
20570 rem 36
20580 Print"■■■da 9ui Potete chiaramente
 vedere che le torri e le ";
20590 Print"9u9lie sono ora le sbrecciat
e rovine di una fiera citta' ";
20600 Print" abbandonata da9li abitanti.
"
20610 n=26:e=37:s=46:w=0:9oto4150
20620 rem 37
20630 Print"■■Infine la sabbia Prende il
 Posto dei ciottoli e sassi -";
20640 Print"■dovrebbe essere il mar9ine
del deserto. Ci sono ";
20650 Print"alcuni alberi, simili a cesP
u9li con frutti come noci di cocco ";
20660 Print"Pendono da essi.":n=0:e=38:s
=47:w=36:9oto4150
20670 rem 38
20680 Print"■■■Siete fermi nel deserto.
A nord Potete vedere un la9o. ";
20690 Print"Ad est il deserto tocca l'or
izzonte.
20700 n=28:r=0:s=38:w=37:9oto4150
20710 rem 42
20720 9osub 22110:n=0:e=43:s=52:w=0:9oto
4150
20730 rem 43
20740 9osub 22070:n=0:e=44:s=53:w=42:9ot
o4150
20750 rem 44
20760 9osub22110:n=0:e=0:s=54:w=43:9oto4
150
20770 rem 46
```

```
20780 Print"■Anche se il sole sta comin
ciando a crearvi ";
20790 Print"confusione nella mente - all
´orizzonte nord appaiono ";
20800 Print"guglie e torri di una citta´
. O e´ solo un miraggio ";
20810 Print"che vi tortura ??":n=36:e=37
:s=56:w=0:goto4150
20820 rem 47
20830 Print"■Mentre camminate rapidi sul
la sabbia rovente potete vedere ";
20840 Print" i segni di dove la vostra a
stronave e´ caduta atterrando. ";
20850 Print"Vi chiedete come farla ripar
tire.
20860 n=37:e=46:s=57:w=46:goto4150
20870 rem 48
20880 Print"■Avete camminato per ore e
potete vedere";
20890 Print" che la vasta distesa di sab
bia non ha fine. Infatti sapete che ";
20900 Print"se non tornate presto indiet
ro morirete a causa  ";
20910 Print"dell´ardore del sole che sem
bra fisso alto nel cielo."
20920 n=38:e=0:s=0:w=47:goto4150
20930 rem 52
20940 gosub 22070:n=42:e=53:s=62:w=0:got
o4150
20950 rem 53
20960 Print"■Siete in una piccola radur
a tra gli alberi. Un soffice muschio ";
20970 Print"ricopre le rive di uno stagn
o azzurro e silenzioso, dalle ";
20980 Print"acque immote; rimanete per u
n po´ prima di ricominciare la ";
20990 Print"ricerca.":n=43:e=54:s=63:w=5
2:goto4150
21000 rem 54
21010 Print"■Mentre lottate con rovi e
fogliame mandate occhiate ";
21020 Print"speranzose alla radura verso
 ovest, ma vi sembra di ";
```

```
21030 Print"non essere capaci di raggiun
gerla !!":n=44:e=55:s=64:w=0:goto4150
21040 rem 55
21050 Print"■■Siete proprio nella fores
ta ora, con alberi grossi come case ";
21060 Print"slanciati al cielo, che impe
discono la vista del sole. ";
21070 Print"Animaletti balzano fuori e f
uggono davanti a voi."
21080 Print"Due sentieri si inoltrano -
uno a sud l'altro ad ovest."
21090 n=0:e=56:s=65:w=54:goto4150
21100 rem 56
21110 Print"■■Siete al limite di un des
erto - la sabbia cambia subito ";
21120 Print"in erba di boscaglia.  Verso
 ovest ";
21130 Print"un'oscura foresta riempie il
 cielo come uno spesso tappeto ";
21140 Print" verde.":n=46:e=57:s=0:w=55:
goto4150
21150 rem 57
21160 Print"■■■Ora siete all'esterno de
ll'astronave. Attorno ci sono tracce ";
21170 Print"di bruciature e di frammenti
 carbonizzati dell'astronave. Siete ";
21180 Print"caduti in un arido deserto d
i sabbia rossastra.
21190 n=47:e=58:s=67:w=56:goto4150
21200 rem 58
21270 Print"■■■Siete nella cabina.
21290 Print"■■■Una delle tute spaziali m
anca. ";
21300 Print"Sul pavimento vedete un pezz
etto di vestito che ";
21310 Print"riconoscete come un pezzetto
 della gonna che la Principessa Aurora";
21320 Print"indossava l'ultima volta che
 l'avevate vista..":goto21380
21380 n=0:e=59:s=0:w=57:goto4150
21390 rem 59
21430 Print"■■■Siete nella cabina dell'a
stronave; la caduta ha sparso ";
```

```
21440 Print"il contenuto del vostro baul
e tutto sul Pavimento.
21460 Print"Nonostante cio' scor9ete i s
e9ni di una lotta.
21470 Print"■■■***La Principessa e' sPar
ita!***"
21480 Print"■■Il Pannello di controllo
registra che la cabina e' stata ";
21490 Print"aPerta dall'esterno e Poi ri
chiusa..."
21500 n=0:e=0:s=0:w=58:9oto4150
21510 rem 62
21520 9osub 22070:n=52:e=63:s=0:w=0:9oto
4150
21530 rem 63
21540 9osub 22110:n=0:e=64:s=0:w=62:9oto
4150
21550 rem 64
21560 Print"■■La foresta e' assolutament
e troPPo 9rande Per trovare ";
21570 Print"la via Per uscire; state com
inciando a PreoccuParvi ";
21580 Print"chiedendovi cosa mai Potrebb
e succedere.
21590 n=54:e=65:s=74:w=63:9oto4150
21600 rem 65
21610 Print"■■■Questo sentiero sembra co
ndurre via dal sole, dentro ";
21620 Print"un'oscurita' che non Pensava
te Possibile nella luce ";
21630 Print"del sole. Dietro di voi ci s
ono dei rumori, come se 9ualche ";
21640 Print"animale colossale vi stesse
se9uendo.":n=55:e=0:s=0:w=64:9oto4150
21650 rem 67
21660 Print"■■Il deserto diventa all'imP
rovviso Pieno di ve9etazione";
21670 Print" che sembra infoltirsi ad o9
ni Passo. Guardando me9lio intorno ";
21680 Print" - il terreno sembra farsi P
aludoso. ";
21690 n=57:e=68:s=0:w=0:9oto4150
21700 rem 68
```

```
21710 Print"■■Siete in una Palude, cala
, vischiosa e fastidiosa. ";
21720 Print"Potete sentire il rumore di
un fiume,immenso e veloce.";
21730 Print" Ad est c'e' il fiume - ad o
vest la Palude.
21740 n=0:e=69:s=0:w=67:goto4150
21750 rem 69
21760 Print"■■E' cosi' rinfrescante sed
ere sulla riva del fiume e ";
21770 Print" dondolare i Piedi bollenti
nella fresca corrente. Ma il fiume ";
21780 Print" e' largo, come un mare agit
ato all'orizzonte. Sarebbe ";
21790 Print"impossibile da attraversare.
":n=0:e=0:s=0:w=68:goto4150
21800 rem 69
21810 Print"■■Siete sulla strada verso
il sud della foresta. Il manto ";
21820 Print" liscio suggerisce che sia s
tata fatta da un'antica civilta' ";
21830 Print"superiore a qualsiasi altra
nota. Ma la strada e' ancora usata...";
21840 Print" o e' solo un relitto di glo
ria Passata ?"
21850 n=64:e=0:s=84:w=0:goto4150
21860 rem 84
21870 Print"■■La strada si snoda in una
verde Pianura - indugiando a nord ";
21880 Print"e a sud come una corda getta
ta da una mano gigantesca. ";
21890 Print"Dall'altra parte la vegetazi
one si avvicina viePPiu'...";
21900 n=74:e=0:s=94:w=0:goto4150
21910 rem 92
21920 Print"■■Siete in una casa infanga
ta. Pezzi di terracotta Preistorica ";
21930 Print" crepitano sotto i vostri Pi
edi.":n=0:e=93:s=0:w=0:goto4150
21940 rem 93
21950 Print"■■Prendete la via Per il vi
llaggio sPerando che gli indigeni ";
21960 Print" siano amichevoli.":n=0:e=94
```

```
:s=0:w=92:goto4150
21970 rem 94
21980 Print"■■Siete fuori la cinta del
villaggio. Case sudicie ";
21990 Print"sono sparse intorno alla rin
fusa."
22020 Print"Il villaggio pare deserto, m
a sentite una donna che ";
22030 Print"canta mestamente in una capa
nna.
22040 n=84:e=0:s=0:w=93:goto4150
22050 Print"■Siete in una specie di ded
alo di curve e incroci";
22060 Print" temendo non poco' di esserv
i perduti....."':return
22070 Print"■Potete esservi perduti nel
la foresta. Date un'occhiata alla ";
22080 Print"alla radura, un po' ad est,
un po' a sud...";
22090 Print"e sempre il rumore di una qu
alche bestia terribile echeggia e ";
22100 Print"vibra tra le foglie intorno
a voi.":return
22110 Print"■Che foresta - questa ! Semb
rate definitivamente perduti."
22120 Print"Forse avreste dovuto rimaner
e a casa, senza offrire ";
22130 Print"ad Aurora passaggio sulla vo
stra nave !":return
```

Questo modulo è fatto apposta per la programmazione delle Locazioni. Alla fine del modulo ci sono tre subroutine (Linee 22050, 22070 e 22110) che ho inserito per il tempio e la foresta, per risparmiare un po' di spazio e confondere il giocatore – poiché riceverà la stessa risposta sullo schermo in parecchi posti. A parte ciò, si deve battere tutto fino alla fine.

Nella Locazione 2 (Linea 20060) c'è un messaggio alquanto enigmatico scarabocchiato sulla parete – MANGIA LE NOCI – che risulta essere una indicazione di aiuto piuttosto oscura per il giocatore, che può non aver dedotto che 'mangiare' le noci nella Locazione 37 gli permetterà di togliere la tuta spaziale – un requisito necessario se vuole liberare Aurora. Mi piace l'idea di lasciare in giro oscure indicazioni, ed infatti questo tempio

contiene un paio di indicazioni che dovrebbero obbligare il giocatore a tirare ad indovinare. Messaggio a parte, c'è una freccia scarabocchiata nella polvere, che punta nel tempio, nella Locazione 14. È stata lasciata da Aurora? Si trova ella ancora da qualche parte nel tempio?

Ho cercato in effetti di introdurre in vari punti un elemento di mistero e di pesante tensione. A parte le indicazioni che vi portano a provare ed a cercare (senza frutti) Aurora nel tempio – probabilmente lei era lì ma era stata portata via al villaggio indigeno, prima che arrivaste voi sulla scena – ho aggiunto qualcosina circa le bestie misteriose della foresta (che non ho mai materializzato) ed la gemma nella radura della foresta, un semplice specchietto per le allodole, che non ha nessun ruolo utile.
Oltre una certa estensione questa Adventure non può essere complicata. In questo caso è da considerarsi come un programma d'esempio – non una irrisolvibile Adventure in grande stile.

Ricordate i trucchi per battere il testo, che ho descritto nel Capitolo 6 riguardo il provare poche linee man mano che procedete. È abbastanza facile fare ciò in questo modulo perché finiscono tutte in GOTO 4150, linea che non avete ancora immesso – cosicché dando direttamente l'istruzione GOTO 20080 (per la Locazione 3 ad esempio) otterrete una stampa della Locazione, poi il messaggio di UNDEF'D STATEMENT ERROR ed il programma si interromperà, permettendovi di fare LIST delle linee rilevanti ed alterarle, come desiderate.

Non dimenticate inoltre di iniziare ogni nuova Locazione con il tasto CONTROL ed il tasto "1" per colorare il testo di nero.

Il movimento nell'Adventure

Variabili nel Modulo 9.2

OB% Valore intero dell'oggetto in un Vettore Oggetto.
OB$ Descrizione in una parola singola dell'oggetto.
SI$ Descrizione con una frase dell'oggetto per la Locazione.
Questo modulo riprende gli stessi principi del Capitolo 2 del mio programma d'esempio, perciò non intendo entrare in dettagli. Le linee in molti casi sono esattamente le stesse.

MODULO 9.2

```
1 goto3
2 save"@0:m9.2",8:verify"m9.2",8:stop
3 rem
3670 rem**
3680 rem data oggetti
3690 rem**
```

```
3700 fori=1to9:readob%(i),ob$(i),si$(i):
nexti
3710 data 15,corda,un ratto si dimena pr
eso al laccio nell'angolo
3720 data 28,cristallo,il cristallo ener
getico rifulge di luce nascosta
3730 data 37,noci,ci sono alcune noci pe
r terra
3740 data 47,rottame,inciampate in un ro
ttame
3750 data 53,gemma,il riflesso di una ge
mma vi attrae
3760 data 58,tuta-spaziale,la navicella
e' qui ancora intatta
3770 data 59,mitraglietta,la vostra mitr
aglietta e' in terra
3780 data 69,coltello,tra i sassi c'e' u
n coltello preistorico
3790 data 92,aurora,vedete aurora sana e
 salva
3970 rem**
3980 rem p
3990 rem**
4000 p=59:goto4020
4010 p=p2
4020 ifp<11thenonpgoto20000,20020,20080,
20130
4030 ifp<21thenonp-10goto20170,20190,202
10,20250,20310
4040 ifp<31thenonp-20goto0,20360,0,20380
,20430,20480,0,20530
4050 ifp<41thenonp-30goto0,0,0,0,0,20570
,20620,20670
4060 ifp<51thenonp-40goto0,20710,20730,2
0750,0,20770,20820,20870
4070 ifp<61thenonp-50goto0,20930,20950,2
1000,21040,21100,21150,21200,21390
4080 ifp<71thenonp-60goto0,22510,21530,2
1550,21600,0,21650,21700,21750
4090 ifp<81thenonp-70goto0,0,0,21800
4100 ifp<91thenonp-80goto0,0,0,21860
4110 ifp<101thenonp-90goto0,21910,21940,
```

```
21970
4120 rem**
4130 rem vedi o99etti
4140 rem**
4150 fori=1to9:ifob%(i)=pthenprintsi$(i)
4190
4160 nexti
4290 rem**
4300 rem uscite
4310 rem**
4320 print"[illegible]Potete andare a";
4330 ifn>0thenprint" nord";
4340 ife>0thenprint" est";
4350 ifs>0thenprint" sud";
4360 ifw>0thenprint" ovest";
4970 rem**
4980 rem istruzioni
4990 rem**
5000 printchr$(13):x=fre(0)
5010 input"[illegible]Istruzioni:[illegible]";is$
5020 pokev+21,0
5030 ifi$="*"thenprint"[illegible]":9oto5010
6470 rem**
6480 rem i$
6490 rem**
6500 ifi$="nord"theni$="n"
6510 ifi$="est"theni$="e"
6520 ifi$="sud"theni$="s"
6530 ifi$="ovest"theni$="o"
7170 rem**
7180 rem movimento
7190 rem**
7200 ifi$="n"andn>0thenp2=p-10:9osub1800
0:9oto4010
7210 ifi$="e"ande>0thenp2=p+1:9osub18000
:9oto4010
7220 ifi$="s"ands>0thenp2=p+10:9osub1800
0:9oto4010
7230 ifi$="o"andw>0thenp2=p-1:9osub18000
:9oto4010
7260 9osub19000:print"[illegible]Piacente - non P
otete andare di li'":9oto4320
```

Linee 3700-3790; Ho riportato qui sotto nella **Tavola 9.1** nove oggetti del "Pianeta dell'Incubo".

Tavola 9.1 Oggetti del Pianeta dell'Incubo.

1. CORDA	LOCAZIONE 15.
2. CRISTALLO	LOCAZIONE 28.
3. NOCI	LOCAZIONE 37.
4. ROTTAME	LOCAZIONE 47.
5. GEMMA	LOCAZIONE 53.
6. TUTA SPAZIALE	LOCAZIONE 58.
7. MITRAGLIETTA	LOCAZIONE 59.
8. COLTELLO	LOCAZIONE 69.
9. AURORA	LOCAZIONE 92.

Queste linee sistemano gli oggetti presenti in istruzioni DATA.

Linee 4000-4100: Usando lo stesso principio del programma d'esempio, queste linee settano appunto la posizione (P = 59 che è la cabina della nave spaziale – vedi la Griglia nella **Figura 9.3**) poi indirizzano il 64 alla Locazione appropriata una volta che deciderete di muovere, tramite l'istruzione ON...GOTO.

Le linee 4150-4160: Un loop per esaminare i dati relativi agli oggetti. Se sono nella Locazione, allora SI apparirà sotto il testo che avete battuto nel Modulo 9.1. Una volta ancora puntualizzo l'importanza di avere un aspetto conveniente per SI , ovunque l'oggetto venga a trovarsi. Perciò se abbiamo detto "IL CRISTALLO BRILLA NELLE PROFONDITÀ DEL LAGO" per l'Oggetto 2 (il cristallo), quando ci si arriva la prima volta, tutto ciò sembrerà carino; ma supponete ora di lasciar cadere il cristallo nella Locazione 59; allora quello stesso messaggio suonerà alquanto sciocco, trovandoci ora in una nave spaziale.

Linee 4320-5030: Innanzi tutto provocano l'apparizione di "PUOI ANDARE" seguita dalla direzione appropriata, quindi l'input di "ISTRUZIONI?" che esce dopo ogni movimento. Ho scelto l'input "ISTRUZIONI?" piuttosto di "COSA DEVO FARE ORA?" perché mi pareva che fosse più conveniente a Victor ed all'immagine da Fantascienza del gioco.
Notate i 2 "cursor right", il simbolo " " seguito da 3 "cursor left". Ciò mette sullo schermo un asterisco sotto il cursore lampeggiante – che mi pare migliore del semplice cursore. La Linea 5030 previene ogni azione nel caso l'utilizzatore batta il simbolo " " – poiché esso sposta il cursore indietro, sopra l'asterisco.
La Linea 5020 provoca nuovamente il POKE off di Victor.
N.B. Nella Linea 5000 x = FRE(0) è un modo di fare una veloce 'raccolta di rifiuti' per cancellare tutte le stringhe di spazi, in modo da evitare un errore di OUT OF MEMORW.
Linee 6500-7260: Ancora linee standard di movimento. Notate che abbiamo P – 10 e P + 10 per il movimento verso nord e sud poiché abbiamo una griglia 10 x 10, e notate anche le subroutine di beep e di burp (GOSUB 18000 e GOSUB 19000) in queste linee.

Prova dei Moduli 9.1 E 9.2.

Troverete questo compito altrettanto piacevole (se non di più) di quello di testare i Moduli 8.3 e 8.4. Ora potete realmente spostarvi nella mappa e vedere gli oggetti disposti nelle Locazioni appropriate.
Utilizzate un po' di tempo per esplorare, immaginando di giocare per la prima volta. C'è una certa soddisfazione nel sentirsi in grado di spostarsi nella Adventure.
Faccio qui un appunto per quando progetterete il gioco. Io ho per prima cosa definito le Locazioni come se stessi partendo con la nave ed esplorando veramente – per esempio volevo dire "State lasciando il deserto ed entrando nella città diroccata...". Questo suonava bene nel caso della città – ma se in seguito stessi 'lasciando' nuovamente la città, sembrerebbe piuttosto strano vedere il messaggio "State lasciando il deserto..." perché non sarà così! Io sto' lasciando la città!
Perciò provate sempre a scrivere la descrizione, anche se potete andare sia dentro che fuori la Locazione, ed essa vi sembra giusta. Il tempo passato a tirar fuori queste cose in precedenza è ben speso, ed infatti stavo cambiando alcune delle diciture delle mie Locazioni quando proprio all'ultimo momento individuai un altro errore!!
Questo capitolo è stato una prova di programmazione abbastanza estenuante – ma il risultato dovrebbe apparire di grande effetto. Il prossimo passo sarà di cominciare a raccogliere ed a lasciar cadere gli oggetti.

Sommario

Secondo me l'aggiunta delle locazioni è la parte più interessante, probabilmente, dell'intero gioco. Alla fine posso cominciare a sentire l'ampiezza del mio mondo immaginario, posso vedermi in vari scenari, sentire la sabbia scottante sotto i piedi ed il sole sul collo, o dipingermi il fiume ruggente distendersi come un mare che pulsa fino all'orizzonte. La foresta con i suoi piccoli abitanti alieni, la città desolata e scolpita dal vento, le umide profondità dell'oscuro interno del tempio alieno – è tutto qui!
Io amo la fantascienza, amo le terrificanti copertine che ornano molti libri e riviste del genere, la mia mente si libra fuori dell'esistenza quotidiana, in un mondo di emozioni e fantasia, dove posso disfarmi liberamente dei miei limiti e delle necessità, e volarmene via.
Se non altro lo scrivere questo programma mi ha permesso di vivere un po' di più, in questi ultimi pochi mesi, in un mondo che visito così poco, ma che non perderò mai più.

CAPITOLO 10

PRENDIAMOLO PROPRIO ORA!

Come in tutte le buone Adventure è venuta l'ora di iniziare a manipolare i nostri oggetti e il nostro eroe del mondo immaginario dello spazio profondo. L'arte di prelevare o di raccogliere oggetti o di lasciarli cadere, è stata spiegata nel Capitolo 3, quindi adesso tornerò solo a scorrere velocemente le tecniche di programmazione.
Una delle regole d'oro nel fare Adventure è quella di raccogliere sempre un oggetto non appena lo si vede. Quindi, quando studiate sempre più in profondità il tracciato, iniziate col caricarvi di un armamentario variamente assortito. Ho voluto render la vita un po' difficile mettendo alcuni oggetti inaccessibili senza prima aver acquisito altri oggetti – per esempio il coltello in Locazione 64 non può essere raggiunto senza prima aver "preso" la corda in Locazione 15. Altrimenti non potete sfuggire alle sabbie mobili in Locazione 67.
Un altro modo di render la vita difficile è quello di definire un numero massimo di oggetti che potete trasportare nello stesso tempo, cosa che tratterò nel Modulo 10.3. Ma torniamo alla tastiera.

L'Airlock e altre cose

Ho voluto aggiungere un piccolo particolare in più, quando usciamo dall'arilock (Cabina di decompressione): usando suono e immagini si ha l'impressione che la airlock si apra veramente. In verità i puristi potrebbero obiettare che essa si dovrebbe aprire due volte per essere una vera airlock, ma come per l'uso del rumore nella sequenza di avanzamento nello spazio della navicella, la mia airlock si apre solo una volta (per risparmiare tempo).

MODULO 10.1

```
1 goto3
2 save"@0:m10.1",8:verify"m10.1",8:stop
3 rem
6540 ifi$="n"ori$="e"ori$="s"ori$="o"the
n7070
6550 ifi$="i"orleft$(i$,3)="inv"then1000
0
6570 ifi$="istruzioni"thengosub15000
```

```
6580 ifi$="corri"ori$="cammina"then9osub
18000:Print"Quale via?":9oto4320
6590 ifi$="aPri airlock"andP=57then7100
6700 ifi$="aPri airlock"andP=58then7100
6840 ifi$="save"then17000
6850 ifi$="esci"ori$="fine"then9oto17500
7000 ifi$<>"n"andi$<>"e"andi$<>"s"andi$<
>"o"then7300
7040 rem**
7050 rem airlock(cabina - camera decomPr
essione)
7060 rem**
7070 ifP=58andi$="o"then7100
7080 ifP=57andi$="e"then7100
7090 9oto7200
7100 Print"Si aPre la cabina..."
7110 forl=0to24:Pokesc+l,0:nextl
7120 Pokevo,15:Pokea1,17:Pokea2,17:Pokes
1,130:Pokes2,130
7130 Pokew1,17:Pokew2,17:Pokeh1,13:Pokel
1,78:Pokeh2,253:Pokel2,46
7140 fort=1to1500:nextt:Pokevo,0:Print"
                          ":rem 20 sPazi
7150 fort=1to700:nextt:Print"airlock aP
erta......"
7160 fort=1to700:nextt:9oto7200
7240 ifleft$(i$,8)="aPri air"andP=57then
P2=P+1:9osub18000:9oto4010
7250 ifleft$(i$,8)="aPri air"andP=58then
P2=P-1:9osub18000:9oto4010
```

Linea 6540: Questa Linea è un by-pass per tutti i comandi I$ che programmeremo più avanti. Quello che fa è mandare il programma alla sezione di movimento evitando così ogni "messaggio di errore", ecc. conseguente ad una ripetizione di I$. Questo risulterà più ovvio in seguito nella programmazione (oppure potete andare direttamente al programma finito – vedi Appendice D – per vedere cosa intendo dire).

Linee 6550-6580: Alcuni comandi I$ che ho messo qui.

Linee 6690-6700: Sebbene possiate muovervi facilmente all'interno della nave spaziale e possiate anche uscirne dando un comando EST o OVEST, qualcuno potrebbe battere "APRI AIRLOCK", e mi sembra opportuno occuparmi anche di questi. Allora queste due Linee inviano il programma alla Linea 7100 che è la routine che apre la airlock

(notate, a condizione che siate nelle Locazione 57 o 58 – le Locazioni interessate). Se voi battete "APRI AIRLOCK" in qualunque altra Locazione esce una risposta standard.

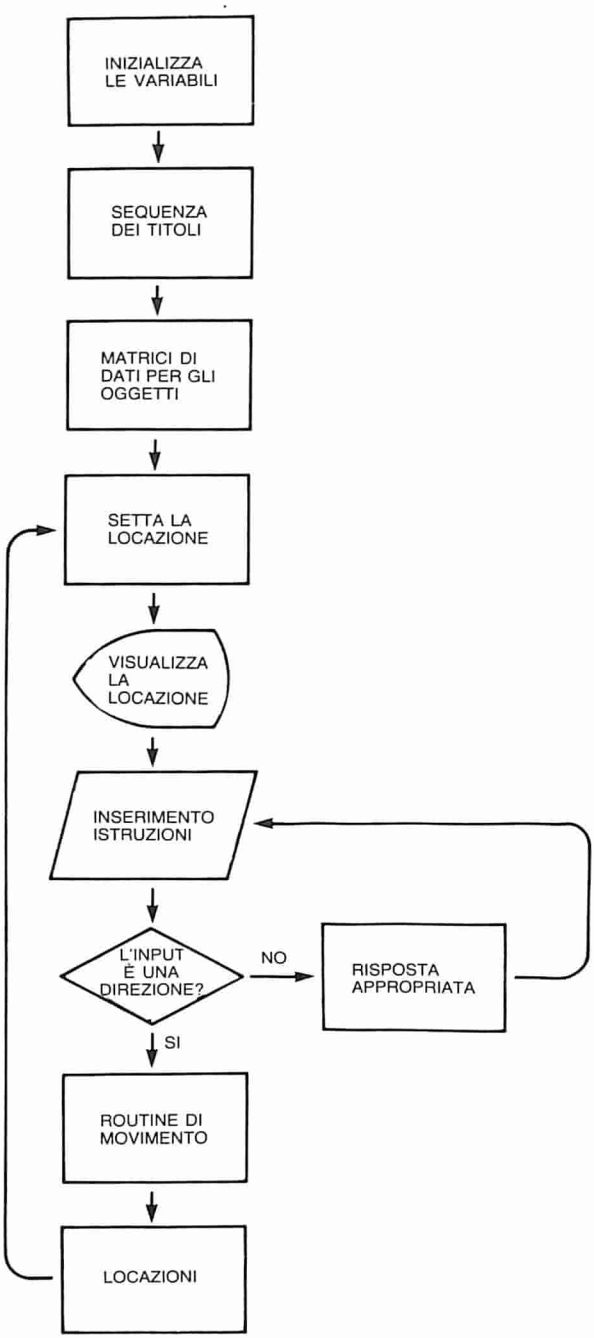
Linee 6840-6850: Vengono considerati i comandi SAVE e ESCI

Linea 7000: Questa Linea manda il programma dopo l'area del movimento, se non è stata battuta nessuna direzione. Ci risparmia di doverci preoccupare di risposte come "NON PUOI ANDARE IN QUESTA DIREZIONE" nell'area dei movimenti.
Se cominciate a non raccapezzarvi sulle varie sezioni del programma, il **Diagramma di flusso 10.1** vi aiuterà a chiarirvi dove siamo a questo stadio.

Linee 7070-7250: Questa sezione è una simulazione visiva e sonora dell'apertura della airlock. Appena voi battete O o APRI AIRLOK il 64 calcola in quale direzione state andando (Linee 7070 e 7080), e stampa le parole "si apre l'airlock...".
Nello stesso momento si sente un suono ronzante (Linee 7110-7130). Come il ronzio finisce le parole "si apre l'airlock..." vengono cancellate (Linea 7140 – notate i 20 spazi per coprire le parole). C'è una breve pausa e quindi viene stampato "airlock aperta..." a rimpiazzare il messaggio originale (Linee 7150 e 7160.
Alla fine appare la prossima Locazione (Linee 7340 e 7250). Queste ultime due Linee operano al posto del comando ON... GOTO nel Modulo 9.2.

Prova del Modulo 10.1

Fate girare il programma, e quindi raggiungete la prossima Locazione, battete O per entrare dentro la airlock. Ora battete APRI AIRLOCK, e dovreste vedere apparire la parola e sentire il delizioso ronzio della porta che si apre. Devo confessare che ottenere questo suono particolare mi ha preso alcune ore di tentativi e di errori, ma ne è valso la pena.
Potete ora tornare indietro nella nave battendo E o APRI AIRLOCK di nuovo. Ancora le parole e il ronzio accompagneranno il vostro movimento.
Non c'è molto altro che voi possiate ancora fare con questo Modulo, eccetto battere CORRI o CAMMINA per ottenere la risposta. Se battete SAVE, lo schermo si dovrebbe sbiancare e dovrebbe apparire un messaggio "SIETE SICURI?". Se volete vedere se avete introdotto correttamente la routine SAVE nel Modulo 7.2, allora date pure il SAVE in questo stadio. Per completare il test rigirate il programma e questa volta date il LOAD (caricatelo da nastro cioè). Dovrebbe andare tutto.



Flowchart 10.1 La struttura di base di Il pianeta dell'incubo

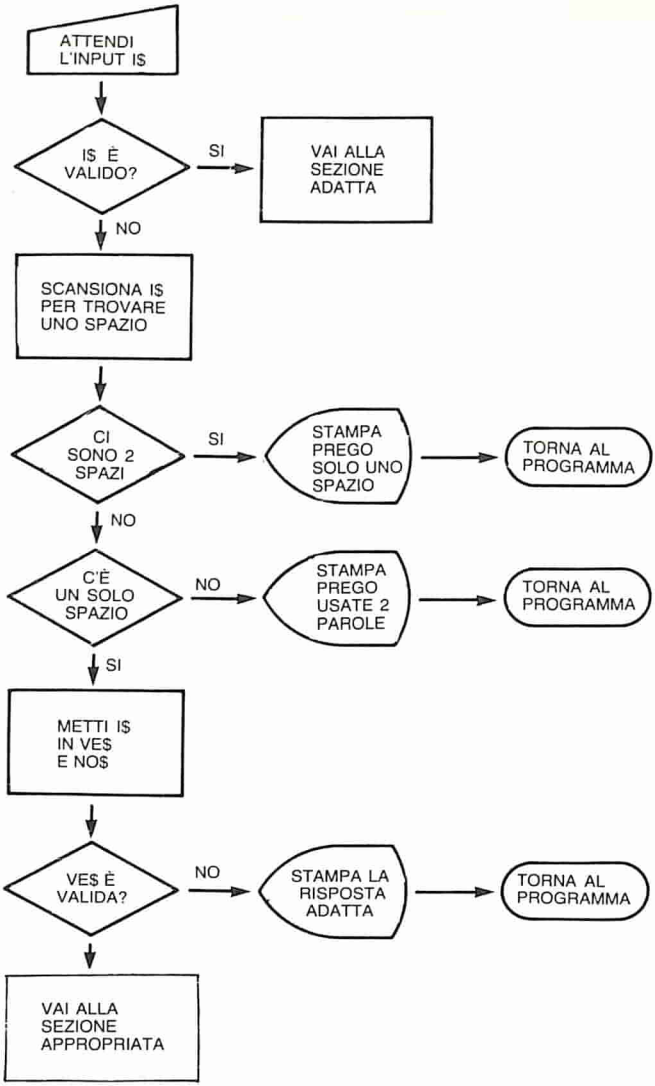
MODULO 10.2

```
1 goto3
2 save"@0:modulo10.2",8:verify"modulo10.
2",8:stop
3 rem
7270 rem**
7280 rem cerca uno spazio su i$
7290 rem**
7300 fori=1tolen(i$)
7310 ifmid$(i$,i,1)=" "then7340
7320 nexti
7330 print"■Potete usare due parole":gos
ub19000:goto5000
7340 fori=1tolen(i$)
7350 ifmid$(i$,i,2)="  "then7380
7360 nextsp
7380 print"■Solo ■uno■ spazio tra parole
":gosub19000:goto5000
7390 rem**
7400 rem i$-ve$&no$
7410 rem**
7420 forsp=pi+1tolen(i$)
7430 ifmid$(i$,i,1)=" "then7450
7450 ve$=left$(i$,sp-1)
7460 no$=right$(i$,(len(i$)-sp))
```

Linee 7300-7330: Queste Linee sono un'esatta copia del Modulo 1.5 nel Capitolo 3, quindi ripeterò solamente che esse sono qui per vedere se il giocatore ha battuto due parole, e se no, allora appare il messaggio "POTETE USARE DUE PAROLE PER FAVORE?".

Linee 7340-7380: Estendendo il concetto precedente, questa sequenza controlla se ci sono due spazi tra le parole (dato che questo impedisce al 64 di riconoscere le parole appropriate). Questa volta il messaggio è "LASCIATE UN SOLO SPAZIO TRA LE PAROLE PER FAVORE". Il principio del riconoscimento di due spazi utilizzando la funzione MID$, è lo stesso che per riconoscerne uno.

Linee 7420-7640: E ancora una volta questa routine è una copia esatta del Modulo 1.6 nel Capitolo 6. I$ viene separato in VE$ e NO$ (verbo e nome).
Per capire meglio il principio della gestione delle stringhe suggerisco di vedere il **Flow-Chart 10.2**.



Flowchart 10.2

Prova del Modulo 10.2

Questo è abbastanza facile da provare.
Una volta che siete in qualche Locazione battete "VAICAVALLO" per esempio, e

dovrebbe apparire il messaggio "USATE DUE PAROLE PER FAVORE". Adesso battete "VAI CAVALLO" (con due spazi) e dovrebbe apparire il messaggio appropriato. Finalmente battete "VAI CAVALLO" (il messagio corretto) e, dato che verrà riconosciuto valido, il programma andrà alla routine delle Istruzioni. Una volta fatto ciò, verrete fermati con un messaggio di errore. Adesso potete battere PRINTVE$ in modo diretto e dovreste vedere "VAI", mentre se battete PRINTNO$ dovreste vedere "CAVALLO". Se ciò non succede, tornate indietro e controllate le vostre Linee dato che avrete fatto senz'altro un errore di sintassi o nelle parentesi.

Stiamo raccogliendo un po' troppo!

MODULO 10.3

```
1 goto 3
2 save"@0:modulo10.3",8:verify"modulo10.
3",8:stop
3 rem
7470 rem**
7480 rem ve$
7490 rem**
7560 ifve$="trova"thengosub18000:print"▒D
ovete  guardare in giro.":goto5000
7610 ifve$="prendi"orve$="afferra"orve$=
"acchiappa"orve$="porta"then11700
9970 rem**
9980 rem inventario
9990 rem**
10000 print:print"▒L'inventario e':":iv=
0
10010 fori=1to9
10020 ifob%(i)=-1thenprint"▒"ob$(i):gosu
b18000:iv=iv+1
10030 nexti
10040 ifiv=0thenprint"▒Niente - spiacent
e."
10050 goto5000
11670 rem**
11680 rem get
11690 rem**
11700 fori=1to9:ifob$(i)=no$then11740
11710 nexti
11730 ifob$(i)<>no$then11850
11740 ifob%(i)=-1thenprint"▒Lo avete gia
```

```
!!":gosub19000:goto5000
11800 ifob%(i)<>pthenprint"Non vedo ";n
o$;" qui.":gosub19000:goto5000
11810 ifg>4thenprint"State raccogliendo
 troppo - "
11820 ifg>4thenprint"Dovete lasciare qua
lcosa!!":gosub19000:goto5000
11830 print"Okay.":gosub18000:ob%(i)=-1
:g=g+1
11840 goto5000
11850 gosub18000:print"Non voglio secca
rmi con tale ";n$;" !!":goto5000
```

Linea 7560: Uno dei problemi che si devono superare nell'Adventure è quello di cercare di anticipare i diversi Comandi che la gente può usare. TROVA sembrava uno di quelli alquanto comuni (infatti io ho avuto persino un giocatore che ha battuto "TROVA PRINCIPESSA" come primo Comando, proprio all'inizio del gioco, come se il computer avesse dovuto rispondere dov'era!!!). Quindi ho aggiunto questa Linea per tener conto di tutti i potenziali "trovatori".

Linea 7610: Questo è il Comando che ci abilita a prendere i nostri oggetti. Notate che ho anche aggiunto "afferra", "acchiappa", "porta" (sinonimi di "prendi") a questo Comando.
Linee 10000-10050: Una routine standard per l'Inventario come nel Capitolo 3. (Modulo 1.7). L'unica aggiunta è che c'è un suono (beep) che accompagna ciascun oggetto, quando viene mostrato.
Notate che non ho messo suoni quando viene stampato "NIENTE". Questo è abbastanza d'effetto dato che il giocatore aspetta i suoni che accompagnano ciascun Messaggio, e, se non sente niente, guarda per vedere il perché. Ci sono volte che il silenzio fa più effetto del rumore.

Linee 11700-11840: La routine PRENDI, di nuovo come nel Capitolo 3. Notate tuttavia l'aggiunta di un paio di Linee (11810 e 11820) che permettono alla variabile G di arrivare a cinque oggetti nell'Inventario prima di provocare "STAI TRASPORTANDO TROPPO".
Linea 11850: La risposta standard al tentativo di prendere un oggetto che non é nei dati e quindi considerato non valido.

Prova del Modulo 10.3

Provate a battere "Inventario" una volta che siete nel programma, allo scopo di avere la risposta "NIENTE". Quindi se cominciate ad andare in giro a prendere oggetti, dovreste poterli ritrovare nell'Inventario in qualunque momento lo chiediate.
Potete anche eccedere, e allora con un simpatico richiamo non vi verrà permesso di prendere nient'altro.
Ora che cominciate a dovervi muovere nell'Adventure per provare ciascun Modulo, potete trovare più semplice salvare il programma su nastro per poi ricaricarlo subito all'inizio di ciascun test.
Avendo raccolto una quantità di oggetti, dobbiamo esser capaci di lasciarli; il prossimo Modulo descriverà come farlo.

Lasciar cadere di nuovo

MODULO 10.4

```
1 goto 3
2 save"@0:modulo10.4",8:verify"modulo10.
4",8:stop
3 rem
7620 ifve$="esamina"thengosub18000:goto7
820
7630 ifve$="butta"orve$="perdi"orve$="la
scia"orve$="togli"then12000
7720 ifve$="getta"thengosub18000:goto786
5
7820 ifp=25andleft$(no$,4)="port"thenPri
nt"■solo un Palazzo vuoto.."：goto5000
7830 ifp=28andno$="lago"thenprint"■Il cr
istallo e' proprio qui"：goto5000
7840 ifp=37andno$="cespuglio"thenPrint"■
Le noci sembrano buone."：goto5000
7850 ifp=37andleft$(no$,5)="frutt"thenPr
int"■Sembrano commestibili."：goto5000
7860 Print"■Non c'e' nulla di utile qui.
"：goto5000
7865 ifno$="victor"thenhr=2：goto11300
7870 fori=1to9：ifob$(i)=no$thengoto7900
7880 nexti
7890 Print"■Smettila di Perder tempo!!"：
goto5000
7900 ifob%(i)=-1thenprint"■Non Potete ve
dere molto qui!!"：goto12090
7910 Print"■Non l'avete!：goto5000
11970 rem**
11980 rem butta
11990 rem**
12000 fori=1to9：ifob$(i)=no$then12040
12010 nexti
12030 ifob$(i)<>no$thengosub18000：Print"
Nessun Punto li'."：gosub5000
12040 ifob%(i)<>-1thenPrint"■Non l'avete
 ancora!!!"：gosub19000：goto5000
12080 Print"■Okay."：gosub18000
12090 ob%(i)=p：g=g-1：goto5000
```

Linee 7620 e 7820-7860: Uno dei Comandi che la gente usa spesso nell'Adventure è ESAMINA. La Linea 7620 accetta questo Comando e manda il programma alla Linea 7820 dove vengono dati alcuni Messaggi. Avevo deciso di non coinvolgermi troppo nel problema di "esaminare" oggetti, perché avevo già abbastanza di che occuparmi.

Linee 7820-7850: Queste linee sono connesse con alcune Locazioni (dove immaginavo che qualcuno potesse provare a inserire "ESAMINA" qualcosa) e dà i Messaggi alle porte della città (notate LEFT$ (NO,4)="PORT"), al lago, e per i frutti e le noci nel deserto. Avrei potuto espandere questa Sezione finché avessi voluto, ma lascio a voi farlo, quando avrete finito, e saprete allora quanta memoria vi rimane!

Linea 7860: È l'ultima della serie che dà il Messaggio "NIENTE DI UTILE QUI" se il giocatore prova ad "esaminare" qualche cosa d'altro.

Linea 7630: I Comandi BUTTA, PERDI, LASCIA, TOGLI attiveranno la routine "BUTTA". La possibilità di avere diverse scelte per questo Comando è utile perché mentre qualcuno potrebbe facilmente usare BUTTA, per un coltello, è più verosimile che usi TOGLITI, per la tuta spaziale o LASCIA, per Aurora, e quindi aggiungendo queste parole in più si rende più versatile il gioco.

Linee 7720-7910: Queste Linee formano una piccola routine che si occupa del Comando LANCIA.

Ho deciso che dovevo occuparmene per le stesse ragioni della routine "butta", infatti se lanciate un oggetto vi aspettate che non si trovi più nell'Inventario. Quindi diventa necessario controllare se l'oggetto che viene lanciato è un oggetto valido (Linee 7870-7880) e se l'oggetto è o no nell'Inventario (Linea 7870). Se si, il programma va alla Linea opportuna (Linea 7900) e stampa il Messaggio "NON POTETE VEDERE MOLTO QUI", così il giocatore sa che non è riuscito in nient'altro.

A questo stadio andiamo alla 12090, dove gli oggetti possono uscire dall'Inventario ed entrare nella Locazione (vedi la routine "butta").

Se l'oggetto non è nell'Inventario, allora la Linea 7910 stampa "NON L'AVETE".

La Linea 7890 è una Linea di base per ogni altro oggetto che possa essere lanciato, es. "LANCIA CAVALLO". La gente può scrivere le cose più pazze in un gioco d'Adventure (sono sicuro che lo sapevate), quindi abbiamo bisogno di messaggi che considerino sia i commenti stravaganti che i genuini tentativi di lanciare un oggetto che è citato nella descrizione ma che non è un oggetto valido (per es. il vasellame nella Locazione 92). Quindi un Messaggio "NON LO VEDO QUI" sarebbe inaccurato dato che il vasellame è ovviamente qui. Ho pensato che "SMETTETELA DI DIRE SCIOCCHEZZE" fosse un buon compromesso.

Finalmente la Linea 7865 è una Linea che ho aggiunto in seguito per considerare il fatto che qualcuno poteva pensare di lanciare Victor. Se lo fate, andate al Modulo 14.1 (Linea 11300) dove vi troverete una bella sorpresa.

Linea 12000-12090: È la Sezione "butta" standard come nel Capitolo 3. Niente da aggiungere qui, eccetto che dobbiamo ricordarci di togliere oggetti (G=G-1) nella linea 12090 per permettere al giocatore di prenderne altri se è vicino al limite.

Allora volete finire?

MODULO 10.5

```
1 goto 3
2 save"@0:modulo10.5",8:verify"modulo10.
5",8:stop
3 rem
17470 rem**
17480 rem esce
17490 rem**
17500 pokebs,2:pokebc,2
17510 print"■":printtab(5)"■■■■■■■Siete
sicuro ";n$;" ??":gosub19000
17520 printtab(5)"■■Batti ■c■ per ■c■ont
inuare:"
17530 printtab(5)"■■Batti ■f■ per ■f■in
ire il gioco:"
17540 geta$:ifa$<>"c"anda$<>"f"then17540
17550 ifa$="f"then17570
17560 pokebs,11:pokebc,11:print"S":goto4
020
17570 print"■":print"■":poke53272,21:pok
ebs,14:pokebc,6:end
```

Linea 17500: Lo schermo diventa rosso, per conferire aspetto il dramma.

Linee 17510-17550: Questa Sezione controlla che voi vogliate veramente lasciare il gioco. Se è così il programma andrà alla Linea 17570.

Linea 17560: Se premete E lo schermo torna al familiare grigio e si pulisce, e voi tornate all'inizio del gioco.

Linea 17570: Questa è la reazione al vostro ESCI, e fa tre cose. Per prima cosa pulisce lo schermo. Quindi c'è una POKE 53272,21 per riconvertire in caratteri maiuscoli. Infine lo schermo tornerà ai familiari colori blu scuro e blu chiaro del 64. Il Programma finisce col computer che appare come dovrebbe, dopo aver terminato un gioco.

Prova dei Moduli 10.4 e 10.5

Per provare questi Moduli dovete passare da tutte le possibili combinazioni di Comando che potete pensare. Dopo essere entrati nel gioco dovete:

1. Esaminare qualcosa (in qualunque Locazione).
2. Buttare via un oggetto che avete preso precedentemente.
3. Uscire e rientrare in una Locazione per vedere se l'oggetto è mostrato sullo schermo.

4. Buttar via un oggetto valido che non avete preso.
5. Buttar via un oggetto non valido (come un sasso).
6. Guardare il vostro inventario ad ogni stadio.
7. Provare a dare ESCI o FINE e provare poi a continuare (C) o terminare veramente (Q).

Dovreste riuscire a fare tutte queste cose ed avere le risposte corrette con un beep o un rumore. La struttura standard del gioco è quasi finita adesso, ed è ora di pensare ad aggiungere un po' di pericoli e di emozioni nella vostra storia.

Sommario

La maggior parte di questo Capitolo è stato un sommario della programmazione originale nella Parte 1 di questo libro e perciò sono stato alquanto schematico. È abbastanza interessante notare come ho sviluppato alcuni Moduli per aggiungere una vena di sofisticatezza a *Il Pianeta Dell'Incubo*. Spero che ciò vi dia delle idee che possano essere utili per la vostra Adventure.

CAPITOLO 11

NON PER QUELLA VIA

Il Pianeta dell'Incubo contiene solo sei pericoli di base – non tanti come potreste immaginare. Ma la programmazione di questi soli sei pericoli mi ha dato la prova di essere la più frustrante e la maggior perdita di tempo di tutto il gioco. Quindi quando batterete le linee dei prossimi comandi, pensate un pochino allo sforzo che ho dedicato ad eseguire ciò che sembrerà essere una parte relativamente piccola dell'intero programma. Quando iniziate a pensare ai pericoli del vostro gioco è meglio essere cauti all'inizio, perché potete sempre aggiungere qualcosa in un momento successivo, se il tempo, la memoria e la salute ve lo consentiranno.
I sei pericoli di base sono:

1. Un serpente nella Locazione 24 che metterà in pericolo la vostra vita.
2. L'anguilla sott'acqua nella Locazione 28 che farà lo stesso.
3. L'atmosfera velenosa che vi ucciderà se non avete la tuta spaziale.
4. Le sabbie mobili nella palude nella Locazione 67 che vi inghiottiranno finché non morirete.
5. Un dinosauro nella Locazione 57 che vi impedirà di andare verso sud finché non avrete risolto il problema di come passare.
6. Gli indigeni ostili nella Locazione 92 che vi impediscono di portare via Aurora.

Oltre ai sei ovvi pericoli ve ne sono altri di minore entità, maggiormente in relazione con l'obiettivo finale di vincere il gioco. Questi sono:

1. Il fatto che non potete scappare dal pianeta senza Aurora.
2. Potete scappare con Aurora solo se la baciate (dopo che l'avete trovata).
3. Non potete baciare Aurora senza togliervi la tuta spaziale – nonostante l'atmosfera velenosa.
4. Dovete mangiare le noci nella Locazione 37 per essere in grado di togliervi la tuta spaziale in sicurezza – notate l'indizio nella Locazione 2.
5. Il fatto che il cristallo di energia dell'astronave è stato perso e dovete ritrovarlo per scappare dal pianeta.
6. Non potete immergervi nell'acqua del lago (per prendere il cristallo) a meno che non abbiate qualcosa di 'pesante' per immergervi (questo è il rottame nella Locazione 47).7. Se non avete la tuta spaziale, non appena vi immergete in profondità affogate.
8. Dato che il vostro mitra non funziona sott'acqua, potete uccidere l'anguilla solo col coltello (trovato nella Locazione 69).
9. Per prendere il coltello dovete attraversare le sabbie mobili nella Locazione 67 – e non uscirete di lì se non avete la corda (trovata nella Locazione 15).
10. Potete uccidere il serpente solo con il mitra.
11. Non potete uccidere il dinosauro e dovete scalare un albero per oltrepassarlo dondolandovi di albero in albero.

Come vedete dalla seconda lista, i sei pericoli originali hanno creato un bel po' di programmazione. Per mettervi in grado di capire come ho sviluppato l'idea e per aiutarvi a sviluppare la vostra, ho provato nell'Adventure a collegare diversi oggetti con diversi pericoli, ed a rendere abbastanza difficile trovare la giusta strada.

L'uso delle variabili.

Una delle prime cose da considerare è quella di programmare il 64 in modo che il pericolo appaia quando lo incontrate la prima volta – ma dovete anche fare in modo di nasconderlo nel caso che torniate alla stessa Locazione in un momento successivo. Ho realizzato ciò impostando un certo numero di variabili a 0 (Linea 290 nel programma – Modulo 6.1) e quindi una volta che i pericoli sono stati superati, impostando la variabile a 2. Questo principio è stato spiegato nel Capitolo 4 – l'unica differenza qui è che ci sono un numero maggiore di variabili.

Tabella delle Variabili per i Pericoli.

PR	impostato a 2 quando la Principessa è salvata.
CV	impostato a 2 quando il Cristallo è trovato.
SV	impostato a 2 quando il Serpente è stato ucciso.
EV	impostato a 2 quando l'Anguilla è stata uccisa.
QV	impostato a 2 quando le Sabbie Mobili sono state superate.
DV	impostato a 2 quando il Dinosauro è stato oltrepassato.
NV	impostato a 2 quando gli Indigeni sono diventati amici.
FV	impostato a 2 quando le Noci sono state mangiate.
ES	impostato a 2 quando l'Anguilla è stata uccisa;
	impostato a 1 quando l'Anguilla vi ha impedito i movimenti.
NP	impostato a 1 quando siete arrivati alla capanna una volta.
NA	impostato a 1 per permettervi il movimento dalla capanna.
Z	impostato a 1 quando le noci sono state mangiate.
RE	impostato a 2 per impedire il movimento dal lago.
HH	impostato a 2 quando Aurora è sulla nave.
WW	impostato a 1 per la prova dell'Anguilla.
H	impostato a 2 quando Victor non vi può aiutare.

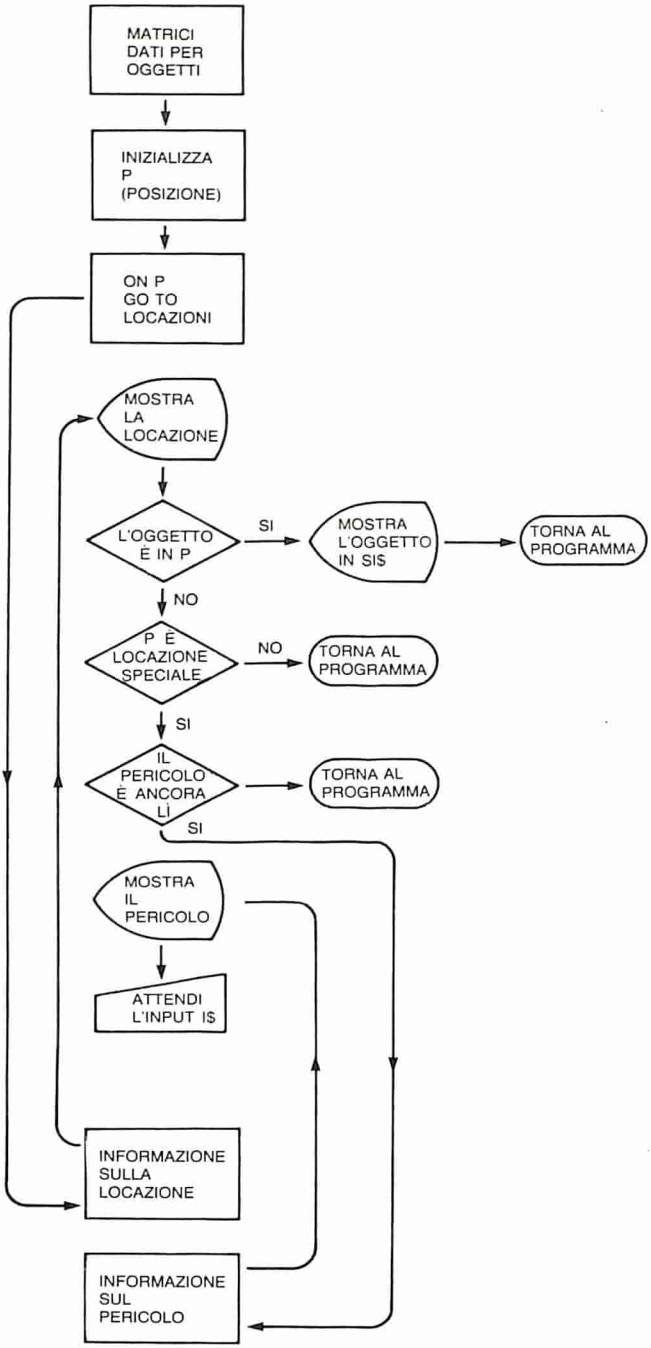
I valori che ho deciso per queste variabili sono alquanto arbitrari perché li ho creati lungo la strada, e a volte ho scelto 1, altre volte 2. La maggior parte delle variabili viene impostata a 0 all'inizio del programma (Linea 290 di nuovo), il che significa che la prima volta il pericolo sarà operativo.

Ci sembra di avere un problema.

Possiamo ora tornare alla tastiera e programmare di nuovo. Il primo obiettivo è quello di far apparire i pericoli sullo schermo quando il giocatore introduce la Locazione appropriata.

Il **Diagramma di Flusso 11.1** illustra il principio dei Moduli 9.2 e 11.1.

```
1 goto 3
2 save"@0:modulo11.1",8:verify"modulo11.
1",8:stop
3 rem
4170 rem**
4180 rem Punta ai Pericoli
4190 rem**
4200 ifp=24andsv<>2thengoto30000
4210 ifp=28andev<>2thengosub30500
4220 ifp=57andfv<>2andob%(6)<>-1thengoto
31000
4230 ifp=59andob%(2)<>-1thengosub32000
4240 ifp=59andpr<>2thengosub33240
4250 ifp=67andgv<>2thengoto32600
4260 ifp=74anddv<>2thengoto32700
4270 ifp=92andnv<>2andp=0thengoto33000
4280 ifp=92andnv<>2andp=1thengoto33060
29970 rem**
29980 rem serpente
29990 rem**
30000 print"■Fuori le rovine striscia un
 gigantesco serpente - il corpo ";
30010 print"grosso il doppio del vostro.
 Vi blocca L'unica uscita.":goto5000
30470 rem**
30480 rem anguilla
30490 rem**
30500 print"■E' molto profonda nell'acqu
a.":return
30970 rem**
30980 rem aria venefica
30990 rem**
31000 print"■All'improvviso vi sentite s
offocare. L'atmosfera ";
31010 print"e' velenosa. Se non farete n
ulla morrete presto..."
31030 form=1to5000:nextm
31040 print"■Cominciate a pensare a ";n$
:print"■*";
31050 forl=0to24:pokesc+l,0:nextl
31060 fori=0to30print"*";
```



Flowchart 11.1 Stampa di oggetti e pericoli

```
31070 pokevo,15:pokea1,41:pokes1,89:poke
w1,17:pokeh1,56:pkel1,99
31080 fort=1to200:nextt:pokevo,0:pokea1,
0:pokew1,0:pokes1,0
31090 nexti
31100 printchr$(13):>"■■Troppo tardi!!":
print"troppo tardi!!:goto190
31970 rem**
31980 rem cristallo
31990 rem**
32000 print"■■Notate che il cristallo e'
 sparito dalla teca energetica. ";
32010 print"Senza di esso la nave non pu
o' volare!! Deve averlo preso ";
32020 print"qualcuno! Dovete trovarlo pr
ima che scappi.
32030 return
32570 rem**
32580 rem sabbie mobili
32590 rem**
32600 print"■■State sprofondando nelle
sabbie mobili. in breve vi sono alle ";
32610 print"ginocchia.Bisogna far qualco
sa!!":q=0:goto5000
32670 rem**
32680 rem dinosauro
32690 rem**
32700 print"■Un colossale dinosauro esce
 dagli alberi e vi taglia la strada. ";
32710 print"Non potete passare.":goto500
0
32970 rem**
32980 rem indigeni
32990 rem**
33000 print"■una folla di indigeni ostil
i appare all'ingresso della ";
33010 print"capanna - bloccando la via.
Avanzano puntandovi le lance alla ";
33020 print"gola. Vogliono occuparsi di
";n$:q=0:goto5000
33030 rem**
33040 rem ancora alla capanna
33050 rem**
```

```
33060 print"■Indietro ancora!! Spero che
 abbiate pensato a come dare ad ";
33070 print"aurora il bacio che vi aiute
ra' a portarla via di qui!!"
33080 na=0:q=0:goto5000
33210 rem**
33220 rem niente aurora
33230 rem**
33240 print"■■Perduta aurora non potete
completare la missione.":return
```

Linee 4200-4280: Questa sezione del programma è dedicata alla parte che scriverà gli appropriati messaggi di errore. Quindi nella Linea 4200, se siete nella Locazione 24 (la città) e SV non è uguale a 2 (cioè non è la prima volta che si passa di lì, dato che l'abbiamo messa a 0 all'inizio del programma), allora andate alla linea 3000, che scriverà il messaggio "UN SERPENTE STA PER ACCHIAPPARVI" o qualcosa di simile.
Una volta che avete eliminato il serpente la variabile SV viene riportata a 2 – così se dovrete tornare alla Locazione 24 il serpente non apparirà più. Le altre condizioni, naturalmente, si riferiscono ad un'altra Locazione.

Linea 4220: È un esempio di condizione AND. Se lasciate la nave spaziale "AND" FV non è 2 (in altre parole, se non avete mangiato le noci) AND OB (6) non è – 1 (in altre parole non avete la tuta spaziale nell'inventario) allora arriverete alla parte del programma che dirà "ATMOSFERA VELENOSA".

Le Linee 4230 e 4240 si riferiscono entrambe ad Aurora ed al cristallo – una parte alquanto complicata di programmazione a cui tornerò nel Modulo 13.1.

Linee 4270-4280: Se siete nella capanna degli indigeni per la prima volta, allora NV = 0 così gli indigeni vi minacceranno e vi impediranno di prendere Aurora. Dopo 10 tentativi vi viene permesso di partire (magari per cercare le noci) ma adesso NP è impostato a 1, cosicché quando tornate il messaggio sarà "INDIETRO ANCORA" invece del messaggio della prima volta che si arriva.

Quest'ultima può sembrare una raffinatezza – ma appare molto meglio quando state veramente giocando. Infatti quando abbellite il vostro gioco personale, spesso avete bisogno di giocare veramente, per trovare questi piccoli tocchi che lo migliorano.

Per esempio, originariamente avevo un messaggio "NON LO VEDO QUI" in corrispondenza a comandi come "LANCIA" se non implicava un oggetto nella matrice degli oggetti. Quindi se il giocatore batteva "LANCIA CAVALLO" prima veniva mandato il messaggio "NON LO VEDO QUI". Questo era buono finché il giocatore cercava di gettare cose come cavalli o Ford Cortina (in altre parole, cose da pazzi), ma in certe Locazioni parliamo di "sassi o vasellame", che non sono validi oggetti da buttare. Quindi se battete LANCIA VASELLAME e avete la risposta "non ne vedo", tutto ciò suona male. La risposta è cambiare da "NON LO VEDO QUI" a "SMETTETELA DI DIRE SCEMENZE", che risulta valida per qualsiasi oggetto, in entrambe le situazioni.
Sento a questo punto il bisogno di chiedere scusa per il fatto che, pur scrivendo e

riscrivendo, questa parte del libro può sembrare confusa per qualcuno. Questa sezione del programma richiede una conoscenza dell'Adventure che è più facile per me (l'autore della trama), dal momento che ho vissuto e respirato l'atmosfera del racconto per dei mesi, ed ho perduto del tempo per trattare questa parte del programma. Se vi sentite un po' frastornati dalla complessità della trama e dei moduli relativi, potrà consolarvi il fatto che tratterò ciascun pericolo in moduli separati, il che renderà più chiara la situazione. Per adesso fate affidamento su ciò, e riguardate piuttosto al fatto che questa parte del programma creerà le istruzioni PRINT che manderanno i messaggi sullo schermo, e poi impedirà al giocatore di fuggire con il semplice spostarsi fuori dalla locazione.
Quando creerete la vostra Adventure, tutto diventerà molto più facile, dal momento che avrete definito voi la trama ed i pericoli, ed avrete maggiore familiarità con essi.

Linee 30000-30010: Messaggio riguardante il serpente.

Linea 30050: Questa linea si riferisce al solo cristallo. Per il momento abbandono l'anguilla, perché richiederà un intero capitolo tutto per sé.

Linee 31000-31100: Per lasciare il pericolo costituito dall'atmosfera velenosa, ho aggiunto dopo un po' una breve routine che simula il tempo che trascorre prima che siate presumibilmente soffocati a morte. Per prima cosa appare il messaggio "STATE SOFFOCANDO..." ed il programma ha una pausa (Linea 31030) per darvi il tempo di leggerla. Quindi appare un altro messaggio in color porpora: "COMINCIATE A PREOCCUPARVI..." ed un asterisco viene stampato nella riga sottostante (Linea 31040).

La linea 31050 azzera tutte le variabili relative al suono, poi comincia un ciclo di lunghezza 30, nella Linea 31060, contenente un effetto sonoro (Linee 31070 e 31080) al suo interno.
L'effetto è un beep di alta frequenza ogni volta che un '*' appare sullo schermo, e un asterisco comincia a spostarsi, aggiungendo altri '*' fino a 30. Poi appare il messaggio "TROPPO TARDI – TROPPO TARDI", e il programma va alla sezione SIETE MORTO (Modulo 11.3 – da aggiungere ancora).

Non c'è modo di sfuggire a questo pericolo. Una volta incappati nell'atmosfera venefica siete condannati. Naturalmente questa routine viene attivata anche se aveste preso la tuta spaziale, uscendo la prima volta, ma l'abbiate levata più tardi (per baciare Aurora, per. es.), senza prima mangiare le noci. Ma farò queste aggiunte al programma più tardi.

Linee 32000-32020: Messaggio standard per l'interno dell'astronave, se non avete trovato il cristallo.

Linee 32600-32610: Il messaggio per le sabbie mobili. Notate qui Q=0, che setta un contatore. In breve questo vi permette un certo numero di tentativi di fuga, prima che il gioco sia concluso col messaggio "SIETE AFFOGATO".

Linee 32700-32710: Messaggio per il dinosauro. La differenza tra questa sezione e le altre è che tutto quello che il dinosauro fa è bloccarvi la strada verso sud – senza in effetti uccidervi.

Linee 33000-33080: Queste due sezioni trattano degli indigeni – la prima volta quando

entrate nella Locazione 92, la seconda se non riuscite a liberare Aurora al primo tentativo e a tornare con lei. La variabile NA impedisce la fuga dalla capanna finché non abbiate tentato 10 volte di liberare Aurora (notate ancora Q=0).
Linee 33240: l'ultima linea del modulo stampa semplicemente il messaggio della Locazione 59 (l'interno dell'astronava) se non avete liberato Aurora.

Prova del Modulo 11.1

Per testare questo modulo dovete dare il RUN al programma e cominicare a muovervi tra le varie locazioni. La prima cosa che noterete è che partendo col gioco ancora dalla prima locazione (la cabina della vostra astronave) vedrete un'intera pagina di testo, avendo aggiunto la sezione relativa al cristallo e ad Aurora. NON prendete la mitraglietta o la tuta spaziale, ma andate direttamente ad ovest. Uscendo quindi dall'astronave dovreste trovare l'atmosfera velenosa, che vi avvelenerà... mentre appaiono gli asterischi e voi osservate senza poter intervenire, dal momento che il vostro destino è segnato. Terminando questa sezione, dovrebbe apparire il messaggio di errore UNDEF'D STATEMENT ERROR IN 31100.
Ridate ora il RUN, e prendete questa volta la tuta, girando poi per il pianeta. Arrivando alle locazioni apposite, dovrebbero essere mostrati i pericoli. Per il momento potete fuggire battendo EST o OVEST (o qualsiasi altra cosa) per spostarvi nella sezione seguente. Dobbiamo tuttavia impedire che ciò accada nel gioco reale e questo sarà il compito del prossimo modulo.

Nessuna uscita per quella via

Avete notato come sia stato facile sfuggire ai pericoli, quando avete testato il Modulo 11.1 – come possiamo impedirlo nel nostro gioco? Molto semplicemente abbiamo bisogno di una routine tra l'input ISTRUZIONI e la routine MOVIMENTO. Sarà proprio il Modulo 11.2

MODULO 11.2

```
1 goto 3
2 save"@0:modulo11.2",8:verify"modulo11.
2",8:stop
3 rem
5970 rem**
5980 rem nessuna uscita
5990 rem**
6000 ifp=24andsv<>2then6110
6010 ifp=28andes=2then6500
6020 ifp=28andev<>2then6110
6030 ifp=67and9v<>2then6110
6040 ifp=74anddv<>2then6170
6050 ifp=92andnv<>2anda=1then6500
6060 ifp=92andnv<>2anda=0then6110
```

```
6070 goto6500
6110 ifi$="n"ori$="e"ori$="s"ori$="o"the
ngosub19000:goto6130
6120 goto6150
6130 print"Non Potete fuggire di qui!":
print"Dovete Pensare a ";
6140 print"qualcos'altro!":goto5000
6150 ifi$="corri"thengosub19000:print"N
on c'e' uscita di li'.":goto5000
6160 ifi$="cammina"thengosub19000:print"
Non c'e' uscita di li'.":goto5000
6170 ifi$="sud"ori$="s"thengosub19000:Pr
int"Niente da fare ";n$
6180 ifi$="sud"ori$="s"thenPrint"C'e' u
n dinosauro sulla via!":goto5000
```

Linee 6000-6070: queste linee controllano se voi siete in una delle locazioni speciali, e se è così il programma sarà indirizzato alla routine IMPEDIRE L'USCITA. Se non ci siete allora la Linea 6070 bypassa quella routine.

Le Linee 6010 e 6050 differiscono di poco, in quanto entrambi alla fine conducono all'area NESSUN MOVIMENTO. Nel caso della 6010 il motivo è che ES=2 (significa che è la seconda volta che visitate quella locazione). Questo diventerà più chiaro nella sezione del pericolo "anguilla". La linea 6050 differisce per il fatto che la variabile NA è stata posta a 1 (avete visitato solo una volta quella locazione). Penso che sia meglio non porre limiti al numero di volte che si torna a visitare la capanna degli indigeni, per liberare Aurora, specialmente se ciò è dovuto al fatto che avete letteralmente 'dimenticato' di battere "PRENDI AURORA", dopo che siete riusciti a liberarla, per cui dovete tornare indietro a riprenderla.

Linee 6110-6180: sono semplicemente le diverse risposte ai vari tentativi di sfuggire dalla locazione battendo CORRI e CAMMINA. Notate anche le Linee 6170-6180 impediscono un movimento verso sud per affrontare e superare il dinosauro – mentre è ancora possibile andare a nord.

Linee 190-240: infine questo piccolo modulo costituisce il messaggio terminale nel caso si vada incontro ad una soffocante fine, in qualsiasi posto lungo la via.

MODULO 11.3

```
1 goto 3
2 save"@0:modulo11.3",8:verify"modulo11.
3",8:stop
3 rem
160 rem**
170 rem siete morto
180 rem**
190 Pokev+21,0:Print"■■■Lo avete supera
to bene,stavolta, vero ";n$
200 Print"■■■Per fortuna Posso farvi riv
ivere e rimettervi";
210 Print" al via.":Print"■■■Volete ? (s
/n)"
220 getr$:ifr$<>"s"andr$<>"n"then220
230 ifr$="s"then280
240 ifr$="n"then17570
```

Prova dei Moduli 11.2 e 11.3

Se usate la medesima procedura di test del Modulo 11.1, troverete che non potete fuggire a lungo, poiché arriverete alla fine ad una locazione con un pericolo. Battendo O, per esempio, dovrebbe comparire il messaggio "NON POTETE FUGGIRE PER QUELLA VIA – DOVETE PENSARE A QUALCOSA D'ALTRO", (seguito da un burp volgare!). Anche se dimenticate la vostra tuta spaziale, non vi si dichiara semplicemente morti, ma vi si offre la possibilità di ricominciare a giocare.
Essendo stati ora illustrati la maggior parte degli elementi del gioco, il **Diagramma di Flusso 11.2** illustra la struttura di base di un'Adventure.

Sommario

Come ho dichiarato prima, questa parte del libro può sembrare probabilmente più complicata di quanto è in realtà, ad una prima lettura. Ho tentato di renderla la più semplice possibile, ma in realtà solo il sapere ciò che ci aspetta, può essere di vero aiuto, e quindi è meglio programmare questa sezione e andare avanti, tornando per ulteriori spiegazioni su ciò che sta accadendo, linea per linea, solo quando avrete un'idea migliore di come ho affrontato i vari rischi.
È anche riguardo a questa parte di programmazione, che dovreste diventare consapevoli del formidabile potere che avete (come programmatore). Forse non dovrei ripetermi ancora, ma non si può negare cheVOI controllate il gioco, VOI decidete dove il giocatore può andare, dove può spostarsi, e cosa può fare. Non importa come la pensiate voi, ma io trovo tutto ciò molto affascinante.

MODULO 11.3

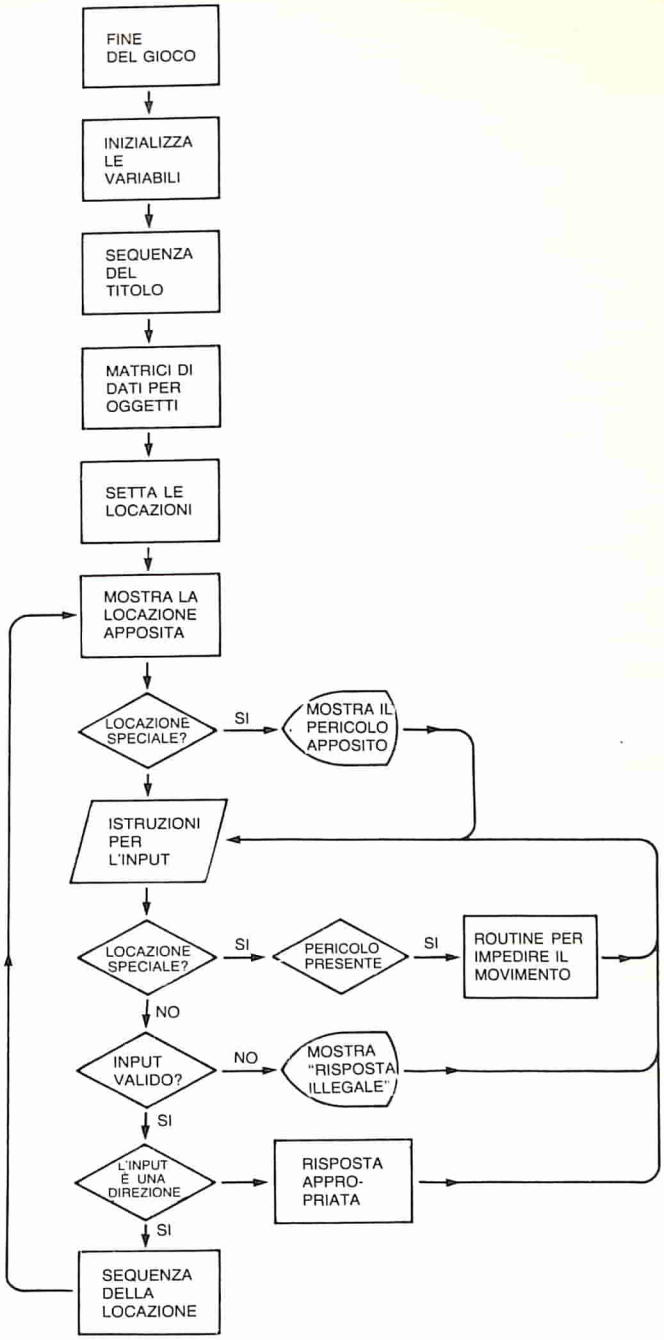
```
1 goto 3
2 save"@0:modulo11.3",8:verify"modulo11.
3",8:stop
3 rem
160 rem**
170 rem siete morto
180 rem**
190 Pokev+21,0:Print"■■■Lo avete supera
to bene,stavolta, vero ";n$
200 Print"■■■Per fortuna Posso farvi riv
ivere e rimettervi";
210 Print" al via.":Print"■■■Volete ? (s
/n)"
220 getr$:ifr$<>"s"andr$<>"n"then220
230 ifr$="s"then280
240 ifr$="n"then17570
```

Prova dei Moduli 11.2 e 11.3

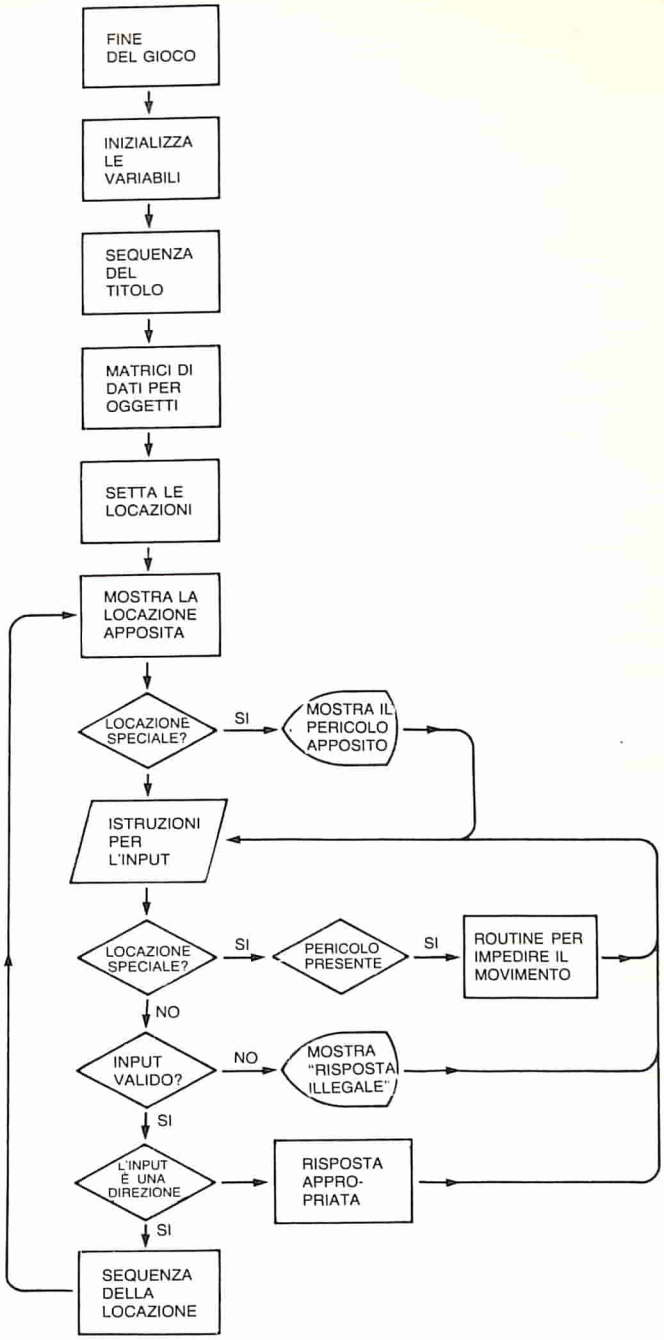
Se usate la medesima procedura di test del Modulo 11.1, troverete che non potete fuggire a lungo, poiché arriverete alla fine ad una locazione con un pericolo. Battendo O, per esempio, dovrebbe comparire il messaggio "NON POTETE FUGGIRE PER QUELLA VIA – DOVETE PENSARE A QUALCOSA D'ALTRO", (seguito da un burp volgare!). Anche se dimenticate la vostra tuta spaziale, non vi si dichiara semplicemente morti, ma vi si offre la possibilità di ricominciare a giocare.
Essendo stati ora illustrati la maggior parte degli elementi del gioco, il **Diagramma di Flusso 11.2** illustra la struttura di base di un'Adventure.

Sommario

Come ho dichiarato prima, questa parte del libro può sembrare probabilmente più complicata di quanto è in realtà, ad una prima lettura. Ho tentato di renderla la più semplice possibile, ma in realtà solo il sapere ciò che ci aspetta, può essere di vero aiuto, e quindi è meglio programmare questa sezione e andare avanti, tornando per ulteriori spiegazioni su ciò che sta accadendo, linea per linea, solo quando avrete un'idea migliore di come ho affrontato i vari rischi.
È anche riguardo a questa parte di programmazione, che dovreste diventare consapevoli del formidabile potere che avete (come programmatore). Forse non dovrei ripetermi ancora, ma non si può negare cheVOI controllate il gioco, VOI decidete dove il giocatore può andare, dove può spostarsi, e cosa può fare. Non importa come la pensiate voi, ma io trovo tutto ciò molto affascinante.



Flowchart 11.2 Adventure di base



Flowchart 11.2 Adventure di base

CAPITOLO 12

BELLA GIORNATA PER UN TUFFO

In ogni programma di Adventure degno di tale nome vi troverete a trattare con grossi moduli di programma, e se usate il BASIC allora, quando l'intreccio si fa serrato, vi può capitare di andare fuori memoria tra i moduli (perché sorgeranno dei problemi non previsti). Questo fatto richiede una rinumerazione giudiziosa, ma anche se avete delle utilities potete ugualmente avere dei problemi.
Avevo trovato che ciò stava succedendo a me, perciò ho escogitato una carta speciale (vedi **Figura 12.1**) che mi mette in grado di dare una stima dell'inizio della numerazione di ogni specifica routine o modulo.
È anche essenziale quando si crea un programma con un grosso numero di GOTO e GOSUB, poter mettere i numeri di linea man mano che si procede, altrimenti li dimenticherete sicuramente – con un supplmento di lavoro nel correre dietro ai vostri errori.
Inoltre vi consiglio di sovrastimare, anche più di quanto immaginate, la vostra numerazione, senza tener troppo conto dello spazio che lasciate. Per darvi un esempio di come ho usato la carta, la Figura 12.2 vi mostra dove siamo arrivati nel programma.

Uccidetemi delicatamente

Uno dei comandi sempre presenti negli Adventure deve essere UCCIDI (non consideriamo le implicazioni psicologiche...), quindi prima di entrare in una delle sequenze più dure del programma – il pericolo costituito dall'anguilla nel lago – dobbiamo entrare nel modulo UCCIDI.

MODULO 12.1

```
1 goto 3
2 save"@0:m12.1",8:verify"m12.1",8:stop
3 rem
12270 rem**
12280 rem uccidi
12290 rem**
12300 ifno$="aurora"thengosub19000:print
```

(segue a pag. 145)

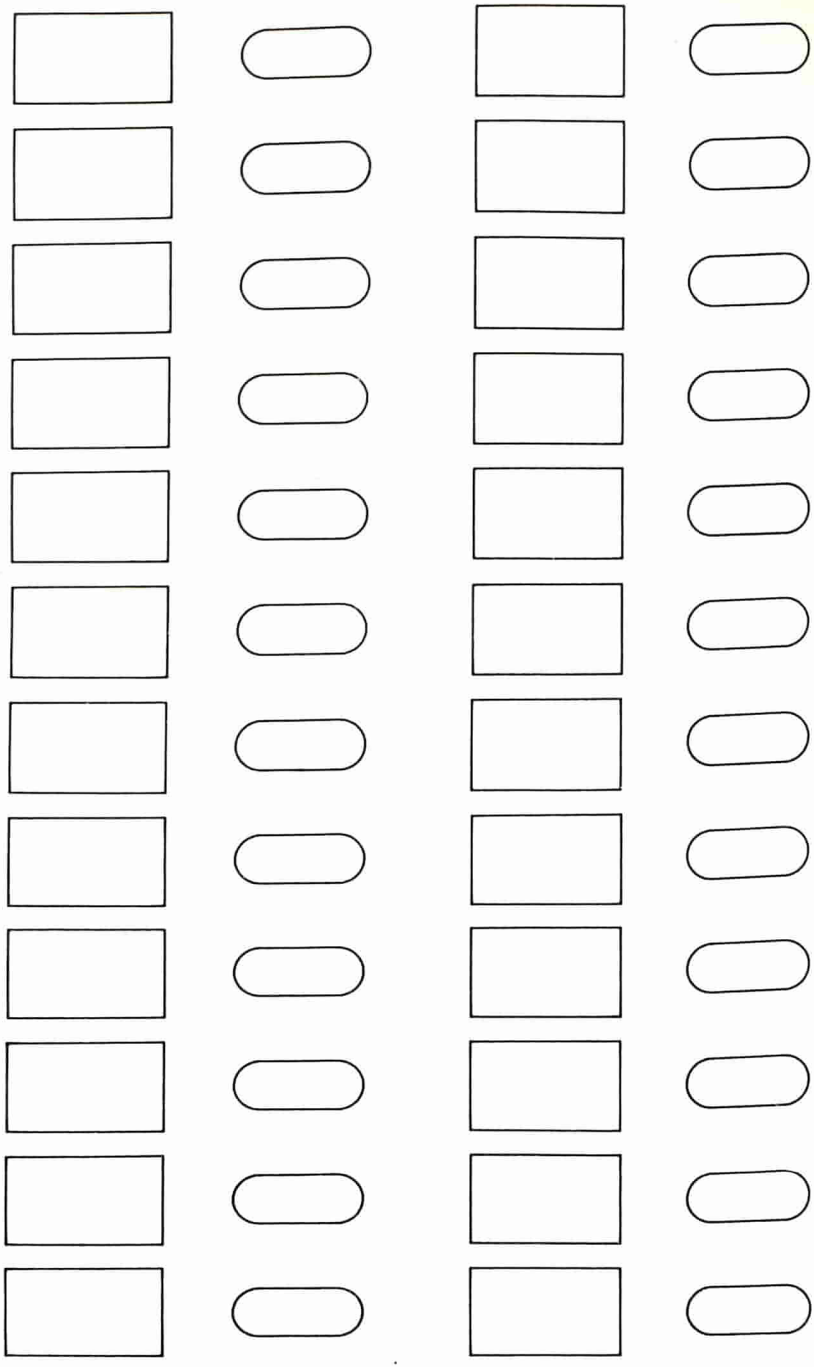


Figura 12.1 Carta per pianificazione moduli

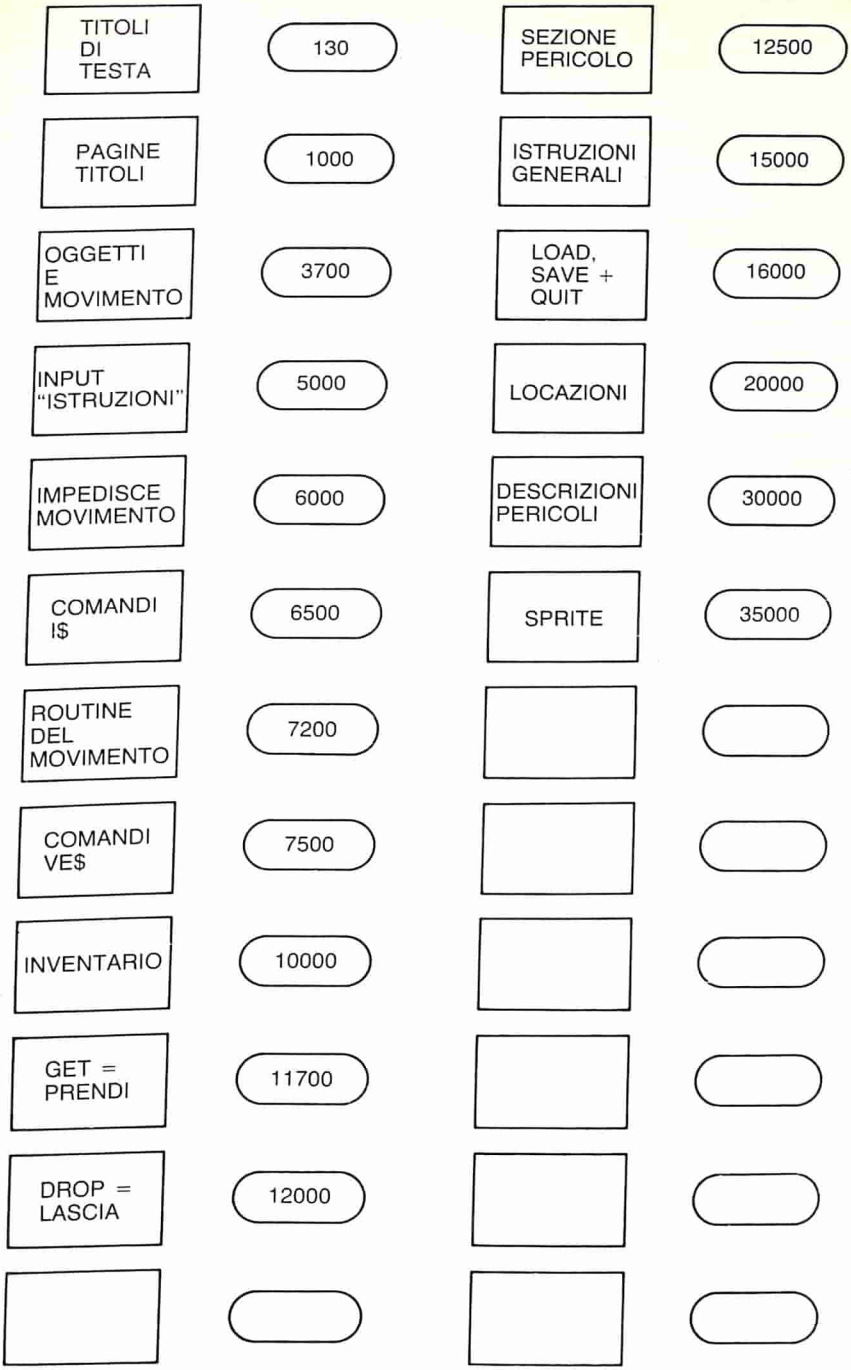
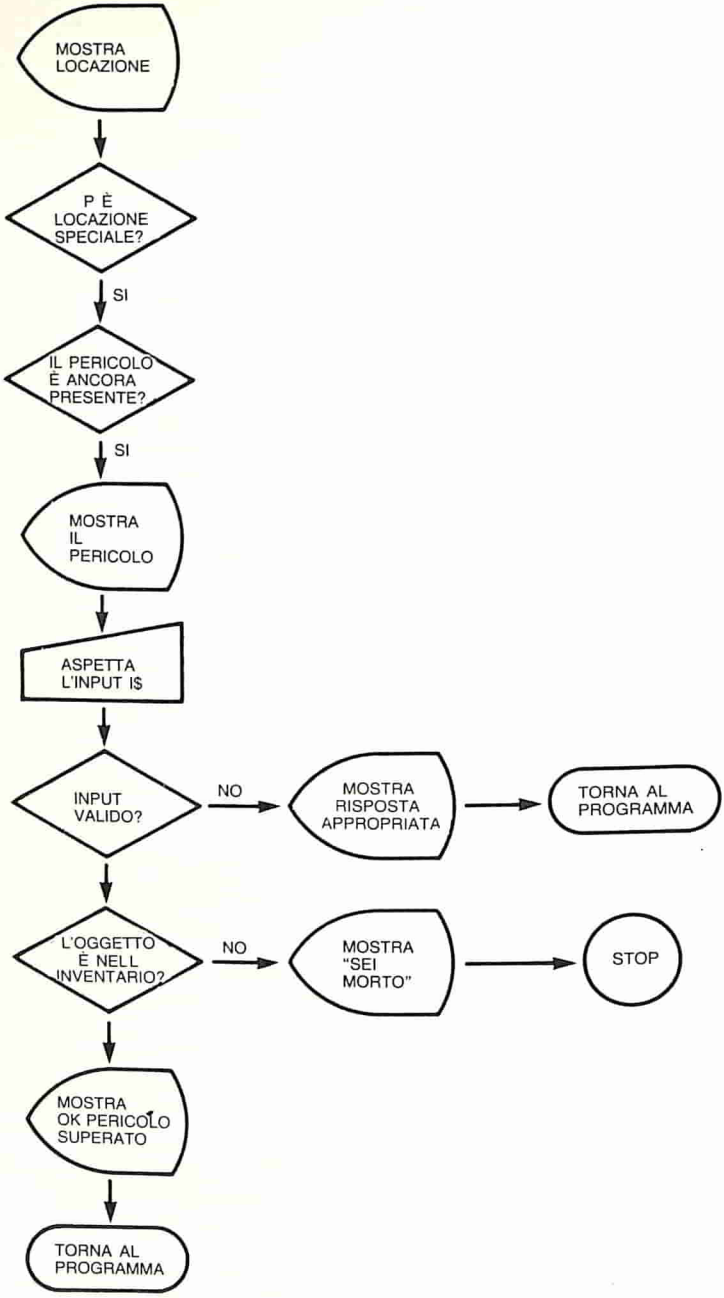


Figura 12.2 Numeri di inizio delle subroutines per Il pianeta dell'incubo



Flowchart 12.1 La sfida col pericolo

(mod. 12.1 seguito)

```
"■■State scherzando !!!":goto190
12305 ifP=24then13000
12310 ifP=28then12500
12330 ifP=74then13500
12340 ifP=92then13700
12350 gosub18000:print"■■Non vedo Proprio
 niente da ":ve$:goto5000
```

Linea 12300: Mi è capitato che un mattacchione scrivesse "UCCIDI AURORA" solo per Dopo la risposta "STAI SCHERZANDO" il programma va alla subroutine alla Linea 190 (che è la sezione 'Sei morto!') e termina il gioco.

Linea 12305: Questa linea serve nel caso che qualcuno batte "UCCIDI VICTOR" (come se si potesse!), e manderà il programma alla sezione di AIUTO dove una simpatica sorpresa attende il giocatore (vedi Modulo 14.1).

Linee 12310-12340: Queste linee fanno riferimento alle sole Locazioni in cui potete provare a uccidere qualcosa, e manda il programma alla sezione appropriata (che aggiungeremo più tardi).

Linea 12350: Una risposta standard nel caso cerchiate di uccidere qualcuno in ogni altro posto lungo la strada.

L'anguilla batte in ritirata

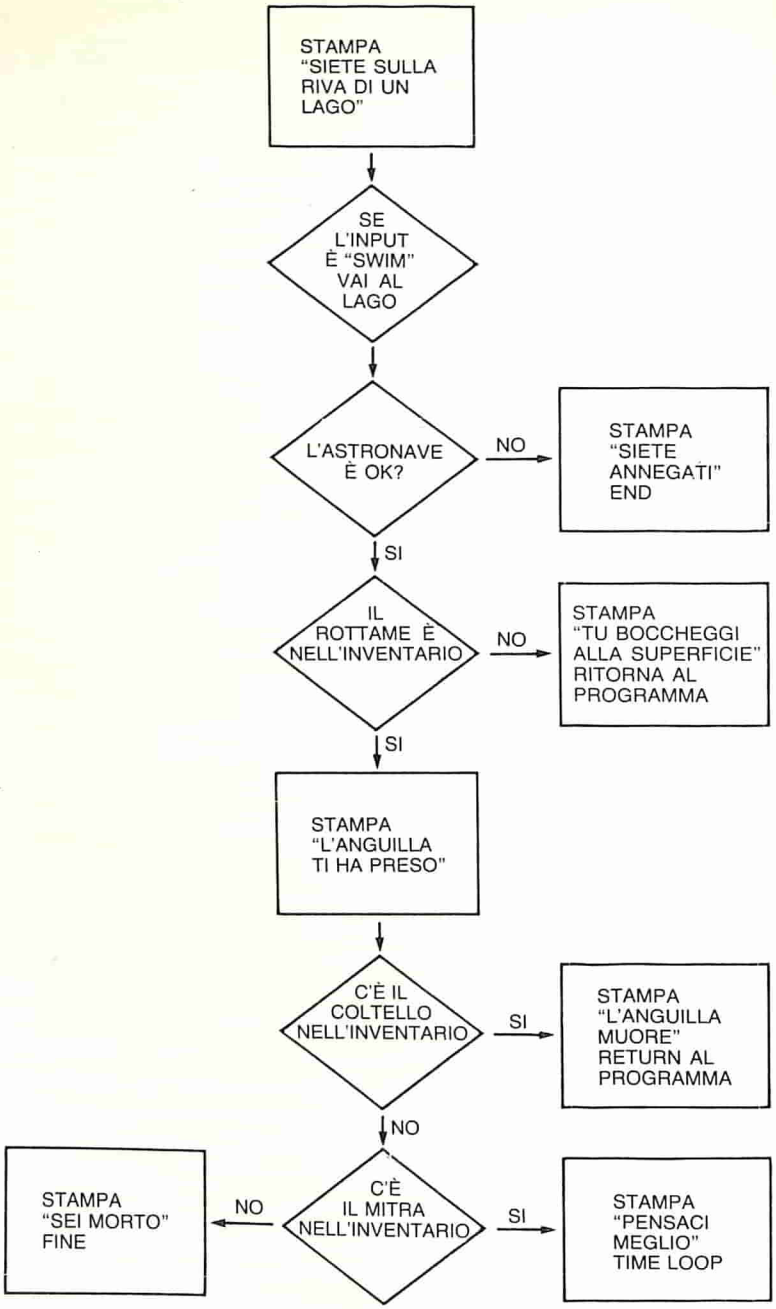
Mi scuso per il titolo di questa sezione – licenza poetica, suppongo.

Ora possiamo cominciare ad aggiungere tutte quelle perfidie che renderanno la vita più interessante per il nostro giocatore. Sono partito con la parte forse più dura dal punto di vista della programmazione – così da qui in poi è tutta discesa.

Quando dico la più dura – non intendo ovviamente riferirmi al fatto di copiare le linee. Ma progettare e pianificare questa sezione è stata per me la parte più intricata – e magari la più difficile da capire per voi, a causa del numero di variazioni possibili nelle risposte che il giocatore può dare (nel tentativo di superare il pericolo che gli abbiamo preparato) e nel risultante numero di possibilità che dobbiamo programmare.

Infatti si sta cercando qui di anticipare tutte le parole che l'Adventurer userà nel suo tentativo di fuggire, e ciò vi darà le maggiori preoccupazioni, nel cercare di farli rientrare tutti nella vostra Adventure. Qualunque cosa succeda, voi non tratterete mai tutte le eventualità, quindi è meglio accettare questo fatto fin dall'inizio.

Vediamo come ho progettato il trattamento di ogni possibile pericolo nel **Flow-Chart 12.1.**



Flowchart 12.2 Il pericolo dell'anguilla

Probabilmente la maniera più facile per capirlo è quella di percorrere il flowchart ponendo domande al computer o dandogli dei comandi – come segue:

DOMANDA 1: La Locazione è speciale?
Cioè si riferisce alle Locazioni 24, 28, 57, 67, 74 e 92 ed è preso in considerazione dal Modulo 11.1.

DOMANDA 2: Il pericolo è ancora presente?
Le Variabili SV, EV, DV ecc., (Modulo 11.1 di nuovo) controllano ciò, col loro valore. Se sono impostate a 0 il pericolo è presente. Se sono impostate a 2 il pericolo non c'è. A questo stadio il pericolo non sarà mostrato sullo schermo.

DOMANDA 3: L'input è valido?
Il programma adesso controlla se l'input I$ è valido. Se no, viene data una risposta appropriata. In effetti, ciò che si riferisce a errori, scritture senza spazi, ecc., viene considerato nel Modulo 10.2.

DOMANDA 4; L'oggetto è nell'inventario?
Questa non è proprio tutta la verità, dato che potete sfuggire ad alcuni pericoli senza un oggetto nell'inventario (per es. il dinosauro). Tuttavia il problema è principalmente quello di uccidere i mostri e perciò avete bisogno di una mitraglietta, o di un coltello, quindi la prima cosa che il programma fa è controllare che abbiate l'oggetto appropriato. Se avete l'arma giusta, allora il programma dice "okay" e siete salvi (e la Variabile è impostata a 2 per tenerne conto la prossima volta che passate dalla stessa Locazione). Se non avete l'oggetto giusto potreste scoprire di essere morto!
Naturalmente il flowchart è semplificato, ma delinea la strategia di base che ho usato per occuparmi della maggior parte dei pericoli.
Il **Flowchart 12.2** illustra questi principi nella Locazione 28.

Riguardo il nuotare

MODULO 12.2

```
1 goto 3
2 save"@0:m12.2",8:verify"m12.2",8:stop
3 rem
6590 ifi$="g"orleft$(i$,6)="guarda"theng
osub18000:goto6610
6600 goto6640
6610 ifp=28andgg=1thenprint"Mmon vedete
molto sotto acqua!!":goto5000
6620 goto4020
6710 ifi$="vai lago"andp=28andrc=2then30
650
6720 ifleft$(i$,5)="nuota"andp=28andrc=2
then30650
6730 ifleft$(i$,5)="tuffa"andp=28andrc=2
```

```
then30650
6740 ifi$="vai lago"andP=28then30510
6750 ifi$="vai riva"andP<>28then7810
6760 ifi$="vai riva"andP=28then6780
6770 goto6800
6780 ifes=2thengosub18000:goto20540
6790 ifes<>2thengosub19000:goto6130
6800 ifleft$(i$,5)="nuota"andP=28then305
10
6810 ifleft$(i$,5)="tuffa"andP=28then305
10
6820 ifleft$(i$,3)="vai"thengosub18000:p
rint"Quale direzione ?":goto4320
6830 ifleft$(i$,6)="vaffan"thengosub1900
0:print"E voi ";n$:goto5000
```

Quando arriviamo la prima volta alla Locazione 28, all'inizio non ci troviamo effettivamente nel lago. Perciò ci serve un qualche modo per entrare veramente nell'acqua. Ho messo alla prova diverse persone per vedere cosa potessero scrivere, e le risposte più frequenti sono state NUOTA, VAI AL LAGO o TUFFATI DENTRO. Questo Modulo si occupa di tutti questi comandi.

Linee 6590-6620: Queste linee possono confondervi un po' perché sono messe un po' fuori dal contesto. Si riferiscono al comando GUARDA. La prima Linea 6590 è il comando I$. Poi la Linea 6600 suppone che abbiate superato questa sezione del programma (per andare in un'altra area che non abbiamo ancora messo).

Linea 6610: Si riferisce alla risposta "GUARDA" se siete nella Locazione 28 è '*sotto*' l'acqua (da qui la variabile WW = 1 che viene impostata una volta che entrate nell'acqua). La Linea 6620 è la risposta standard al comando "GUARDA" e porta il programma alla routine di movimento, cosicché verrà scritto "SEI NEL DESERTO" o in qualunque altra parte.
Lasciatemi divagare ancora un po' su questo argomento. A volte nel programma potete trovare che state battendo un certo numero di comandi (come "ARRAMPICATI SULL'ALBERO" o "ESAMINA I SASSI" e le risposte fanno scorrere in su lo schermo in modo che la vostra Locazione sparisce. Quando decidete di muovervi potete aver dimenticato dove siete o quale direzione dovete prendere, quindi battendo "GUARDA" vi verrà ripristinato il testo che ve lo dice.
Ora nella Locazione 28 abbiamo un problema speciale – una parte di questa Locazione è il lago (dove siamo alle prese con l'anguilla). Sarebbe stupido che battendo "GUARDA" venisse fuori di nuovo "SEI SULLA SPONDA DEL LAGO".
Le Linee 6610 e 6620 prevengono questo fatto.
Prima di andare alla prossima sezione guardiamo un attimo di nuovo le variabili implicate nella Locazione 28 e nel pericolo dell'Anguilla.

Tavola 12.1: Variabili per il pericolo "anguilla" nella Locazione 28.

CV impostata a 2 quando viene trovato il Cristallo.
EV impostata a 2 quando l'Anguilla è stata uccisa.
ES impostata a 2 quando l'anguilla è stata uccisa; impostata a 1 quando l'Anguilla vi ha bloccato.
RE impostata a 2 per fermare il movimento del nuoto dalla Locazione 28.
WW impostata a 1 quando l'anguilla vi ha catturato, per bloccare la risposta ad un GUARDA.

Linee 6710-6730: Queste linee si applicano quando RE = 2 (in altre parole l'anguilla vi ha preso) e voi cercate di sfuggire battendo NUOTA o TUFFATI. Ci sarà una risposta "LA BESTIA VI TIENE BEN STRETTI PER IMPEDIRVI DI FUGGIRE" (nel Modulo 12.4 – ancora da fare).

Linee 6740-6800 e 6810: Queste sono le linee che vi portano nel lago. Quindi se siete nella Locazione 28 e battete uno di questi tre comandi (NUOTA, TUFFATI o VAI LAGO) andrete al Modulo 12.4.

Linea 6750; Questa linea è per quelli che battono "VAI RIVA" quando non sono nella Locazione 28 (è sicuro che qualcuno lo farà). Manda il programma a una risposta "TI STAI COMPORTANDO IN MODO UN PO' STRANO".

Linee 6760-6770: Se siete nel lago e battete "VAI RIVA" allora dobbiamo controllare diverse cose prima di permettervi di scappare – altrimenti qualcuno potrebbe sfuggire dalle spire dell'anguilla dicendo semplicemente "VAI RIVA", e non possiamo permetterlo! La Linea 6770 è per comandi che non si riferiscono al lago e all'anguilla e bypassa le prossime poche linee.

Linee 6780-6790: Bene, siete nel lago e battete "VAI RIVA". Per prima cosa controlliamo che ES = 2 (ES è messo a 2 dopo che l'anguilla è stata 'uccisa' con successo). Se ES = 2 allora andiamo alla Linea 20540 che è l'inizio della descrizione della Locazione 28 – in altre parole la scritta "SIETE SULLA SPONDA DEL LAGO" appare di nuovo.

Se ES non è 2 allora siete ancora nelle spire dell'anguilla – quindi il programma torna indietro a 6130 (che è la routine per impedire il movimento nel Modulo 11.2).

Linea 6820: Questa linea non si riferisce propriamente a questa Locazione. È una semplice routine nel caso che il giocatore batta VAI in qualunque momento. Notate come ho usato LEFT$ (I,3) = "VAI" perché in questo modo vengono usati anche comandi come "VAI OVEST" o "VAI EST".

Linea 6830: Se vi sentite particolarmente frustrati potreste essere tentati di scrivere "VAFFANBAGNO" e questa linea aggiunge un tocco di umorismo.

MODULO 12.3

```
1 goto 3
2 save"@0:m12.3",8:verify"m12.3",8:stop
3 rem
7520 ifve$="prendi"andp=28then32070
7530 ifve$="afferra"andp=28then32070
7540 ifve$="acchiappa"andp=28then32070
7640 ifve$="uccidi"orve$="sfascia"orve$=
"distruggi"then12300
32040 rem**
32050 rem non potete prendere cristallo
32060 rem**
32070 ifno$="cristallo"then32090
32080 goto11700
32090 ifev<>2thengosub19000:print"[illegible]non p
otete...";:form=1to500:nextm
32100 ifev<>2thenprint"...ancora!!":prin
t"E' [illegible]sotto[illegible] acqua.":goto4320
32110 ifve=2then11700
```

Supponete di arrivare alla Locazione 28, vedete il Cristallo, e battete "PRENDI CRISTALLO". In questo momento non c'è nulla che impedirà al programma di dire "OKAY", e il cristallo è nell'inventario. Questo Modulo previene ciò.

Linee 7520-7550: Semplici comandi concernenti il comando "PRENDI" quando siete nella Locazione 28. Per assicurarvi di non ricevere una risposta scorretta il 64 bypasserà il solito modulo "PRENDI" e manderà il programma alla Linea 32070.

·Linea 7640: Questa è la linea per il comando UCCIDI ecc., che vi porterà al Modulo 12.1 (che abbiamo introdotto all'inizio del Capitolo).

Linee 32070-32080: La Linea 32070 controlla se NO$ è "CRISTALLO" Se eravate nella Locazione 28 e avete battuto "PRENDI" ma non avete battuto "CRISTALLO" (per esempio potreste aver lasciato cadere qualcosa alla visita precedente e ora vorreste riprenderlo), allora la Linea 32080 manderà il 64 alla sezione appropriata del programma.

Linee 32090-32100: Queste due linee vi impediscono semplicemente di "prendere" il cristallo se EV non è impostato a 2 (l'anguilla è ancora viva).

Linea 32110: Se l'anguilla è morta (EV = 2) allora siete rispediti indietro alla routine PRENDI nel Modulo 10.3 e potete "prendere" il cristallo con successo.

L'anguilla vi ha preso.

Possiamo supporre che il nostro giocatore abbia trovato il modo di entrare nel lago per provare a riprendere il cristallo. Ci sono varie situazioni che influiscono sul modo di trattare il prossimo stadio.
Le possibilità sono:
1. Non indossate la tuta spaziale.
2. Indossate la tuta spaziale ma non avete ancora raccolto il rottame.
3. Indossate la tuta spaziale ed avete il rottame.

MODULO 12.4

```
1 goto 3
2 save"@0:m12.4",8:verify"m12.4",8:stop
3 rem
12020 if p=28 and no$="tuta-spaziale" then go
sub 19000:goto 30510
12050 if p=28 and no$="tuta-spaziale" then go
sub 19000:print"▒State affondando":goto 30
540
30510 if ob%(6)<>-1 then goto 30540
30515 if ve=2 then gosub 18000:print"bel gio
rno per un tuffo! Cosa ora?":goto 4320
30520 if ob%(4)<>-1 then goto 30560
30530 goto 30590
30540 gosub 19000:print"Non avete piu' l'
astronave, vero ";n$
30550 print"Maledizione!! Siete annegato
!!":form=1 to 2000:next m:goto 190
30560 gosub 18000:print"▒Galleggiate come
 un sughero. Vi occorre qualcosa ";
30570 print"di pesante per tuffarvi   ";
30580 print"giu' per prendere il cristal
lo.":goto 5000
30590 if ob%(2)=-1 then gosub 18000:print"▒B
ella giornata per un tuffo!!!":goto 4320
30610 es=1:print"▒All'inizio scendete fi
no a prendere il cristallo. ";
30620 print"All'improvviso esce dal prof
ondo una gigantesca anguilla ";
30630 print"e vi si attorciglia intorno.
"
30640 print"Vi sta stritolando...":gosub
```

```
19000:rc=2:q=0:yy=1:goto5000
30650 gosub18000:print"[illegible]L'animale vi h
a stretto cosi' forte che ";
30660 print"non potete fuggire !":goto50
00
```

Linee 12020-12050: Queste due linee si riferiscono all'improbabile evento che vi togliate la tuta spaziale mentre siete nel lago. Entrambe vi rimandano alla Linea 35040. Esse vi introdurranno alla sezione BUTTA del programma.
Linee 30510-30530: Queste linee esaminano l'inventario per controllare se state indossando la tuta spaziale, se avete sconfitto l'anguilla, o se state portando il rottame. La Linea 30530 vi porterà oltre le prossime poche linee alla sezione dell'anguilla.
Linea: 30540-30550: Niente tuta spaziale – affogate. C'è una piccola pausa e uscite per giungere alla Linea 190.
Linee 30560-30580: Se non avete il rottame allora 'boccheggiate in superficie' e attendete istruzioni.
La rilevanza di RE diviene evidente qui. A questo stadio non avete ancora incontrato l'anguilla – quindi RE è impostata a 0 e voi potete scappare battendo "TORNA A RIVA" per cercare di trovare qualcosa di pesante.

Linea 30590: Questa linea vi permette di tornare al lago dopo che avete trovato il cristallo, se lo desiderate.

Linee 30610-30640: Questo è il pericolo principale – l'anguilla stessa. Come vi immergete nel lago, l'anguilla viene su e vi prende. L'uso delle Variabili necessita qui di un po' di spiegazione. Prima di partire abbiamo impostato ES a 1 – che vi impedisce di andar via dal lago di nuovo battendo "S" (Modulo 11.2, Linea 6010). Quindi impostiamo RE a 2 (che vi impedisce pure di scappare battendo NUOTA o TUFFATI invece di "N" o "S"). La ragione della necessità sia di RE che di ES è il fatto che potete scappare sia con una direzione che con la parola NUOTA. Q viene impostata a 0 perché stiamo contando il numero dei tentativi che il giocatore fa per sfuggire prima di essere ghermito dalla morte. Alla fine WW viene messa a 1 (questa variabile impedisce che il comando GUARDA dia la solita risposta – Modulo 12.2).

Linee 30650-30660: Queste linee si riferiscono ai tentativi di scappare battendo NUOTA ecc. che ho menzionato nel Modulo 12.2.

Linea 12500: Questa linea si riferisce ad un comando relativo allo sparare all'anguilla e si assicura che andiate alla linea opportuna. Nel caso di questo pericolo la mitraglietta non vi trarrà fuori dai guai – dato che sott'acqua non funziona.

Linea 12510: Se avete battuto UCCIDI o PUGNALA questa linea controlla che il coltello sia nell'inventario. Se lo è allora avete successo – il programma va alla Linea 12610.

Successo o sconfitta

MODULO 12.5

```
1 goto 3
2 save"@0:m12.5",8:verify"m12.5",8:stop
3 rem
12470 rem**
12480 rem anguilla
12490 rem**
12500 ifve$="spara"orve$="mitraglia"then
12520
12510 ifob%(8)=-1thenprint"OK. Avete us
ato il coltello.":gosub18000:goto12610
12520 ifob%(7)=-1thengosub18000:print"I
l mitra non funziona sott'acqua...";
12530 ifob%(7)=-1thenprint" e dovete esc
ogitare qualcosa di meglio..."
12540 ifob%(7)=-1thenprint"State diventa
ndo piu' debole...":goto5000
12550 print"L'anguilla vi tiene strett
o e vi stritola ";
12560 print"sempre piu'... state perdend
o i sensi..."
12570 form=1to2000:nextm
12580 gosub19000:print"Maledetti cuoch
i! Siete morto!"
12590 print"Avreste dovuto estrarre qu
alcosa per ucciderlo!!"
12600 gosub19000:goto190
12610 print"L'anguilla si dibatte ampia
mente nell'agonia - poi sprofonda ";
12620 print"nel lago.":gosub18000
12630 print"Non dimenticate il cristallo
...":ev=2:es=2:rc=0:yy=0:goto5000
```

Linee 12520-12540: Queste linee sono la risposta se non avete il coltello ma avete la mitraglietta. Avendo controllato nell'inventario che avete la mitraglietta vi arriva la risposta "LA TUA MITRAGLIETTA NON FUNZIONA SOTT'ACQUA". Venite mandati indietro alle Istruzioni per pensare e provare qualcosa d'altro.

Linee 12550-12600: Un'allegra sezione che interviene in due situazioni:
1. Non avete né coltello né mitraglietta
2. Se avete solo la mitraglietta e dopo 10 tentativi non siete sfuggiti.
Naturalmente non c'è modo di sfuggire questo piccolo rischio senza il coltello, e dato che il programma si ferma, non potete scappare a prendere il coltello – è quindi una specie di abbraccio mortale. Ma dare al giocatore la sensazione di avere una possibilità di salvezza, aiuta ad aggiungere mistero all'Adventure. (Infido, eh?).

Linee 12610-12630: Finalmente il successo. L'anguilla muore e con un accenno il programma aiuta a non dimenticare il cristallo, vi consente di lascire la Locazione e imposta le variabili di nuovo.
Potreste, se volete, omettere la Linea "NON DIMENTICARE IL CRISTALLO" e lasciare che il giocatore torni indietro alla nave solo per scoprire che ha dimenticato di battere "PRENDI CRISTALLO" al lago.

MODULO 12.6

```
1 goto 3
2 save"@0:m12.6",8:verify"m12.6",8:stop
3 rem
5040 rem**
5050 rem loop anguilla
5060 rem**
5070 ifp=92then5290
5080 ifp=67then5200
5090 ifp<>28then6000
5100 ifev=2then6000
5110 ifp=28andes=2then6500
5120 q=q+1
5130 ifq<5goto6000
5140 ifq=5thenprint"[RV]state[2] veramente i
ndebolendovi...":goto5120
5150 ifq=10thengosub19000:print"[RV]Ecco !
";n$
5160 ifq=10thenprint"Siete stati stritol
ati a morte.":goto190
```

Linee 5070-5110: Queste linee assicurano che il programma bypassi questo piccolo loop se non siete nella Locazione 58 e non siete tra le spire dell'anguilla. Dato che ci sono altri loop da considerare, il programma passa all'area opportuna. È meglio che voi semplicemente battiate queste linee e torniate indietro poi, per vedere esattamente dove vanno.

Linee 5120-5160: Questo è il vero loop. Ogni volta che il giocatore dà un comando al 64 (come UCCIDI e altro) Q si incrementa da 1. Quando Q = 5 c'è un piccolo richiamo che il tempo sta per scadere... "STAI DIVENTANDO SEMPRE PIÙ DEBOLE". Se non ne venite fuori al decimo tentativo, tornate al primo quadro.

Prova dei Moduli 12.1-12.6

Sfortunatamente non potete provare questi Moduli dato che alcune linee contengono dei GOTO che mandano il programma a una sezione che aggiungeremo nel prossimo Capitolo.

Sommario

Questo Capitolo è un tour de force di programmazione nei giochi di Adventure, contenente un confuso miscuglio di idee, variabili e situazioni. Non scoraggiatevi se non siete sicuri di aver capito tutto – perché il peggio l'ho passato prima io! Ho trovato che il risolvere la programmazione fosse estremamente stimolante (una volta avrei smesso di lavorarci su) e al tempo in cui finii l'intera Adventure, ero veramente dispiaciuto che l'ampiezza del testo che avevo scelto per aggiungere atmosfera fosse limitata, diminuendo le mie possibilità di scelta dei pericoli.
Sto tuttora pensando a come cambiare ciò nella mia prossima Adventure.

CAPITOLO 13

DISCORRENDO DI AMENITÀ

Una lontana storia...

Il nostro intrepido eroe ha cercato di pilotare la sua astronave per atterrare in verticale in un vasto deserto di un remoto pianeta, lontano dalle normali rotte spaziali. Egli deve evitare di distruggere la sua astronave, ha perso il suo prezioso cristallo di energia, e, come se non bastasse, l'amore della sua vita è stato rapito da una persona sconosciuta. La salvezza sembra lontana, e adesso egli deve adoperarsi per trovare la Principessa, e il suo cristallo di energia, se vuole ancora vedere gli azzurri cieli e le verdi colline di casa sua.
Egli si mette in cammino attraverso il territorio alieno, incontrando rischi e misteri nella sua ricerca, finché alla fine dopo un po' di tempo trova il suo cristallo in fondo ad un lago, e riesce a superare l'attacco di un'anguilla gigante che tenta di impedirgli il recupero. Finalmente riporta il cristallo sulla sua astronave.
Rileggiamola...
Quando avevo raggiunto questo stadio nella mia programmazione sono incappato in un piccolo problema – quando avevo programmato le locazioni avevo inizialmente dimenticato di ritornare o con il cristallo o con Aurora (o con entrambi) alla locazione 58 e 59 (la navicella). Quindi la prima volta che sono tornato alla nave, (supponendo con il cristallo, la Principessa, o entrambi) ho previsto solo che potesse apparire sullo schermo che entrambi erano ancora mancanti.
In altre parole avevo bisogno di alterare la scritta sullo schermo se tornavo indietro con OB%(2) – il cristallo – o con OB%(9) – Aurora – nell'inventario. Adesso è ora di rimediare a questo inconveniente.

Un finale felice

Il modulo che segue contiene le linee mancanti dalla sezione delle Locazioni (Modulo 9.1), e le linee per "trovare" appropriatamente Aurora e il cristallo. Mi sono basato sulle seguenti possibili combinazioni:

1. Ritornate con la Principessa, ma non con il cristallo
2. Ritornate col cristallo, ma non con la Principessa
3. Ritornate con entrambi
4. Salvate la Principessa, ma dimenticate di prenderla.

Avevo bisogno di quest'ultima opzione perché era possibile superare l'ostacolo degli alieni (di cui nel Modulo 13.4), ma semplicemente dimenticare di battere PRENDI AURORA. Ho pensato che non fosse necessario prevedere anche la non presa del cristallo, perché questo era più ovvio (e non ho messo segnalazioni nel testo in proposito).

```
1 goto 3
2 save"@0:m13.1",8:verify"m13.1",8:stop
3 rem
3330 print"■■Dopo sole poche ore di lavo
ro potete decollare con l'astronave ";
12070 ifp=59andos$="aurora"then33350
21205 ifob%(2)=-1andob%(9)=-1then33280
21210 ifob%(2)<>-1then21240
21220 ifob%(2)=-1thenprint"■Magnifico■ A
vete il cristallo.":cv=2
21230 print"■Mettetelo a riposare nella
teca nella cabina principale.":goto21380
21240 ifpr=2andob%(9)=-1then21330
21250 ifhh=2then21270
21260 ifpr=2andob%(9)<>-1then21350
21280 ifpr=2then21380
21330 print"■Indietro con aurora - Potre
i portarla nella cabina principale ";
21340 print"e offrirle una tazza di te'!
!":goto21380
21350 gosub19000:print"■Dimenticate di ■
prendere aurora nel villaggio ";
21360 print"indigeno ";n$
21370 print"Un po' stupido - tutto cio'!
! Dovete tornare per lei."
21400 ifcv=2andpr=2then33280
21410 ifcv=2andpr<>2then33400
21420 ifcv=0andpr=2then33460
21450 ifpr=2then21500
22010 goto22040
33250 rem**
33260 rem aurora trovata
33270 rem**
33280 gosub35100:print"■"
33290 print"■■■Congratulazioni!! Avete
compiuto la missione!!"
33300 print"■■Dopo solo poche ore potete
 decollare con l'astronave ";
33310 print"e tuffarvi nello spazio prof
ondo."
33320 print"■■E la principessa aurora ha
```

```
 alla fine deciso di non sposare ";
33330 print"il crudele tiranno di Zen in
fine!!":print" Preferisce voi ";n$
33340 print"■■Ce l'avete fatta eh ?":pok
ev+21,0:end
33350 gosub18000:print"■Aurora vi bacia
Per la fortuna che avete avuto nel ";
33360 print"trovare il magico cristallo.
":hh=2:goto12090
33370 rem**
33380 rem cristallo trovato
33390 rem**
33400 gosub35100:print"■":print"■Okay -
cosi' avete il cristallo energetico.";
33410 print"Ora tutto cio' che dovete fa
re e' trovare la principessa!"
33420 print"Non dimenticte di lasciare q
ui il cristallo.":goto21500
33430 rem**
33440 rem niente cristallo
33450 rem**
33460 gosub35100:print"■":print"■Okay. a
urora e' salva e voi potete ";
33470 print"lasciarla qui se volete.":go
to21500
```

Linea 12070: Questa linea viene dalla subroutine "BUTTA" caduta e viene applicata se battete "LASCIA AURORA" nella locazione 59. Essa vi manda ad una piccola sezione alla fine del programma (vedi più avanti in questo modulo).

Linea 21205: Stavo dimenticando questa linea (come potete vedere dal numero di linea che è stato aggiunto) che controlla se avete nell'inventario sia Aurora che il cristallo. Se li avete entrambi, allora c'è un "GOTO" alla linea 33280 che è il messaggio finale di successo del gioco.

Linea 21210: Nel momento in cui entrate nella locazione 58 dall'esterno, questa linea inizia a controllare lo stato del vostro inventario. Se non avete il cristallo, questa linea vi invia alla Linea 21240, dove controllerà se avete o no Aurora.

Linea 21220-21230: Queste linee presumono che voi abbiate il cristallo (altrimenti le avreste bypassate) e impostano CV a 2, stampano un messaggio carino, e vi inviano alla Linea 21380 (l'ultima linea della sezione delle Locazioni nel modulo 9.1), che vi consentirà di muovervi nella cabina. Notate che abbiamo trascurato tutto il precedente testo sul brandello del vestito di Aurora ecc., che non è mai più applicabile.

Linea 21240: Avendo controllato il cristallo nella Linea 21210, controlliamo adesso se

la Principessa è stata salvata. Questa linea controlla anche se avete lei nel vostro inventario, e, se si, vi invia alla Linea 21330 che dice "RITORNO CON AURORA...".
Linea 21250: Questa linea è l'inizio di una sequenza abbastanza complicata. Supponiamo che voi abbiate Aurora con voi e che la portiate sulla navicella. Adesso dovete passare di nuovo attraverso la camera di decompressione per tornare indietro e cercare il cristallo: tutto il testo nella camera di decompressione diventa un po' superfluo (brandelli del vestito di Aurora ecc.). Quindi ho spezzato questo testo in due parti.

1. Le semplici parole "SEI NELL'AIRLOCK" (Linea 21270) che apparirà ogni volta che voi passate dalla locazione 58.

2. Il resto del testo della camera (Linee 21990-21320). Quindi questa linea annota se HH = 2 (che è vero solo quando voi avete portato Aurora nella navicella e l'avete lasciata lì) e, se si, vi spedisce alla linea "SEI NELLA CAMERA DI DECOMPRESSIONE".
Linea 21260: Questa è la linea per i giocatori che hanno salvato Aurora ma che hanno dimenticato di prenderla.

Linea 21280: Questa linea viene dopo la linea "SEI NELL'AIRLOCK" ed è qui per controllare se avete già salvato la Principessa dagli alieni (PR = 2). Se no verrà scritto il resto del testo. Se sì, bypassate questa linea e andate all'ultima Linea di questa Locazione.

Linee 21330-21370: Le appropriate risposte.
Il problema che ho avuto in questa sezione è stato abbastanza considerevole, quello cioè di cercare di assicurarmi che avessi coperto tutte le possibilità. In un certo senso quest'area di programma è un perfetto esempio di come una piccola parte richiede una lunga riflessione – e anche di come la più lunga parte del lavoro di creazione di un gioco di Adventure sia la raffinazione e la correzione degli errori alla fine. Non sottostimate questi aspetti.
Adesso iniziamo la seconda sezione – le stesse risposte per la Locazione 59. Linea 21400: Se avete sia il cristallo che la Principessa, allora c'è un GOTO al messaggio di successo.
Linea 21410: Non c'è la Principessa, ma c'è il cristallo.
Linea 21420: Non c'è il cristallo, ma c'è Aurora.
Linea 21450: Questa è un'altra linea inserita nel testo solo per tagliare le parti relative a segni dello struggimento della Principessa nel caso voi l'abbiate salvata.

Linee 22000-22010: Queste linee sono inserite nel villaggio alieno. Forse non ricordate, ma quando avete battuto questa sequenza, ad un certo punto avete sentito una donna che cantava tristemente nelle vicinanze. Naturalmente si suppone che sia Aurora, quindi se voi la salvate dobbiamo omettere queste linee quando tornate indietro dalla capanna.

Linee 33280-33340: Ciò che ho liberamente battezzato il "messaggio di successo" Una volta che avete sia Aurora che il cristallo, entrate in questa subroutine. Appare Victor e viene fuori il messaggio di congratulazioni. Alla fine viene cancellato Victor e il programma finisce.

Linee 33350-33360: Se avete Aurora ma non il cristallo, questo messaggio appare quando voi lasciate la Principessa nella locazione 59 (ricordate la linea 12070). Notate che HH viene impostato a 2. Infine siete ritornati alla sezione "CADUTA" così che Aurora può essere lasciata cadere e verrà fatta l'appropriata sottrazione dell'inventario.

Linee 33400-33470: Queste due routine sono relative al testo che appare quando arrivate alla cabina portando o Aurora o il cristallo, ma non entrambi.
Rileggendo la spiegazione di questo modulo, io stesso che ne sono l'autore l'ho trovato un po' complicato, quindi andiamo su qualcosa di più semplice: vediamo gli altri pericoli nell'Adventure.

Paludi e mostri

MODULO 13.2

```
1 goto3
2 save"@0:m13.2",8:verify"m13.2",8:stop
3 rem
6640 ifleft$(i$,5)="salta"andp=67then133
10
6650 ifleft$(i$,4)="tira"andp=67then1331
0
6660 ifleft$(i$,9)="galleggia"andp=67the
n13310
6670 ifleft$(i$,5)="nuota"andp=67then133
10
7710 ifve$="lancia"andp=67thengosub18000
:goto13300
13270 rem**
13280 rem sabbie mobili
13290 rem**
13300 ifob%(1)=-1andno$="corda"thenprint
"■scaltra mossa ";n$:goto13305
13302 goto7865
13305 print"La fune pende da un albero e
 voi la prendete. ":qv=2:goto4320
13310 gosub19000:print"■Non vi sara' d'a
iuto quella via ";n$:goto5000
13320 print"■Affondate sempre piu'!"
13330 form=1to1000:nextm
13340 gosub19000:print"Glu... glu... sie
te perduti":goto190
```

Questo modulo riguarda alcune risposte I$ e le sabbie mobili che incontrerete nella locazione 67 (giusto a sud della navicella). Per uscire dalle sabbie mobili dovete avere la corda (trovata nella locazione 15) e dovete valutare il rischio di prendere il coltello

che vi serve per uccidere l'anguilla (essa sta cominciando a mugolare il motivo "La casa che Jack ha fatto"!!).

Linee 6640-6670: Una gamma di possibili parole che potete usare.

Linea 7710: Il comando VE$ che vi porta al successo.

Linee 13300-13305: Se avete una corda, allora potete lanciarvi fuori dalle sabbie mobili, impostare QV = 2, e proseguire la vostra strada. La linea 13302 serve solo a evitare l'istruzione di scrittura nella linea 13305 nel caso in cui voi battiate "LANCIA UNA BOMBA" o qualcosa d'altro.

Linea 13310: Se vi capita di non avere la corda con voi – Avete solo una piccola speranza, un loop di tempo, prima di sprofondare nella palude. Mentre state cercando di uscire, questo messaggio verrà fuori ogni volta che ci provate, e "LANCIATE" qualcosa.

Linee 13320-13340: Alla fine, dopo che avete fatto quattro tentativi falliti per uscire, questa piccola routine esce e stampa "Glug... glug... Sei affondato!!". Per capire completamente questa sezione dovete introdurre il Modulo 13.5.
Il prossimo modulo riguarda il dinosauro che vi aspetta sulla vasta strada a sud verso il villaggio degli alieni. Ho voluto rendere il dinosauro un po' difficile da passare – ma non un pericolo mortale che possa stroncarvi. Forse uno dei piccoli extra della creazione del vostro gioco è il controllo che voi avete sul vostro giocatore potenziale. Egli o ella non hanno idea di quanto letali possano essere i pericoli, ne di quanti falsi richiami e trappole avete messo in giro.

Torniamo al dinosauro. Quando ho scritto la storia l'idea era di ucciderlo con alcuni colpi ben piazzati del mitra, ma più ci pensavo e meno mi piaceva. Pensavo che il serpente morisse troppo facilmente. Dopo averci pensato un po' su, e aver sfogliato la mia grande collezione di fantascienza, decisi che il modo logico di sfuggire fosse di arrampicarsi su un albero e dondolarsi al di là. E questo è il modo in cui il programma finì per essere.

Appena mi venne in mente l'idea di arrampicarsi su un albero, decisi di costruire una sezione dedicata all'arrampicamento sugli alberi in qualunque posto della foresta – e questo è quanto il Modulo 13.3 è in grado di fare.

MODULO 13.3

```
1 goto3
2 save"@0:m13.3",8:verify"m13.3",8:stop
3 rem
6860 ifi$="scala"thengosub18000:print"▮▮C
osa volete che scali ?":goto5000
7500 ifve$="scala"then7920
7700 ifleft$(ve$,3)="cor"orleft$(ve$,3)=
"cam"thenprint"▮▮uale via ?":goto32500
```

```
7920 ifno$<>"albero"then7970
7930 ifp=42orp=43orp=44then9oto32500
7940 ifp=52orp=53orp=54orp=55then9oto325
20
7950 ifp=62orp=63orp=64orp=65then9oto325
40
7960 ifp=74then13580
7970 9osub19000:print"Non potete propri
o scalare ";no$:9oto5000
7980 9osub19000:print"Spiacente! Ma non
 so come ":printve$:9oto5000
13470 rem**
13480 rem dinosauro
13490 rem**
13500 ifob%(7)<>-1then9osub18000:9oto135
40
13510 9osub19000:print"Sebbene sparate
continuamente al meglio - i vostri ";
13520 print"colpi rimbalzano sulla sua p
elle corazzata !!!"
13530 print"Cosa ora ";n$;" ?":9oto5000
13540 print"Non sembrate in realta' pos
sedere il necessario ";
13550 print"Per uccidere una cosi' 9ross
a bestia";
13560 print"Penso che sarete capaci di e
vitarlo e di rimanere vivi!! ";
13570 print"Ma il problema e' come super
arlo !!":9oto5000
13580 print"":9osub3510
13590 print"Mossa furba,9uesta ";n$;:pr
int"Ora dovete passare tra 9li alberi ";
13600 print"e superarlo. Discenderete an
cora piu' avanti.":dv=2:9oto4320
32470 rem**
32480 rem scala albero
32490 rem**
32500 9osub18000:print"A nord c'e' una
citta' in rovina - a sud molti ";
32510 print"alberi aarivano all'orizzont
e. Discendete.":9oto4320
32520 9osub18000:print"A sud potete ved
```

```
ere la vostra navicella - ovunque ";
32530 print"alberi su alberi ... Discend
ete.":goto4320":goto4320"
32540 gosub18000:print"A sud potete ved
ere del fumo salire al cielo. Puo' ";
32550 print"esserci vita o cose simili.
Discendete.":goto4320
```

Linee 6860 e 7500: I comandi IS$ e VE$ per SCALA. Qui potrebbe essere una buona occasione per sottolineare la differenza tra i due. Il comando IS$ è per il giocatore che semplicemente batte SCALA e nient'altro, e la risposta è "DOVE MI SUGGERISCI DI ARRAMPICARMI/". Il comando CLIMB di VE$ sarà sempre la prima parte di un comando, che potrebbe essere SCALA MURO o SCALA ALBERO.

Linea 7700: Questa linea è in relazione coi possibili comandi CAMMINA o CORRI.

Linee 7920 e 7970: L'unico valido sostantivo che voi potete arrampicare è ALBERO, quindi se il giocatore batte qualunque altra cosa queste istruzioni rispondono con "NON PUOI ARRAMPICARTI SU UN" e quindi la parola appropriata (es. muro o sassi).

Linee 7930-7960: Se vi arrampicate su un albero di una foresta del nord ricevete un messaggio, se vi arrampicate su un albero nel centro ne prendete un'altro, e ancora un terzo se siete al sud. La linea finale 7960 è per arrampicarsi su un albero durante lo scontro col dinosauro.

Linea 7980: Una risposta finale ad un comando VE$ che coinvolga una parola non valida (come STRANGOLA o LEGGI).

Linee 13500 e 13540-13570: Se provate e uccidete il dinosauro senza avere il mitra prendete questo messaggio.

Linee 13510-13530: Se avete il mitra allora queste linee vi dicono 18380 che i colpi rimbalzano sulla sua pelle. E ora?

Linee 13580-13600: Alla fine il successo. Vi siete arrampicati sull'albero e potete passare al di là.
Solo una nota. In questa locazione voi potete andare di nuovo a nord dove incontrereste il dinosauro, ma non potete andare a sud. Le linee che permettono questa situazione sono le Linee 6170 e 6180 nel Modulo 11.2

Linee 32500-32550: Queste linee sono la descrizione di quello che potete vedere dalla cima dell'albero. Ho cercato di dare solo qualche indizio senza lasciar trapelare troppa informazione, un buon modo di mantenere l'interesse.
Il prossimo modulo si occupa di altri due pericoli – il serpente nella città e gli alieni che tengono Aurora. Il serpente è un pericolo abbastanza lieve, facile da evitare e che non causerà nessun problema ad un vero Adventurer (eccetto il caso in cui il vero Adventurer

si stia aspettando un qualche trabocchetto – così è una specie di doppio bluff). Gli alieni saranno un po' più difficoltosi, e descriverò più estesamente la programmazione nella prossima sezione.

MODULO 13.4

```
1 goto3
2 save"@0:modulo13.4",8:verify"modulo13.
4",8:stop
3 rem
7570 ifve$="prendi"andp=92then33120
7580 ifve$="afferra"andp=92then33120
7600 ifve$="acchiappa"andp=92then33120
12970 rem**
12980 rem serpente
12990 rem**
13000 ifob%(7)=-1thenprint"■Okay":gosub1
8000:goto13030
13010 ifob%(8)=-1thenprint"Avete solo un
 coltello - troppo poco !":goto5000
13020 print"■Non avete nulla con cui ";v
e$;" .. e ora ?":goto5000
13030 print"■Il serpente cozza contro un
a casa vicina sollevando nuvole di ";
13040 print"polvere in aria, nel rantolo
 mortale."
13050 print"■Contenti di avere questo mo
tra"n$:sv=2:goto4320
13670 rem**
13680 rem indigeni
13690 rem**
13700 ifob%(7)=-1then13730
13710 gosub18000:print"■Sono troppo lont
ani per ucciderli! Meglio tentare ";
13720 print"un'altra via. Forse aurora p
otrebbe aiutare ...":goto5000
13730 gosub18000:print"■Il vostro mitra
ha bisogno di molte ore per ";
13740 print"ricaricarsi dopo la lotta co
l dinosauro.";
13750 print" Dovete pensare a qualcos'al
tro.":goto5000
```

```
33090 rem**
33100 rem liberazione illegale di aurora
33110 rem**
33120 ifnv<>2andno$="aurora"then33170
33130 ifnv<>2andno$="Principessa"then331
70
33140 ifnv<>2andno$="Principessa"then331
70
33150 goto11700
33170 gosub35100:print"[illegible]"
33180 print"[illegible]Spiacente ";n$:print"ma gli
 indigeni non amano i Piloti !!";
33190 print"  e vi hanno impedito di ten
tare di liberarla.
33200 print"Vi hanno ucciso ! A questo v
i ha portato la via!!":goto190
```

Linee 7570-7600: Se voi provate a prendere Aurora in questa locazione queste linee vi mandano all'appropriata area del programma.

Linea 13000: Se avete il mitra (e ogni Adventurer degno del suo nome si assicurerà di averlo preso subito all'inizio) allora ricevete questo messaggio se voi "uccidete" o 'sparate' al serpente.

Linee 13010-13020: Se vi capita di non avere il mitra cadrete alla Linea 13010 che controlla se avete il coltello. Se l'avete allora vi dà questo messaggio. Se non avete neppure quello andate alla Linea 13020.

Linee 13030-13050: Se riuscite ad uccidere il serpente vi appare il messaggio appropriato e la variabile SV viene impostata a 2.

Linee 13700-13750: Questa sezione si riferisce ad un tentativo di uccidere gli alieni. Per prima cosa va a vedere se avete il mitra (Linea 13700) e se l'avete, ricevete la cattiva notizia che ha bisogno di essere ricaricato (Linee 13730-13750). Se non avete il mitra le Linee 13710-13720 vi danno un piccolo indizio sul fatto che Aurora potrebbe aiutare.

Linee 33120-33200: Supponete che abbiate tentato di prendere Aurora in questa locazione (92), allora sareste spediti a questa parte del programma. Per prima cosa guarda se avete battuto "aurora", "principessa" o "Principessa" (notate la P maiuscola) e se no la Linea 33150 vi spedirerà indietro alla routine PRENDI.

Se lo avete fatto tuttavia le Linee 33170-33200 dicono, da parte di Victor, che gli alieni "... non amano gli intraprendenti..." e vi avrebbero ucciso.

```
1 goto3
2 save"@0:m13.5",8:verify"m13.5",8:stop
3 rem
5170 rem**
5180 rem loop della palude
5190 rem**
5200 ifp=92then5290
5210 ifp<>67then6000
5220 ifqv=2then6500
5230 q=q+1
5240 ifq<5goto6000
5250 ifq=5goto13320
5260 rem**
5270 rem loop indigeni
5280 rem**
5290 ifnv<>2then6000
5300 ifnv=2then6000
5310 ifp=92andpr=2then6500
5320 q=q+1
5330 ifq<5then6000
5340 ifq=5thengosub18000:print"▮▮Tenta ";
n$:print"Cosa il prossimo?":goto5320
5350 ifq=10andi$="bacia aurora"then14030
5370 ifq=10thengosub18000:print"▮▮troppo
tardi! Gli indigeni vi hanno scacciato";
5380 ifq=10thenprint" fuori dalla capann
a - non potete prendere aurora ";
5390 ifq=10thenprint"con voi. Meglio and
ar via e pensare ad altro ";
5400 ifq=10thenprint"per liberarla.":q=0
:na=1:np=1:goto4320
```

Questo Modulo aggiunge due loop di tempo per le sabbie mobili e gli alieni. Seguendo lo stesso modello del Loop dell'anguilla (Modulo 12.6) vengono contati gli input e dopo 5 (per la palude) e 10 (per gli alieni) indirizza il programma all'appropriato responso. Nel caso delle sabbie mobili – l'avete già avuto. Nel caso degli alieni voi potete lasciare la capanna per andar via e pensare ancora.

Sommario

Il programma adesso è pressoché completo. Io spero che questo e il prossimo capitolo comincino a darvi idee che potrete sviluppare voi stessi per fare la vostra Adventure

interessante e divertente per il giocatore. Come ho già detto uno degli aspetti che mi ha lasciato scontento è stato il fatto che per creare una certa atmosfera – tale che vi sentiste veramente sul pianeta – ho usato molta memoria per il testo della descrizione, e ciò mi ha lasciato con troppo poco spazio per abbellire i pericoli e aggiungere enigmi più complicati e problemi da risolvere. Penso comunque di aver preso la giusta decisione in quanto io non amo questo genere di stampe:

siete in una cella

chiave in bottiglia

e ora?

comuni tanti Adventure. "*Il Pianeta Dell'Incubo*" è arricchito con prosa da fantascienza (un po' adattata per le costrizioni del 64) e non mi sarebbe piaciuto in nessun altro modo.

CAPITOLO 14

UN BACIO TEMPESTIVO

Dobbiamo solo riordinare alcune cosette sparse, e la nostra storia è completa. Può essere sembrata un po' lunghetta, ma io mi sono divertito, e così spero di voi.
In questo Capitolo ci occuperemo di alcune parole extra che i giocatori potrebbero usare – e come un gran finale – del salvataggio della Principessa.

Con una spintarella degli amici

AIUTO è probabilmente la parola più usata e più noiosa di tutte le parole negli Adventure. Può essere la luce che vi illumina dopo quello che prima era un problema insormontabile, e può essere la frustrazione che minaccia l'esistenza del vostro micro. Ci sono state volte che volevo sfasciare la tastiera per certe sue stupide risposte che non mi aiutavano per niente, e ci sono state volte che l'avrei baciata (metaforicamente, naturalmente). Quindi quando sono arrivato al **Pianeta dell'Incubo**, ho dovuto decidere quanto aiuto avrei dato al mio giocatore potenziale. Ho deciso che Victor doveva scendere giù e introdurre la sezione di AIUTO per dare alcuni suggerimenti. Naturalmente voi potete dare degli indizi fuorvianti (molto astuto) ma io sono fondamentalmente onesto ed ho deciso di giocare correttamente.

MODULO 14.1

```
1 goto3
2 save"@0:m14.1",8:verify"m14.1",8:stop
3 rem
6560 ifi$="a"ori$="aiuto"then11000
10970 rem**
10980 rem aiuto
10990 rem**
11000 ifhr=2then11250
11005 ifp=24thengosub35100:print"◙":goto
11090
11010 ifp=58thengosub35100:print"◙":goto
11100
11020 ifp=67thengosub35100:print"◙":goto
11120
11030 ifp=92thengosub35100:print"◙":goto
11140
11040 ifp=1orp=2orp=11orp=12orp=22thengo
```

```
sub35100:Print"■":goto11190
11050 ifp=42orp=62orp=63thengosub35100:p
rint"■":goto11170
11060 gosub35100:Print"■"
11070 Print"■Ho un Piccolo Problema che
mi turba ";n$
11080 Print"Tentare di tener testa al vo
stro Per un Po'...":goto5000
11090 Print"■Tenta di ucciderlo ";n$:Pri
nt"Prima che lo faccia lui!!":goto5000
11100 Print"■Ora Pensa seriamente ";n$:P
rint"Ci Puo' essere qualcosa intorno ";
11110 Print"che Puo' aiutarvi...":goto50
00
11120 Print"■Situazione difficile,questa
 ";n$:Print"Suggerisco di buttare";
11130 Print" qualcosa Per lanciarvi fuor
i.":goto5000
11140 Print"■■■Poiche' gli indigeni Pens
ano che aurora sia la loro regina ";
11150 Print"e' meglio se vi fate vedere
suo amico.":goto5000
11170 Print"■Ci siamo Perduti nella fore
sta, eh? Perche' non scalate un ";
11180 Print"albero e guardate intorno.":
goto5000
11190 Print"■Sembriamo Perduti entrambi
";n$:Print"Meglio ";
11200 Print"incamminarci, alla fine trov
erete una via.":goto4320
11250 Print"■":gosub35100:Print"■Non sia
mo amici, al momento ";n$:goto11310
11280 Print"■":gosub35100:Print"■Non Pot
ete uccidermi ";n$
11290 Print"■Sono il nuovo modello invul
nerabile. Ma...":goto11310
11300 Print"■":gosub35100:Print"■Cio' no
n e' carino, ";n$
11310 Print"Non Posso ■aiutarvi■ Piu'";
11320 Print" Potete farvi fronte da solo
.":goto5000
```

Linea 6560: il comando I$.

Linea 11000: Nel modulo 10.4 e 12.1 potete ricordare che le linee 7865 e 12305 prevedono la possibilità di scagliare qualcosa contro Victor, o di ucciderlo. Se ciò accade sembrerebbe logico che Victor si risenta un po', e non sia disposto ad aiutare così facilmente; quindi per tener conto di questo fatto ho impostato la variabile HR a 2. Questa linea istruisce il programma ad evitare le routines di AIUTO se HR non è uguale a 2.
Linee 11010-11050: Questi sono tutti i puntatori che intervengono nel caso voi battiate AIUTO in una qualunque delle opportune Locazioni.
Linea 11060-11080: La risposta standard se voi chiedete l'AIUTO in qualunque altra Locazione.
Linee 11090-11200: Queste linee sono le risposte specifiche, di nuovo dipendenti dalla locazione.
Linee 11250-11320: Se vi siete comportati male con Victor allora questa sezione si occupa della reazione ad ucciderlo o a tirargli qualcosa. Il tocco di Victor piuttosto di malumore perché lo avete spazientito è stato suggerito da Brendon Gore – e illustra l'importanza di chiamare qualcuno a provare il vostro gioco prima che voi esauriate tutta la memoria a disposizione. Io sapevo che i suggerimenti di Brendon erano geniali, e che sarebbe stato molto lieto che li avessi incorporati.
Avevo lasciato degli spazi in questa sezione per permettermi di aggiungere altre reazioni in data successiva, ma sfortunatamente la carenza di memoria me lo ha impedito.

Utilizzare e sparare

MODULO 14.2

```
1 goto3
2 save"@0:modulo14.2",8:verify"modulo14.
2",8:stop
3 rem
7660 ifve$="usa"then11500
11470 rem**
11480 rem usa.
11490 rem**
11500 ifp<>24orp<>74andleft$(no$,5)="mit
ra"then11580
11510 ifp<>24orp<>74goto11610
11520 ifp=24orp=74andleft$(no$,5)="mitra
"then11580
11530 ifp=24orp=74andno$="coltello"then1
1550
11540 goto11660
11550 ifob%(8)<>-1thengosub19000:print"▒
```

```
Non l'avete!!":goto5000
11560 ifob%(8)=-1thengosub35100:print"■"
:print"■Proprio un coltello ";n$;"?"
11570 ifob%(8)=-1thenprint"■E' un po' am
bizioso, non vi pare?":goto5000
11580 ifob%(8)<>-1thengosub19000:print"■
Non l'avete!!":goto5000
11590 ifp=24andob%(7)=-1thengosub18000:p
rint"■Bella pensata!":goto13030
11600 ifp=74andob%(7)=-1thengosub35100:p
rint"■":print"■Bella pensata!":goto13510
11610 fori=1to9
11620 ifob$(i)=no$then11630
11630 ifob%(i)=-1$then11660
11640 nexti
11650 gosub19000:print"■Non ha senso usa
re cio'!":goto5000
11660 print"■Come pensate di usare tale
";no$:gosub18000:goto5000
```

Questo modulo è un modulo standard che segue la stessa strutturazione di molti altri, e si occupa del comando USA. Ho concentrato l'attenzione sull'uso della mitraglietta e del coltello, dato che questi sembrerebbero essere gli oggetti più logici.
Notate che le ultime due linee del modulo permettono di riconoscere un oggetto come valido (linea 11650) e quindi di essere riconosciuti (linee 11660).

MODULO 14.3

```
1 goto3
2 save"@0:m14.3",8:verify"m14.3",8:stop
3 rem
7670 ifve$="fuoco"then12700
7680 ifve$="spara"orve$="mitraglia"then1
2300
7690 ifve$="pugnala"orve$="taglia"then12
760
12670 rem**
12680 rem far fuoco/pugnalare
12690 rem**
12700 ifp=24andob%(7)=-1andleft$(no$,5)=
"mitra"then13030
12710 ifp=28andob%(7)=-1andleft$(no$,5)=
"mitra"then12520
```

```
12720 ifp=74andob%(7)=-1andleft$(no$,5)=
"mitra"then13510
12730 ifp=92andob%(7)=-1andleft$(no$,5)=
"mitra"then13730
12740 ifleft$(no$,5)="mitra"andob%(7)<>-
1thenprint"Non l'avete!":goto5000
12750 gosub18000:print"Non vi sara' d'a
iuto.":goto5000
12760 ifp=28andob%(8)=-1then12500
12770 ifno$="coltello"andob%(8)<>-1thenp
rint"Non l'avete!":goto5000
12780 goto12750
```

Un'altro modulo standard che è un duplicato del Modulo UCCIDI, ma che vi permette di eseguire i comandi SPARA, MITRAGLIA, PUGNALA, TAGLIA e quindi indirizza il programma.

La locazione dei frutti e delle noci

I frutti e le noci sul cespuglio situato subdolamente nel deserto hanno un significato speciale. Una volta mangiati potete togliervi la tuta spaziale senza respirare l'atmosfera velenosa (qui dovete crederci!), e se non vi togliete la tuta spaziale non potete baciare Aurora e fuggire.
La programmazione di questo modulo sul 64 ha richiesto un parto del cervello ed un lavoraccio fino a tarda notte! Dovevo disseminare linee nel modulo PRENDI in modo abbastanza considerevole dato che avevo bisogno di poter far fronte a situazioni in cui qualcuno avrebbe provato a prendere le noci e i frutti nella Locazione 37, oppure mangiare i frutti o le noci in questa o in un'altra Locazione. Allo stesso tempo dovevo anche prepararmi al ritorno del giocatore alla Locazione 37 una seconda o terza volta, e al tentativo di prendere o mangiare le noci e i frutti una seconda, terza o quarta volta. Il risultato è il Modulo 14.4.

MODULO 14.4

```
1 goto3
2 save"@0:modulo14.4",8:verify"modulo14.
4",8:stop
3 rem
6680 ifi$="mangia"thenprint"Cosa vi piac
erebbe mangiare?":goto5000
7650 ifve$="mangia"then31600
11720 ifp=37andleft$(no$,5)="frutt"then1
1860
11750 ifp=37andno$<>"noci"then11800
11760 ifp=37andleft$(no$,5)<>"frutt"then
```

```
11800
11770 ifP=37andno$="noci"then11790
11780 goto11800
11790 ifP=37andz=1then11880
11860 ifz=1then11880
11870 goto11830
11880 gosub18000:Print"▒Prova a mangiare
 ";no$:goto5000
12060 ifob%(6)=-1andno$="tuta-spaziale"t
hen31500
31470 rem**
31480 rem se rimuovete la tuta
31490 rem**
31500 ifP=58then12080
31510 ifP=59then12080
31520 ifP<>2then31000
31530 ifP=2then12080
31570 rem**
31580 rem se mangiate le noci
31590 rem**
31600 ifno$="noci"ornleft$(no$,5)="frutt
"then31620
31610 gosub19000:Print"▒Uuh! Perche' vol
ete mangiare ▒Questo▒?":goto5000
31620 ifP=37thenPrint"▒Okay.Mm! buon sap
ore!":goto31650
31630 ifob%(3)=-1thenPrint"▒Okay.Mm! buo
n sapore!":goto31650
31640 ifob%(3)<>-1thengosub19000:Print"▒
Non avete nessun ";no$:goto5000
31650 ifz=1then31700
31660 gosub35100:Print"▒":Print"▒Ispirat
o, muovete questo ";n$
31670 Print"▒▒Potreste provare una stran
a sensazione percorrervi il corpo ";
31680 Print" e Potete respirare (felicem
ente) senza la vostra ";
31690 Print"tuta. Potete toglierla.":ob%
(3)=0:z=1:fv=2:g=g-1:goto5000
31700 gosub18000:Print"▒▒Questa volta no
n sembra diverso. ";
31710 Print"Blocca ancora il vostro sens
o di fame.":ob%(3)=0:goto5000
```

Linee 6880 e 7650: I comandi I$ e VE$.

Linee 11720-11790: Queste linee sono mescolate alle linee standard del modulo PRENDI (modulo 10.3). Fondamentalmente esse vi permettono di battere PRENDI FRUTTA o PRENDI NOCI sia nella Locazione 37 che in qualunque altra Locazione. La variable Z è riportata a zero finché non avete mangiato le noci la prima volta – dopo di che viene rimessa a 1. Ciò fa cambiare la ripetizione (come vedrete tra un momento).

Linee 11860-11880: Sempre nel Modulo "PRENDI" la Linea 11860 inizia con un controllo della variabile Z. Se Z = 1 allora la risposta alla linea 11080 è già tentato di mangiarli". La ragione di ciò è dovuta alla situazione in cui il giocatore prova a prendere le noci una seconda volta, dopo averle già mangiate.

Se Z = O allora al comando "PRENDI FRUTTA" il programma torna alla Linea 11830 che è la parte "OKAY" del modulo "PRENDI".

Linea 12060: Questa Linea viene dal modulo BUTTA e si riferisce alla situazione in cui decidete di lasciar cadere o togliere la tuta spaziale prima che abbiate mangiato le noci. Vi manda alla linea 31500.

Queste prossime due sezioni del Modulo 14.4 servono a far fronte al fatto che voi provate a togliervi la tuta spaziale troppo presto – morireste nell'atmosfera venefica... a meno che non abbiate mangiato le noci.

Linee 31500-31530: Ovviamente vi potete togliere la tuta sulla navicella spaziale – quindi le linee 31500 e 31510 si occupano di ciò. La variabile FV controlla il fatto di mangiare i frutti o le noci (un valore zero significa che non avete ancora mangiato, mentre 2 vuol dire che avete mangiato). Quindi la linea 31520 vi invia all'area "VOI MORIRETE" e la linea 31530 vi invierà alla risposta "OK" nel modulo BUTTA.

Linee 31600-31610: In risposta al comando MANGIA la linea 31600 controlla se NO$ è NOCI o FRUTTA. Se no andate alla risposta standard "HUH! PERCHÉ VUOI MANGIARE CIÒ?" nel caso in cui il giocatore batte "MANGIA ALIENI" o qualche altro sciocco comando.

Linee 31620-31710: Questa situazione fa fronte alle seguenti situazioni:

1. Voi mangiate le noci senza averle prima prese (nella Locazione 37) – Linea 31620.
2. Voi mangiate le noci dopo averle prese (che può succedere in qualsiasi Locazione) – Linea 31630.
3. Voi provate a mangiare le noci fuori dalla Locazione 37 mentre esse non sono nel vostro inventario – Linea 31640.
4. Voi mangiate le noci una seconda, terza, ecc., volta – Linea 31650.

Notate che la linea 31690 contiene alcune variabili che fanno quanto segue:

1. Tolgono le noci dall'inventario
2. Impostano Z a 1
3. Impostano FV a 2 per permettere di togliere in sicurezza la tuta spaziale
4. Sottraggono 1 a G (la variabile inventario).

Uomo straordinario!

Questo deve essere il momento più importante del gioco – voi avete trovato Aurora e l'avete salvata. Ho voluto qualcosa di speciale per rimarcare il momento del trionfo, e ho deciso di ampliare la programmazione per il momento del bacio includendo suoni e immagini che siano veramente speciali. Questo modulo finale riguarda questo momento.

MODULO 14.5

```
1 goto3
2 save"@0:m14.5",8:verify"m14.5",8:stop
3 rem
7730 ifve$="bacia"andp=92andno$="aurora"
then14000
7740 ifve$="bacia"andob%(9)=-1andno$="au
rora"then14210
7750 ifve$="bacia"thengoto7810
7760 ifp<>92then7780
7770 ifve$="abbraccia"orve$="ama"orve$="
stringi"thengoto7800
7780 ifve$="abbraccia"orve$="ama"orve$="
stringi"thengoto7810
7790 goto7890
7800 gosub18000:print"■Non e' abbastanza
 socievole. Tenta dell'altro.":goto5000
7810 gosub19000:print"■Agite in modo un
po' strano ";n$:goto5000
13970 rem**
13980 rem bacia la principessa
13990 rem**
14000 ifob%(6)=-1thengosub19000:print"■N
on potete baciarla con la tuta";
14010 ifob%(6)=-1thenprint", vero? Avevo
 pensato che lo sapeste!"
14020 ifob%(6)=-1thenprint" Sta indicand
o le sue labbra...":goto5000
14030 forl=0to24:pokesc+l,0:nextl
14040 pokel3,3:pokew3,16:pokesc+3,1:poke
vo,143:pokes1,240:pokesc+4,65
14050 fr=5389:fort=1to200:f9=fr+peek(sc+
27)*225
14060 hf=int(f9/256):lf=f9-hf*256
14070 pokel1,lf:pokeh1,hf
14080 nextt
```

```
14090 printchr$(13)
14100 fort=1to5
14110 print"▒▒UOMO▒ ▒TERRIBILE▒ ▒!!!▒"
14120 form=1to500:nextm
14130 print"▒                    "
14140 form=1to500:nextm
14150 nextt
14160 pokevo,0
14170 print"▒▒▒Cio' non vi ha dato solo
un potere positivo, ma gli indigeni ";
14180 print"vi stanno sorridendo. Potete
 fuggire con aurora.":nv=2:pr=2
14190 print"▒▒▒Ora dovete tornare all'a
stronave e completare la vostra ";
14200 print"missione. Non dimenticate di
 prendere aurora!":goto4320";"
14210 gosub18000:print"▒Mm! Bene. Ma dov
reste esercitare la mente ";
14220 print"al gioco ";n$:goto4320
```

Linee 7730-7750: In risposta al comando BACIA queste tre linee fanno fronte a tutte le possibili eventualità. La prima è per baciare Aurora all'interno della capanna, la seconda di baciare in qualunque altra parte del gioco, e la terza di baciare chiunque o qualunque cosa.

Linee 7800-7810: Se cercate di abbracciare Aurora la risposta è "NON ABBASTANZA CORDIALE – PROVA QUALCOS'ALTRO". Se voi provate a baciare qualcun'altro o qualcos'altro, allora la linea 7810 ritorna con "TI STAI COMPORTANDO IN MODO UN PO' STRANO". Questa linea sembra essere apprezzata dalla gente che ho visto giocare.

Linee 14000-14020: Queste linee danno una risposta se voi provate a baciare Aurora con la tuta spaziale indossata. C'è un sottile suggerimento nella linea 14020 – "lei sta indicando la sua bocca" che potrebbe significare molte cose, ma il significato di riferimento al comando "mangiare noci".

Linee 14030-14220: Eccoci! Il bacio! Prima abbiamo un effetto sonoro simile ad una scala che sale e poi scende, quindi le parole "UOMO STRAORDINARIO" lampeggiano cinque volte sullo schermo. La linea 14110 contiene il reale "UOMO STRAORDINARIO", ma è stato scritto con simboli grafici. Premete shift.

Finalmente appare un messaggio circa gli alieni diventati amici e voi potete partire con Aurora.

Linee 14210-14220: Sono giusto qualche cosa in più – nel caso in cui decidiate di baciare Aurora di nuovo in qualunque punto del programma.

Prova dei moduli 14.1-14.5
Non avete bisogno di provare questi moduli separatamente perché avete finito il gioco. Tutto ciò che dovete fare è trovare rapidamente la vostra strada per la capanna, salvare su nastro, quindi provare a congratularvi, abbracciare, baciare, ecc., sia dentro che fuori dalla capanna. Devono arrivare le risposte appropriate.

Sommario

Questo capitolo vuol farvi capire l'importanza di fare sempre quel qualche cosa in più ogni volta – in questo caso il bacio. Ho trovato che aggiungere la musica e le immagini non comporta troppo lavoro e tempo nella correzione degli errori e nelle prove. E da al vostro programma questa raffinatezza.
La maratona è finita, il gioco è completo. Rilassatevi e provatelo per un po' decidendo se volete cambiare qualche linea e aggiungere il vostro tocco personale prima di provarlo sui vostri amici. Soprattutto Buon Divertimento.

CAPITOLO 15

GUARDANDO AL FUTURO

Estendere il 'Pianeta dell'Incubo'

Adesso che avete finito il libro non dovete fermarvi qui. Potete provare ad estendere il programma. Potete aggiungere o toglier Locazioni. Potete render possibile giocare come uomo o come donna (provando ad agguingere alcune variabili in più come cambiare 'Aurora' con P$), quindi avere l'input "sei maschio o sei femmina?" all'inizio e cambiare le variabili di conseguenza. Ci sono diverse possibilità aperte.
Mi vengono alcune idee mentre sto scrivendo. Magari il vostro eroe potrebbe avere delle bombole di ossigeno che sono in pericolo di svuotarsi – a meno che egli non trovi altro ossigeno da qualche parte nell'Adventure. Cosa ne pensate di un indizio nello stagno in quella radura della foresta? Magari la gemma potrebbe essere utile dopo tutto... a salvare Aurora! Sono sicuro che potrete pensarne ancora.
Ho già menzionato il problema della memoria – ma se togliete tutte le istruzioni REM trovate che vi rimangono ancora un po' di byte extra. Perché non creare la sequenza dei titoli e poi ricaricare il programma appena dopo questi? Perché non avere una sezione completa dell'Adventure sottoterra (passaggi che portano ad una città sotterranea che raggiungete andando "giù" nelle rovine del tempio) che viene caricata dopo che voi andate in una certa locazione del programma. Le possibilità sono limitate solo dalla vostra mente ed immaginazione.
Il 64 può darvi grafici, sprite e suoni. Usateli.

Tornando a voi

Solo voi potete sedervi e creare la vostra trama, scrivere la vostra storia, programmare il vostro Adventure. Non sconcertatevi dalla lunghezza del mio programma – il vostro primo può essere metà della lunghezza, anche un decimo. Ciò che importa è che voi proviate. C'è una fortissima sensazione di realizzazione quando cominciate a scrivere le vostre Istruzioni, e cominciate a mettere e togliere cose. Io so che voi potete ricordare il momento in cui il vostro primo programma è andato bene. È come sentire per la prima volta un disco che vi colpisce – e quando lo risentite e ricordate la prima volta, avete quasi la sensazione di tornare indietro nel tempo, di risentire il profumo dell'aria e rivedere il luogo. È come la prima volta che siete andati in bicicletta, ma meglio, perché il programma è proprio vostro – nessun altro ha mai scritto esattamente le stesse cose. È strano ma vero! Il vostro primo Adventure vi darà più soddisfazione (molta di più) del completamento di "Il Pianeta Dell'Incubo". E non preoccupatevi dell'abilità nel Programmare – io stesso, quando ho cominciato a scrivere questo libro non ero molto bravo – ma ciò non è veramente una difficoltà. Se scrivete una storia veramente buona mandatemela al Sunshine Books – io sono ancora fondamentalmente un giocatore col gusto per gli Adventure.

CONCLUSIONE

Questo libro è stata un'impresa ritemprante per me – ed anche una bella esperienza. Ho tentato molte e molte volte, di cercare ed eliminare gli errori, sia nella parte programmatoria che nel testo. Ne avrò inevitabilmente lasciato qualcuno.
I miei più sentiti ringraziamenti vanno a tutti coloro che mi hanno aiutato nel giocare e scrivere il programma. Quando cominciai non sapevo se ce l'avrei fatta, ma così è stato. Ho imparato a conoscermi, come si scrive un'Adventure e a migliorare le mie capacità di programmazione lungo la via.
Considero anche le Adventure il mio gioco preferito. Sembra che sia così anche per molte altre persone. Divertitevi a creare la vostra.

APPENDICI

APPENDICE A

TABELLA DELLE VARIABILI PER IL PIANETA DELL'INCUBO

V	L'indirizzo di partenza del chip video (V = 53248)
SC	L'indirizzo di partenza del chip sonoro (SC = 54272)
BS	L'indirizzo di partenza del colore del bordo dello schermo (BS = 53280)
BC	L'indirizzo di partenza del colore del centro dello schermo (BC = 53281)
SS	L'indirizzo di partenza dello schermo (SS = 1024)
CS	L'indirizzo di partenza della memoria colore (CS = 55296)
VO	Volume del suono (VO = 54296)
W1	Forma d'onda della voce 1 (W1 = 54276)
W2	Forma d'onda della voce 2 (W2 = 54283)
W3	Forma d'onda della voce 3 (W3 = 54290)
A1	Attacco/Decadimento per la voce 1 (A1 = 54277)
A2	Attacco/Decadimento per la voce 2 (A2 = 54284)
A3	Attacco/Decadimento per la voce 3 (A3 = 54291)
S1	Mantenimento/Rilascio per la voce 1 (S1 = 54278)
S2	Mantenimento/Rilascio per la voce 2 (S2 = 54285)
S3	Mantenimento/Rilascio per la voce 3 (S3 = 54292)
H1	Alta frequenza per la voce 1 (H1 = 54273)
H2	Alta frequenza per la voce 2 (H2 = 54280)
H3	Alta frequenza per la voce 3 (H3 = 54287)
L1	Bassa frequenza per la voce 1 (L1 = 54272)
L2	Bassa frequenza per la voce 2 (L2 = 54279)
L3	Bassa frequenza per la voce 3 (L3 = 54286)
PR	Posta a 2 quando la Principessa viene salvata
CV	Posta a 2 quando viene trovato il cristallo
SV	Posta a 2 quando viene ucciso il serpente
EV	Posta a 2 quando viene uccisa l'anguilla
QV	Posta a 2 quando vengono superate le sabbie mobili
DV	Posta a 2 quando è stato ucciso il Dinosauro
NV	Posta a 2 quando i selvaggi sono amichevoli
FV	Sono state mangiate le noci
ES	Posta a 2 quando viene uccisa l'anguilla
	Posta a 1 quando l'anguilla vi impedisce di muovervi
NP	Posta a 1 quando siete stati una volta nella capanna
NA	Posta a 1 per consentire di muoversi dalla capanna
Z	Posta a 1 quando sono state mangiate le noci
RC	Posta a 2 per impedire il movimento dalla locazione 28
HH	Posta a 2 quando Aurora è nell'astronave

YY	Posta a 1 quando l'anguilla impedisce la risposta al comando "GUARDA"
HR	Posta a 2 quando Victor non vi aiuterà
P	La posizione della locazione corrente
P2	La posizione della nuova locazione
OB%	Valore intero di un oggetto nel vettore degli oggetti
OB$	Descrizione in una parola dell'oggetto
SI$	Descrizione lunga dell'oggetto della locazione
A$	Variabile locale per attendere l'input
N$	Nome del giocatore
I$	Comando atteso dal giocatore
VE$	Verbo atteso in un comando di due parole
NO$	Sostantivo in un comando di due parole
PI	Variabile locale usata per separare I$
SP	Variabile locale usata per separare I$
M	Variabile locale usata per creare un loop di temporizzazione per la pausa
G	Variabile locale usata per aggiungere oggetti nell'inventario
I	Variabile locale usata per contare gli oggetti nel vettore egli oggetti

APPENDICE B

I NUMERI DI LINEA

TITOLI DI TESTA	130		
SIETE MORTO	190		
VARIABILI	280	DATA PER GLI OGGETTI	3700
EFFETTI SONORI	360	POSIZIONE E LOCAZIONI	4000
TITOLO PAGINA	1000	LOCAZIONE DEL-L'OGGETTO	4150
SCRITTURA TITOLO	1300	PUNTATORI PERICOLI	4200
RUMORE ASTRONAVE	2030	USCITA	4320
'S' PER INIZIARE	2100	INPUT "ISTRUZIONI"	5000
ROTTURA ASTRONAVE	2170	LOOP DEL-L'ANGUILLA	5070
SCRITTE DI VICTOR	2400	LOOP DELLE SABBIE MOBILI	5200

LOOP DEI NATIVI	5290
NESSUNA USCITA	6000
IMPEDISCE L'USCITA	6110
LOCAZIONI	20000
LOCAZIONI SPECIALI	21960
DESCRIZIONE DEL SERPENTE	30000
DESCRIZIONE DEL-L'ANGUILLA	30500
DESCRIZIONE DELL'ARIA VELENOSA	31000
SE TOGLI TUTA	31500
SE MANGI NOCI	31600

DESCRIZIONE CRISTALLO	32000
NON PUOI PRENDERE CRISTALLO	32070
SCALA ALBERO	32500
DESCRIZIONE SABBIE MOBILI	32600
DESCRIZIONE DINOSAURO	32700
DESCRIZIONE NATIVI	33000
RITORNO ALLA CAPANNA	33060
PRESA ILLEGALE DI AURORA	33120
TROVATO AURORA	33240
TROVATO CRISTALLO	33380
NIENTE CRISTALLO	33440

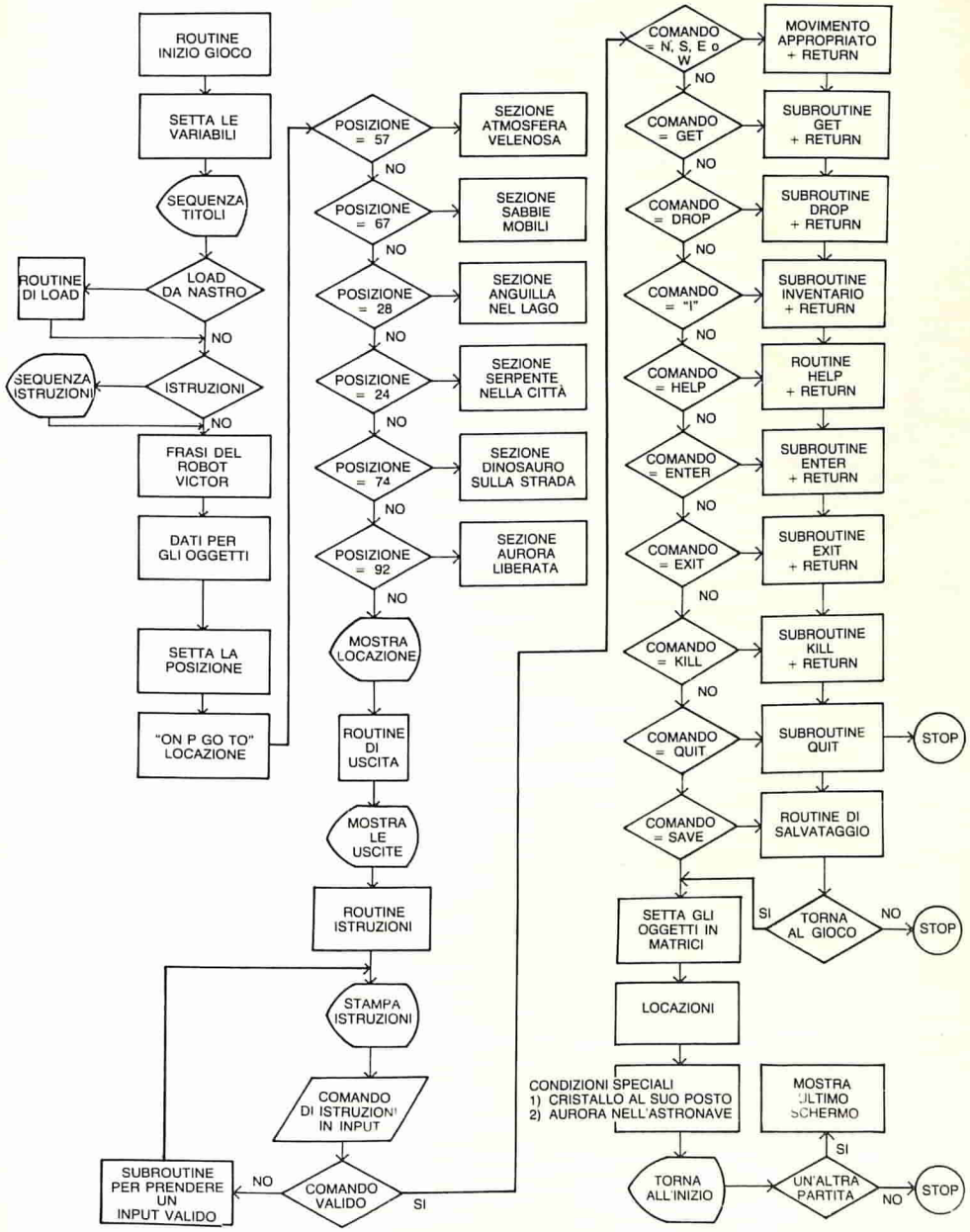
SPRITE DEL MEALY BUG	35000		
SPRITE DI VICTOR	35100	MOVIMENTO	7200
DATA DEL MEALY BUG	35240	CERCA SPAZI IN I$	7300
DATA DI VICTOR	35330		
SETTA SPRITES	35420	I$ IN VE$ + NO$	7420
COMANDI I$	6500		
DIREZIONE DI BYPASS	7030	COMANDI VERBI	7500
CAMERA DI DECOMPRES-SIONE	7070		

		PERICOLO ANGUILLA	12500
		FIRE + STAB	12700
INVENTARIO	10000		
AIUTO	11000	PERICOLO SERPENTE	13000
USA	11500		
PRENDI	11700	PERICOLO SABBIE MOBILI	13300
LASCIA	12000	PERICOLO DINOSAURI	13500
		PERICOLO NATIVI	13700
		BACIO ALLA PRINCIPESSA	14000
UCCIDI	12300		

		PIP	18000
ISTRUZIONI GENERALI	15000	BURP	19000
CARICA	16000		
SALVA	17000		
FINISCI	17500		

APPENDICE C

DIAGRAMMA DI FLUSSO COMPLETO PER IL PIANETA DELL'INCUBO



APPENDICE D

PROGRAMMA A COMPETO

```
1 goto3
2 save"@0:modulo 1.10",8:verify"modulo 1
3 rem
99 rem*************************************
**********************
100 rem   gioco d'avventura
101 rem*************************************
**********************
107 :
108 :
109 rem*************************************
**********************
110 rem  inizializza le variabili
111 rem*************************************
**********************
120 sw%=0:mw%=0
147 :
148 :
149 rem*************************************
**********************
150 rem  Predispone le matrici per gli o
ggetti
151 rem*************************************
**********************
160 dim ob%(5),ob$(5),si$(5)
170 for i=1 to 3:read ob%(i),ob$(i),si$(
i):next
180 data 1,coltello,qui c'e' un coltello
190 data 7,fucile,il vostro fucile e' qu
i
200 data 13,gioiello,per terra c'e' un g
ioiello
207 :
208 :
209 rem*************************************
**********************
210 rem   setta la posizione
```

```
211 rem*************************************
*********************
220 p=7
230 goto 250
235 :
236 :
237 rem*************************************
*********************
238 rem   setta la Posizione
239 rem*************************************
*********************
240 p=p2
250 if p<11 then on p goto 6000,6020,0,0
,0,0,6040,6060,0,0
260 if p<21 then on p-10 goto 0,0,6080,0
,0,0,0,6100,0,0
270 if p<26 then on p-20 goto 0,0,0,0,0
287 :
288 :
289 rem*************************************
*********************
290 rem Per stampare l'oggetto nella giu
sta Posizione
291 rem*************************************
*********************
300 for i=1 to 3:if ob%(i)=p then print
si$(i)
310 next
387 :
388 :
389 rem*************************************
*********************
390 rem  ricerca le Posizioni coi perico
li
391 rem*************************************
*********************
400 if p=2 and sw%<>2 then 7000
410 if p=18 and mw%<>2 then 7020
487 :
488 :
489 rem*************************************
*********************
490 rem  mostra le opzioni sulla direzio
```

```
ne
491 rem*********************************
*********************
500 print"puoi andare a ";
510 if n>0 then print" nord";
520 if e>0 then print" est";
530 if s>0 then print" sud";
540 if w>0 then print" ovest";
545 :
546 :
547 rem*********************************
*********************
548 rem   subroutine di istruzioni
549 rem*********************************
*********************
550 print chr$(13)
560 input"istruzioni:";i$
587 :
588 :
589 rem*********************************
*********************
590 rem condizioni speciali per evitare
la stampa delle direzioni
591 rem*********************************
*********************
600 if p=2 and sw%<>2 then 900
610 if p=18 and mw%<>2 then 900
687 :
688 :
689 rem*********************************
*********************
690 rem testa le direzioni di una sola l
ettera per i$
691 rem*********************************
*********************
700 if i$="n"ori$="e"ori$="s"ori$="o" th
en 800
715 goto 950
787 :
788 :
789 rem*********************************
*********************
790 rem  subroutine del movimento
```

```
791 rem*****************************************
*********************
800 if i$="n" and n>0 then p2=p-5:goto 2
40
810 if i$="e" and e>0 then p2=p+1:goto 2
40
820 if i$="s" and s>0 then p2=p+5:goto 2
40
830 if i$="o" and w>0 then p2=p-1:goto 2
40
837 :
838 :
839 rem*****************************************
*********************
840 rem  se non ci sono locazioni possib
ili nella direzione
841 rem*****************************************
*********************
850 print:print"spiacente - non puoi pre
ndere quella via !!":goto 250
887 :
888 :
889 rem*****************************************
*********************
890 rem subroutine per evitare movimenti
 per sfuggire al pericolo
891 rem*****************************************
*********************
900 ifi$="n"ori$="e"ori$="s"ori$="o"then
print"non puoi farlo...ora!!":goto 550
937 :
938 :
939 rem*****************************************
*********************
940 rem testa comandi di una sola letter
a per i$
941 rem*****************************************
*********************
950 ifi$="i" then 2000
960 ifi$="a" then 2100
970 ifi$="nuota" or i$="galleggia" and p
=2 then 2400
987 :
```

```
988 :
989 rem*********************************
*********************
990 rem  subroutine di controllo per due
 parole
991 rem*********************************
*********************
1000 for i=1 to len(i$)
1010 if mid$(i$,i,1)=" " then 1100
1020 next
1030 print"prego. puoi usare due parole"
:goto 550
1087 :
1088 :
1089 rem*********************************
*********************
1090 rem subroutine per convertire i$ in
 ve$ e no$
1091 rem*********************************
*********************
1100 for sp=1 to len(i$)
1110 if mid$(i$,sp,1)=" " then 1130
1120 next
1130 ve$=left$(i$,sp-1)
1140 no$=right$(i$,(len(i$)-sp))
1187 :
1188 :
1189 rem*********************************
*********************
1190 rem subroutine per scandire vari co
mandi ve$
1191 rem*********************************
*********************
1200 ifve$="prendi"orve$="acchiappa"orve
$="afferra"orve$="porta"then 2200
1210 ifve$="butta"orve$="perdi"orve$="la
scia"then 2300
1220 ifve$="uccidi"then 2500
1230 print:print"non so come fare '";ve$
;"' ":goto 550
1987 :
1988 :
1989 rem*********************************
```

```
**********************
1990 rem   sezione dell´ inventario
1991 rem*********************************
**********************
2000 Print"il tuo inventario e´:":iv=0
2010 for i=1 to 3
2020 if ob%(i)=-1 then Print ob$(i):iv=i
v+1
2030 next
2040 if iv=0 then Print"niente"
2050 goto 500
2087 :
2088 :
2089 rem*********************************
**********************
2090 rem   sezione di aiuto
2091 rem*********************************
**********************
2100 if P=18 then Print"Puoi tentare di
ucciderlo":goto 550
2110 Print"non c´e´ molto aiuto qui. ho
Paura":goto 550
2187 :
2188 :
2189 rem*********************************
**********************
2190 rem  subroutine di aquisizione
2191 rem*********************************
**********************
2200 for i=1 to 3
2210 if ob$(i)=no$ then 2230
2220 next
2230 ifob%(i)=-1 thenPrint"ce l´hai":got
o 500
2250 ifob%(i)<>P thenPrint"non e´ qui":g
oto 500
2260 Print"ok":ob%(i)=-1
2270 goto 500
2287 :
2288 :
2289 rem*********************************
**********************
2290 rem   subroutine del trabocchetto
```

```
2291 rem*********************************
**********************
2300 for i=1 to 3
2310 if ob$(i)=no$ then 2330
2320 next
2330 ifob%(i)<>-1 thenPrint"non ce l'hai
":goto 500
2340 Print"ok":ob%(i)=P
2350 goto 500
2387 :
2388 :
2389 rem*********************************
**********************
2390 rem subroutine della palude
2391 rem*********************************
**********************
2400 if ob%(3)=-1then Print"ok - sei soP
ravvissuto":sw%=2:goto 500
2410 Print"sei troPPo Pesante senza il m
agico     gioiello.sei affogato!":stoP
2487 :
2488 :
2489 rem*********************************
**********************
2490 rem subroutine dell' uccisione
2491 rem*********************************
**********************
2500 if P<>18 then 2540
2510 if ob%(2)=-1 then Print"ok":mw%=2:g
oto 500
2520 if ob%(1)=-1 then Print"ProPrio con
 un coltello? scherzi !":goto 550
2530 Print"non ho niente con cui uccider
lo!":goto 550
2540 Print"uccidere che cosa?":goto 500
5991 :
5992 :
5993 rem*********************************
**********************
5994 rem  descrizione delle varie luoghi
5995 rem*********************************
**********************
5996 :
```

```
5997 rem*****************
5998 rem    luogo 1
5999 rem*****************
6000 print"▲siete in un campo."
6010 n=0:e=2:s=0:w=0:goto 500
6017 rem*****************
6018 rem    luogo 2
6019 rem*****************
6020 print"▲siete su un sentiero."
6030 n=0:e=0:s=7:w=1:goto 500
6037 rem*****************
6038 rem    luogo 7
6039 rem*****************
6040 print"▲siete in una casa."
6050 n=2:e=8:s=0:w=0:goto 500
6057 rem*****************
6058 rem    luogo 8
6059 rem*****************
6060 print"▲siete in una foresta."
6070 n=0:e=0:s=13:w=7:goto 500
6077 rem*****************
6078 rem    luogo 13
6079 rem*****************
6080 print"▲siete in una pianura."
6090 n=8:e=0:s=18:w=0:goto 500
6097 rem*****************
6098 rem    luogo 18
6099 rem*****************
6100 print"▲siete presso un lago."
6110 n=13:e=0:s=0:w=0:goto 500
6987 :
6988 :
6989 rem*************************************
*********************
6990 rem mostra i pericoli
6991 rem*************************************
*********************
7000 print"siete caduti ................
7010 goto 550
7020 print"un mostro e' apparso sul lago
"
7030 goto550
```

PROGRAMMA B COMPLETO

```
0 :
100 rem
110 rem titoli di testa
120 rem
130 print"■":poke53280,0:poke53281,0
140 print"■■■■scagliati in lontano futu
ro ....."
150 goto280
160 rem
170 rem siete morto
180 rem
190 pokev+21,0:print"■■■■Stavolta lo ave
te superato bene, vero!? ";n$
200 print"■■■Per fortuna posso farvi riv
ivere e rimettervi";
210 print" al via.":print"■■■Volete ? (s
/n)"
220 getr$:ifr$<>"s"andr$<>"n"then220
230 ifr$="s"then280
240 ifr$="n"then17570
250 rem
260 rem variabili
270 rem
280 clr:v=53248:sc=54272:bs=53280:bc=532
81
290 pr=0:cv=0:sv=0:ev=0:qv=0:dv=0:es=0:n
v=0:na=0:np=0:fv=0:rc=0:yy=0:hh=0:hr=0
300 vo=54296:w1=54276:a1=54277:s1=54278:
w2=54283:a2=54284:s2=54285
310 w3=54290:a3=54291:s3=54292
320 h1=54273:l1=54272:h2=54280:l2=54279:
h3=54287:l3=54286
330 rem
340 rem effetti sonori
350 rem
360 forl=0to24:pokesc+l,0:nextl
370 pokevo,15:pokea1,255:pokea2,255:poke
a3,255
380 pokes1,255:pokes2,255:pokes3,255
```

```
390 Pokew1,17:Pokew2,33:Pokew3,17
400 Pokeh1,37:Pokel1,162:Pokeh2,1:Pokel2
,1:Pokeh3,2:Pokel3,187
410 gosub35420
970 rem
980 rem titolo
990 rem
1000 Pokebs,0:Pokebc,0:Print"■":ss=1024:
cs=55296
1010 Poke53272,21
1020 fori=1to23
1030 Pokecs+40*i,4:Pokess+40*i,102
1040 Pokecs+40*i+38,4:Pokess+40*i+38,102
1050 nexti
1060 Print"■■■
```

```
1320 form=1to200
1330 nextm
1340 Poke53272,23
1350 Printtab(6)"■Siete soli nella sala
di controllo dell´astronave ";
1360 Print"Pensando alla ragazza   che a
vete trasportato a pagamento"
1370 Print"■sul Pianeta Zen: la Principe
ssa che ha segretamente rubato ";
1380 Print"il vostro cuore, ma che e´ pr
omessa al tiranno di Zen";
1390 Print"    in matrimonio Politico"
1400 Printtab(5)"■Ma voi siete solo preo
ccupati di   Aurora, non della vostra ";
1410 Print"abilita´ di    pilota. All´im
provviso e senza nessun   segnale vi";
1420 Print" dirigete verso lo spazio
 Reale troppo vicini ad un misterioso ";
1430 Print"   pianeta - e siete risucchi
ati dalla     gravita´."
1440 Printtab(5)"■Forse la vostra mente
non e´ concentrata -";
1450 Print"Forse il pianeta e´ troppo fo
rte -Non c´e´ modo di controllare ";
1460 Print"la vostra astronave. Mentre l
´ultimo ricordo e´  il suolo ";
1470 Print"che s´avvicina velocissimo...
.."
2000 rem
2010 rem Preliminari
2020 rem
2030 forl=0to24:Pokesc+l,0:nextl
2040 Pokevo,15:Pokea1,255:Pokea2,255:Pok
ea3,255
2050 Pokes1,128:Pokes2,128:Pokes3,128:Po
kew1,65:Pokew2,129:Pokew3,129
2060 Pokeh1,37:Pokel1,162:Pokeh2,1:Pokel
2,12:Pokeh3,2:Pokel3,187
2100 Printtab(11)"■Premi ´s´ per iniziar
e"
2110 geta$:ifa$<>"s"then2110
2120 ifa$="s"thenPokevo,0
2130 form=1to150:nextm
```

```
2150 rem
2160 rem caduta
2170 forl=0to24:Pokesc+l,0:nextl
2180 Pokevo,15:Pokea1,140:Pokea2,140:Pok
ea3,140:Pokes1,0:Pokes2,0:Pokes3,0
2190 Pokew1,35:Pokew2,129:Pokew3,129
2200 Pokeh1,37:Pokel1,162:Pokeh2,1:Pokel
2,12:Pokeh3,2:Pokel3,187
2210 Print"■":Pokebs,0:Pokebc,0
2220 Print"■■■■■■■Penso che vi siate sch
iantati !!!"
2230 form=1to1000:nextm
2240 Print"■■■■■■Volete caricare un gioc
o da nastro ? (s o n):"
2250 geta$:ifa$<>"s"anda$<>"n"then2250
2260 ifa$="s"thenPokevo,0:gosub16000
2270 Print"■■■■■Volete istruzioni (s o n
):"
2280 geta$:ifa$<>"s"anda$<>"n"then2280
2290 ifa$="s"thenPokevo,0:gosub15000
2370 rem
2380 rem victor
2390 rem
2400 Pokebs,11:Pokebc,11:Print"■":Print"
Personal Robot Victor informa:"
2410 Print"■Durante questa avventura sar
o' Pronto ad aiutarvi..."
2420 Print"■Di solito appariro' nei mome
nti di      grande tensione e Pericolo!"
2430 Print"Ma vi saro' sempre appresso,
anche se   non mi vedete."
2440 input "■■battete il vostro nome";n$
2450 gosub18000:Print"■Grazie ";n$;". Vi
 terro' d'occhio !"
2460 form=1to1000:nextm
2470 Print"■■■■■Inizia l'Avventura■...."
2480 form=1to2000:nextm
2490 Pokev+21,0
3670 rem
3680 rem data oggetti
3690 rem
3700 fori=1to9:readob%(i),ob$(i),si$(i):
nexti
```

```
3710 data 15,corda,un ratto si dimena pr
eso al laccio nell'angolo
3720 data 28,cristallo,il cristallo ener
getico rifulge di luce nascosta
3730 data 37,noci,ci sono alcune noci pe
r terra
3740 data 47,relitto,inciampate in un ro
ttame
3750 data 53,gemma,il riflesso di una ge
mma vi attrae
3760 data 58,tuta-spaziale,la tuta-spazi
ale e' qui ancora intatta
3770 data 59,mitra,il vostro mitra e' pe
r terra
3780 data 69,coltello,tra i sassi c'e' u
n coltello preistorico
3790 data 92,aurora,vedete aurora sana e
 salva
3970 rem
3980 rem p
3990 rem
4000 p=59:goto4020
4010 p=p2
4020 ifp<11thenonpgoto20000,20020,20080,
20130
4030 ifp<21thenonp-10goto20170,20190,202
10,20250,20310
4040 ifp<31thenonp-20goto0,20360,0,20380
,20430,20480,0,20530
4050 ifp<41thenonp-30goto0,0,0,0,0,20570
,20620,20670
4060 ifp<51thenonp-40goto0,20710,20730,2
0750,0,20770,20820,20870
4070 ifp<61thenonp-50goto0,20930,20950,2
1000,21040,21100,21150,21200,21390
4080 ifp<71thenonp-60goto0,21510,21530,2
1550,21600,0,21650,21700,21750
4090 ifp<81thenonp-70goto0,0,0,21800
4100 ifp<91thenonp-80goto0,0,0,21860
4110 ifp<101thenonp-90goto0,21910,21940,
21970
4120 rem
4130 rem vedi oggetti
```

```
4140 rem
4150 for i=1 to 9: if ob%(i)=p then print tab i$(i)
4160 next i
4170 rem
4180 rem puntatori pericoli
4190 rem
4200 if p=24 and sv<>2 then goto 30000
4210 if p=28 and ev<>2 then gosub 30500
4220 if p=57 and fv<>2 and ob%(6)<>-1 then goto
31000
4230 if p=59 and ob%(2)<>-1 then gosub 32000
4240 if p=59 and pr<>2 then gosub 33240
4250 if p=67 and qv<>2 then goto 32600
4260 if p=74 and dv<>2 then goto 32700
4270 if p=92 and nv<>2 and p=0 then goto 33000
4280 if p=92 and nv<>2 and p=1 then goto 33060
4290 rem**
4300 rem uscite
4310 rem**
4320 print"[illegible]Potete andare a";
4330 if n>0 then print" nord";
4340 if e>0 then print" est";
4350 if s>0 then print" sud";
4360 if w>0 then print" ovest";
4970 rem**
4980 rem istruzioni
4990 rem**
5000 print chr$(13): x=fre(0)
5010 input"[illegible]Istruzioni:[illegible]";i$
5020 poke v+21,0
5030 if i$="*" then print"[illegible]": goto 5010
5040 rem**
5050 rem loop anguilla
5060 rem**
5070 if p=92 then 5290
5080 if p=67 then 5200
5090 if p<>28 then 6000
5100 if ev=2 then 6000
5110 if p=28 and es=2 then 6500
5120 q=q+1
5130 if q<5 then 6000
5140 if q=5 then print"[illegible]stato[illegible] veramente i
ndebolendovi...": goto 5120
```

```
5150 if q=10then gosub 19000:print"■Ecco ci
o' che vi capita! ";n$
5160 if q=10then print"Siete stati stritol
ati a morte.":goto190
5170 rem**
5180 rem loop palude
5190 rem**
5200 if p=92then5290
5210 if p<>67then6000
5220 if qv=2then6500
5230 q=q+1
5240 if q<5then6000
5250 if q=5then13320
5260 rem**
5270 rem loop indigeni
5280 rem**
5290 if p<>92then6000
5300 if nv=2then6000
5310 if p=92and pr=2then6500
5320 q=q+1
5330 if q<5then6000
5340 if q=5then gosub18000:print"■Tenta ";
n$:print"Ed ora cosa ?":goto5320
5350 if q=10and i$="bacia aurora"then14030
5370 if q=10then gosub18000:print"■troppo
tardi! Gli indigeni vi hanno scacciato";
5380 if q=10then print" fuori dalla capann
a - non potete prendere aurora ";
5390 if q=10then print"con voi. Meglio and
ar via e pensare ad altro ";
5400 if q=10then print"per liberarla.":q=0
:na=1:np=1:goto4320
5970 rem**
5980 rem nessuna uscita
5990 rem**
6000 if p=24and sv<>2then6110
6010 if p=28and es=2then6500
6020 if p=28and ev<>2then6110
6030 if p=67and qv<>2then6110
6040 if p=74and dv<>2then6170
6050 if p=92and nv<>2and na=1then6500
6060 if p=92and nv<>2and na=0then6110
6070 goto6500
```

```
6110 ifi$="n"ori$="e"ori$="s"ori$="o"the
n9osub19000:9oto6130
6120 9oto6150
6130 Print"■Non Potete fu99ire di 9ui!":
Print"■Dovete Pensare a ";
6140 Print"■9ualcos´altro!":9oto5000
6150 ifi$="corri"then9osub19000:Print"■N
on c´e´ uscita di li´.":9oto5000
6160 ifi$="cammina"then9osub19000:Print"
Non c´e´ uscita di li´.":9oto5000
6170 ifi$="sud"ori$="s"then9osub19000:Pr
int"■Niente da fare ";n$
6180 ifi$="sud"ori$="s"thenPrint"■C´e´ u
n dinosauro sulla via!":9oto5000
6470 rem**
6480 rem i$
6490 rem**
6500 ifi$="nord"theni$="n"
6510 ifi$="est"theni$="e"
6520 ifi$="sud"theni$="s"
6530 ifi$="ovest"theni$="o"
6540 ifi$="n"ori$="e"ori$="s"ori$="o"the
n7070
6550 ifi$="i"orleft$(i$,3)="inv"then1000
0
6560 ifi$="a"ori$="aiuto"then11000
6570 ifi$="istruzioni"then9osub15000
6580 ifi$="corri"ori$="cammina"then9osub
18000:Print"■Quale via?":9oto4320
6590 ifi$="9"orleft$(i$,6)="9uarda"then9
osub18000:9oto6610
6600 9oto6640
6610 ifP=28andyy=1thenPrint"■non vedete
molto sotto ac9ua!!":9oto5000
6620 9oto4020
6640 ifleft$(i$,4)="salt"andP=67then1331
0
6650 ifleft$(i$,4)="tira"andP=67then1331
0
6660 ifleft$(i$,5)="9alle"andP=67then133
10
6670 ifleft$(i$,5)="nuota"andP=67then133
10
```

```
6680 ifi$="mangia"thenprint"Cosa vi piac
erebbe mangiare?":goto5000
6690 ifleft$(i$,11)="apri camera"andp=57
then7100
6700 ifleft$(i$,11)="apri camera"andp=58
then7100
6710 ifi$="vai lago"andp=28andrc=2then30
650
6720 ifleft$(i$,5)="nuota"andp=28andrc=2
then30650
6730 ifleft$(i$,5)="tuffa"andp=28andrc=2
then30650
6740 ifi$="vai lago"andp=28then30510
6750 ifi$="vai riva"andp<>28then7810
6760 ifi$="vai riva"andp=28then6780
6770 goto6800
6780 ifes=2thengosub18000:goto20540
6790 ifes<>2thengosub19000:goto6130
6800 ifleft$(i$,5)="nuota"andp=28then305
10
6810 ifleft$(i$,5)="tuffa"andp=28then305
10
6820 ifleft$(i$,3)="vai"thengosub18000:p
rint"Quale direzione?":goto4320
6830 ifleft$(i$,6)="vaffan"thengosub1900
0:print"E voi ";n$:goto5000
6840 ifi$="save"then17000
6850 ifi$="esci"ori$="fine"thengoto17500
6860 ifi$="scala"thengosub18000:print"C
osa volete che scali ?":goto5000
7000 ifi$<>"n"andi$<>"e"andi$<>"s"andi$<
>"o"then7300
7040 rem**
7050 rem airlock - camera decompressione
7060 rem**
7070 ifp=58andi$="o"then7100
7080 ifp=57andi$="e"then7100
7090 goto7200
7100 print"Si apre l'airlock..."
7110 forl=0to24:pokesc+l,0:nextl
7120 pokevo,15:pokea1,17:pokea2,17:pokes
1,130:pokes2,130
7130 pokew1,17:pokew2,17:pokeh1,13:pokel
```

```
1,78:Pokeh2,253:Pokel2,46
7140 fort=1to1500:nextt:Pokevo,0:Print"▒
                          ":rem 20 sPazi
7150 fort=1to700:nextt:Print"▒airlock aP
erta....."
7160 fort=1to700:nextt:goto7200
7170 rem**
7180 rem movimento
7190 rem**
7200 ifi$="n"andn>0thenP2=P-10:gosub1800
0:goto4010
7210 ifi$="e"ande>0thenP2=P+1:gosub18000
:goto4010
7220 ifi$="s"ands>0thenP2=P+10:gosub1800
0:goto4010
7230 ifi$="o"andw>0thenP2=P-1:gosub18000
:goto4010
7240 ifi$="aPri airlock"andP=57thenP2=P+
1:gosub18000:goto4010
7250 ifi$="aPri airlock"andP=58thenP2=P-
1:gosub18000:goto4010
7260 gosub19000:Print"▒Piacente - non P
otete andare di li'":goto4320
7270 rem**
7280 rem cerca uno sPazio su i$
7290 rem**
7300 fori=1tolen(i$)
7310 ifmid$(i$,i,1)=" "then7340
7320 nexti
7330 Print"▒▒Potreste usare due Parole?":
gosub19000:goto5000
7340 fori=1tolen(i$)
7350 ifmid$(i$,i,2)="  "then7380
7360 nexti
7370 goto7420
7380 Print"▒Solo ▒uno▒ sPazio tra Parole
":gosub19000:goto5000
7390 rem**
7400 rem i$-ve$&no$
7410 rem**
7420 forsP=Pi+1tolen(i$)
7430 ifmid$(i$,sP,1)=" "then7450
7440 nextsP
```

```
7450 ve$=left$(i$,sp-1)
7460 no$=right$(i$,(len(i$)-sp))
7480 rem ve$
7490 rem**
7500 ifve$="scala"then7920
7520 ifve$="prendi"andp=28then32070
7530 ifve$="afferra"andp=28then32070
7540 ifve$="acchiappa"andp=28then32070
7560 ifve$="trova"thengosub18000:print"▌
Dovete guardare in giro.":goto5000
7570 ifve$="prendi"andp=92then33120
7580 ifve$="afferra"andp=92then33120
7600 ifve$="acchiappa"andp=92then33120
7610 ifve$="prendi"orve$="afferra"orve$=
"acchiappa"orve$="porta"then11700
7620 ifve$="esamina"thengosub18000:goto7
820
7630 ifve$="butta"orve$="perdi"orve$="la
scia"orve$="togli"then12000
7640 ifve$="uccidi"orve$="sfascia"orve$=
"distruggi"orve$="rovina"then12300
7650 ifve$="mangia"then31600
7660 ifve$="usa"then11500
7670 ifve$="fuoco"then12700
7680 ifve$="spara"orve$="mitraglia"then1
2300
7690 ifve$="pugnala"orve$="taglia"then12
760
7700 ifleft$(ve$,4)="corr"orleft$(ve$,4)
="camm"thenprint"▌quale via ?":goto4320
7710 ifve$="getta"andp=67thengosub18000:
goto13300
7720 ifve$="getta"thengosub18000:goto786
5
7730 ifve$="bacia"andp=92andno$="aurora"
then14000
7740 ifve$="bacia"andob%(9)=-1andno$="au
rora"then14210
7750 ifve$="bacia"thengoto7810
7760 ifp<>92then7780
7770 ifve$="abbraccia"orve$="ama"orve$="
stringi"thengoto7800
7780 ifve$="abbraccia"orve$="ama"orve$="
```

```
strin9i"then9oto7810
7790 9oto7980
7800 9osub18000:Print"■Non e' abbastanza
 socievole. Tenta dell'altro.":9oto5000
7810 9osub19000:Print"■A9ite in modo un
Po' strano ";n$:9oto5000
7820 ifP=25andleft$(no$,4)="Port"thenPri
nt"■solo un Palazzo vuoto..":9oto5000
7830 ifP=28andno$="la9o"thenPrint"■Il cr
istallo e' ProPrio 9ui":9oto5000
7840 ifP=37andno$="cesPu9lio"thenPrint"■
Le noci sembrano buone.":9oto5000
7850 ifP=37andno$="frutti"thenPrint"■Sem
brano commestibili.":9oto5000
7860 Print"■Non c'e' nulla di utile 9ui.
":9oto5000
7865 ifno$="victor"thenhr=2:9oto11300
7870 fori=1to9:ifob$(i)=no$then9oto7900
7880 nexti
7890 Print"■Smettila di Perder temPo!!":
9oto5000
7900 ifob%(i)=-1thenPrint"■Non Potete ve
dere molto 9ui!!":9oto12090
7910 Print"■Non l'avete!":9oto5000
7920 ifno$<>"albero"then7970
7930 ifP=42orP=43orP=44then9oto32500
7940 ifP=52orP=53orP=54orP=55then9oto325
20
7950 ifP=62orP=63orP=64orP=65then9oto325
40
7960 ifP=74then13580
7970 9osub19000:Print"■Non Potete proPri
o scalare ";no$:9oto5000
7980 9osub19000:Print"■Piacente! Ma non
 so come ":Printve$:9oto5000
9970 rem**
9980 rem inventario
9990 rem**
10000 Print:Print"■L'inventario e':":iv=
0
10010 fori=1to9
10020 ifob%(i)=-1thenPrint"■"ob$(i):9osu
b18000:iv=iv+1
```

```
10030 nexti
10040 ifiv=0thenPrint"■■Niente - sPiacent
e."
10050 9oto5000
10970 rem**
10980 rem aiuto
10990 rem**
11000 ifhr=2then11250
11005 ifP=24then9osub35100:Print"■":9oto
11090
11010 ifP=58then9osub35100:Print"■":9oto
11100
11020 ifP=67then9osub35100:Print"■":9oto
11120
11030 ifP=92then9osub35100:Print"■":9oto
11140
11040 ifP=1orP=2orP=11orP=12orP=22then9o
sub35100:Print"■":9oto11190
11050 ifP=42orP=62orP=63then9osub35100:P
rint"■":9oto11170
11060 9osub35100:Print"■"
11070 Print"■Ho un Piccolo Problema che
mi turba ";n$
11080 Print"Tentare di tener testa al vo
stro Per un Po´...":9oto5000
11090 Print"■Tenta di ucciderlo ";n$:Pri
nt"Prima che lo faccia lui!!":9oto5000
11100 Print"■Ora Pensa seriamente ";n$:P
rint"Ci Puo´ essere 9ualcosa intorno ";
11110 Print"che Puo´ aiutarvi...":9oto50
00
11120 Print"■Situazione difficile,9uesta
 ";n$:Print"Su99erisco di buttare";
11130 Print" 9ualcosa Per lanciarvi fuor
i.":9oto5000
11140 Print"■■■Poiche´ 9li indi9eni Pens
ano che aurora sia la loro re9ina ";
11150 Print"e´ me9lio se vi fate vedere
suo amico.":9oto5000
11170 Print"■Ci siamo Perduti nella fore
sta, eh? Perche´ non scalate un ";
11180 Print"albero e 9uardate intorno.":
9oto5000
```

```
11190 Print"Sembriamo Perduti entrambi
";n$:Print"Meglio ";
11200 Print"incamminarci, alla fine trov
erete una via.":goto4320
11250 Print"":gosub35100:Print"Non sia
mo amici, Per il momento ";n$:goto11310
11280 Print"":gosub35100:Print"Non Pot
ete uccidermi ";n$
11290 Print"Sono il nuovo modello invul
nerabile. Ma...":goto11310
11300 Print"":gosub35100:Print"Cio' no
n e' carino ";n$
11310 Print"Non Posso aiutarvi Piu'.";
11320 Print" Non Potete farvi fronte da
soli.":goto5000
11470 rem**
11480 rem uso
11490 rem**
11500 ifP<>24orP<>74andleft$(no$,5)="mit
ra"then11580.
11510 ifP<>24orP<>74goto11610
11520 ifP=24orP=74andleft$(no$,5)="mitra
"then11580
11530 ifP=24orP=74andno$="coltello"then1
1550
11540 goto11660
11550 ifob%(8)<>-1thengosub19000:Print"
Non l'avete!!":goto5000
11560 ifob%(8)=-1thengosub35100:Print""
:Print"ProPrio un coltello ";n$;"?"
11570 ifob%(8)=-1thenPrint"E' un Po' am
bizioso, non vi Pare?":goto5000
11580 ifob%(7)<>-1thengosub19000:Print"
Non l'avete!!":goto5000
11590 ifP=24andob%(7)=-1thengosub18000:p
rint"Bella Pensata!":goto13030
11600 ifP=74andob%(7)=-1thengosub35100:p
rint"":Print"Bella Pensata!":goto13510
11610 fori=1to9
11620 ifob$(i)=no$then11630
11630 ifob%(i)=-1$then11660
11640 nexti
11650 gosub19000:Print"Non ha senso usa
```

```
re cio'!":goto5000
11660 print"■Come Pensate di usare tale
";no$:gosub18000:goto5000
11670 rem**
11680 rem Prendere
11690 rem**
11700 fori=1to9:ifob$(i)=no$then11740
11710 nexti
11720 ifP=37andleft$(no$,5)="frutt"then1
1860
11730 ifob$(i)<>no$then11850
11740 ifob%(i)=-1thenPrint"■Lo avete gia
!!":gosub19000:goto5000
11750 ifP=37andno$<>"noci"then11800
11760 ifP=37andleft$(no$,5)<>"frutt"then
11800
11770 ifP=37andno$="noci"then11790
11780 goto11800
11790 ifP=37andz=1then11880
11800 ifob%(i)<>PthenPrint"■Non vedo ";n
o$;" qui.":gosub19000:goto5000
11810 ifg>4thenPrint"■State raccogliendo
 troPPo - "
11820 ifg>4thenPrint"Dovete lasciare qua
lcosa!!":gosub19000:goto5000
11830 Print"■Okay.":gosub18000:ob%(i)=-1
:g=g+1
11840 goto5000
11850 gosub18000:Print"■Non voglio ProPr
io occuParmene! ";n$;"!!":goto5000
11860 ifz=1then11880
11870 goto11830
11880 gosub18000:Print"■Prova a mangiare
 ";no$:goto5000
11970 rem**
11980 rem abbandonare
11990 rem**
12000 fori=1to9:ifob$(i)=no$then12040
12010 nexti
12020 ifP=28andno$="tuta-sPaziale"thengo
sub19000:goto30510
12030 ifob$(i)<>no$thengosub18000:Print"
Nessun aiuto li'.":gosub5000
```

```
12040 ifob%(i)<>-1thenPrint"Non l'avete
 ancora!!!":gosub19000:goto5000
12050 ifP=28andno$="tuta-spaziale"then go
sub19000:Print"State affondando":goto30
540
12060 ifob%(6)=-1andno$="tuta-spaziale"t
hen31500
12070 ifP=59ando$="aurora"then33350
12080 Print"Okay.":gosub18000
12090 ob%(i)=P:g=g-1:goto5000
12270 rem**
12280 rem uccidere
12290 rem**
12300 ifno$="aurora"thengosub19000:Print
"State scherzando !!!":goto190
12305 ifP=24then13000
12310 ifP=28then12500
12330 ifP=74then13500
12340 ifP=92then13700
12350 gosub18000:Print"Non vedo proprio
 niente da '";ve$;"' ":goto5000
12470 rem**
12480 rem anguilla
12490 rem**
12500 ifve$="spara"orve$="mitraglia"then
12520
12510 ifob%(8)=-1thenPrint"OK. Avete us
ato il coltello.":gosub18000:goto12610
12520 ifob%(7)=-1thengosub18000:Print"I
l mitra non funziona sott'acqua...";
12530 ifob%(7)=-1thenPrint" e dovete esc
ogitare qualcosa di meglio..."
12540 ifob%(7)=-1thenPrint"State diventa
ndo Piu' debole...":goto5000
12550 Print"L'anguilla vi tiene stretto
e vi stritola ";
12560 Print"sempre Piu'... state Perdend
o i sensi..."
12570 form=1to2000:nextm
12580 gosub19000:Print"Maledetti cuoch
i! Siete morto!"
12590 Print"Avreste dovuto estrarre qu
alcosa Per ucciderla!!"
```

```
12600 gosub19000:goto190
12610 Print"■L'anguilla si dibatte amPia
mente nell'agonia - Poi sProfonda ";
12620 Print"nel lago.":gosub18000
12630 Print"Non dimenticate il cristallo
...":ev=2:es=2:rc=0:yy=0:goto5000
12670 rem**
12680 rem far fuoco/Pugnalare
12690 rem**
12700 ifP=24andob%(7)=-1andleft$(no$,5)=
"mitra"then13030
12710 ifP=28andob%(7)=-1andleft$(no$,5)=
"mitra"then12520
12720 ifP=74andob%(7)=-1andleft$(no$,5)=
"mitra"then13510
12730 ifP=92andob%(7)=-1andleft$(no$,5)=
"mitra"then13730
12740 ifleft$(no$,5)="mitra"andob%(7)<>-
1thenPrint"■Non l'avete!":goto5000
12750 gosub18000:Print"■■Non vi sara' d'a
iuto.":goto5000
12760 ifP=28andob%(8)=-1then12500
12770 ifno$="coltello"andob%(8)<>-1thenP
rint"■Non l'avete!":goto5000
12780 goto12750
12970 rem**
12980 rem serPente
12990 rem**
13000 ifob%(7)=-1thenPrint"■■Okay":gosub1
8000:goto13030
13010 ifob%(8)=-1thenPrint"Avete solo un
 coltello - troPPo Poco !":goto5000
13020 Print"■Non avete nulla con cui '";
ve$;"' ...e ora ?":goto5000
13030 Print"■Il serPente cozza contro un
a casa vicina sollevando nuvole di ";
13040 Print"Polvere in aria, nel rantolo
 mortale."
13050 Print"■Contenti di avere questo mi
tra, "n$:sv=2:goto4320
13270 rem**
13280 rem sabbie mobili
13290 rem**
```

```
13300 ifob%(1)=-1andno$="corda"thenPrint
"■scaltra mossa ";n$:goto13305
13302 goto7865
13305 Print"La fune Pende da un albero e
 voi la Prendete. ":gv=2:goto4320
13310 gosub19000:Print"■In questo modo n
on vi state aiutando... ";n$:goto5000
13320 Print"■Affondate sempre Piu´!"
13330 form=1to1000:nextm
13340 gosub19000:Print"Glu... glu... sie
te Perduti":goto190
13470 rem**
13480 rem dinosauro
13490 rem**
13500 ifob%(7)<>-1thengosub18000:goto135
40
13510 gosub19000:Print"■Sebbene sParate
continuamente al meglio - i vostri ";
13520 Print"colPi rimbalzano sulla sua P
elle corazzata !!!"
13530 Print"Cosa ora ";n$;" ?":goto5000
13540 Print"■■Non sembrate in realta´ Pos
sedere il necessario ";
13550 Print"Per uccidere una cosi´ gross
a bestia";
13560 Print"Penso che sarete caPaci di e
vitarlo e di rimanere vivi!! ";
13570 Print"Ma il Problema e´ come suPer
arlo !!":goto5000
13580 Print"■":gosub35100
13590 Print"■Mossa furba, questa ";n$;:P
rint"Ora dovete Passare tra gli alberi "
;
13600 Print"e suPerarlo. Discenderete an
cora Piu´ avanti.":dv=2:goto4320
13670 rem**
13680 rem indigeni
13690 rem**
13700 ifob%(7)=-1then13730
13710 gosub18000:Print"■Sono troPPo lont
ani Per ucciderli! Meglio tentare ";
13720 Print"un´altra via. Forse aurora P
otrebbe aiutare ...":goto5000
```

```
13730 gosub18000:print"Il vostro mitra
ha bisogno di molte ore per ";
13740 print"ricaricarsi dopo la lotta co
l dinosauro.";
13750 print" Dovete pensare a qualcos'al
tro.":goto5000
13970 rem**
13980 rem bacio alla principessa
13990 rem**
14000 ifob%(6)=-1thengosub19000:print"N
on potete baciarla con la tuta";
14010 ifob%(6)=-1thenprint", vero? Avevo
 pensato che lo sapeste!"
14020 ifob%(6)=-1thenprint" Sta indicand
o le sue labbra...":goto5000
14030 forl=0to24:pokesc+l,0:nextl
14040 pokel3,3:pokew3,16:pokesc+3,1:poke
vo,143:pokes1,240:pokesc+4,65
14050 fr=5389:fort=1to200:fq=fr+peek(sc+
27)*225
14060 hf=int(fq/256):lf=fq-hf*256
14070 pokel1,lf:pokeh1,hf
14080 nextt
14090 printchr$(13)
14100 fort=1to5
14110 print"UOMO TERRIBILE !!!"
14120 form=1to500:nextm
14130 print"                    "
14140 form=1to500:nextm
14150 nextt
14160 pokevo,0
14170 print"Ciò non vi ha dato solo
un potere positivo, ma gli indigeni ";
14180 print"vi stanno sorridendo. Potete
 fuggire con aurora.":nv=2:pr=2
14190 print"Ora dovete tornare all'a
stronave e completare la vostra ";
14200 print"missione. Non dimenticate di
 prendere aurora!":goto4320";"
14210 gosub18000:print"Mm! Bene. Ma dov
reste esercitare la mente ";
14220 print"al gioco ";n$:goto4320
14970 rem**
```

```
14980 rem istruzioni
14990 rem**
15000 print"■":pokebs,11:pokebc,11:print
tab(13)"■■Istruzioni■"
15010 print"■■■■Entrate nel mondo della
 fantasia e avventura";
15020 print" col vostro computer come gu
ida. Per superare";
15030 print" questa avventura dovete dar
e i comandi - [illegible] - :"
15040 print"sempre un ■verbo■ e qualche
volta un nome."
15050 print"Per es: - ■Prendi mitra■ o ■
scala albero"
15060 print"■■■Vi verra' data una descri
zione del luogo in cui siete ";
15070 print"e delle direzioni  in cui po
tete andare. Potete usare solo"
15080 print" vai ■nord■, ■sud■, ■est■ o
ovest."
15090 print"■■A volte potete incontrare
un oggetto   che potreste usare ";
15100 print"piu' tardi durante il  gioco
; potete raccoglierlo con ";
15110 print"la parola ■'prendi'■ o ■'aff
erra'. Potete usarlo per  ";
15120 print"■'prendi'■ la vostra tuta o
'porta' la principessa indietro."
15130 printtab(8)"■■■Premi un tasto per
cominciare"
15140 geta$:ifa$=""then15140
15150 print"■■■Potete anche ■'butta'■ o
'togli' un oggetto in ogni momento ";
15160 print" ed esso rimarra' in quel po
sto finche' non lo riprenderete ";
15170 print"quando ne avrete bisogno anc
ora."
15180 print"■■Tasti speciali risparmiera
nno tempo: per es:-"
15190 print"■[illegible] ■e■ ■s■ ■o ■o■ ■sig
nificano ";
15200 print"■nord■, ■est■, ■sud■ o ■oves
t."
```

```
15210 print"■■■■i■ ■vi dara' un ■inventa
rio  de9li  o99etti che racco9liete."
15220 print"■■■■9■ si9nifica ■guarda ■in
torno."
15230 print"■■■■a■ ■si9nifica ■che volet
e aiuto!!"
15240 print"■■■■■battete ■istruzioni■ Per
 avere informazioni 9enerali."
15250 print"■■Se volete salvare il 9ioco
battete save"
15260 print"■■Se volete finire il 9ioco b
attete esci"
15270 print"■■o ■fine■■."
15280 printtab(8)"■■■Premi un tasto"
15290 9eta$:ifa$=""then 15290
15300 print"■■■Usate solo lettere senza ■
shitf:"
15310 print"in altre Parole battete ■■a■
urora non       ■urora."
15320 print"(notate la ■a■■ ,non ■A■■)."
15330 print"■■Su99erimenti:":print"■1. Fe
rmatevi sPesso Per fare il save"
15340 print"2. Tracciate una ■maPPa■■ men
tre esPlorate il Pianeta."
15350 print"3. Siate Precisi nell' ■inve
ntario Potendo Prendere solo";
15360 print" Pochi o99etti alla volta."
15370 print"4. Se volete aiuto battete ■
esamina Per vedere 9li o99etti."
15380 print"5.Ricordate di battere ■istr
uzioni se  rivolete 9uesti messa99i."
15400 print"■■■■■Buona fortuna Per la ric
erca di Aurora";
15410 print"e fate attenzione a9li alien
i!!"
15420 print"■■■■■■■■Batti 'L' Per tornar
e al 9ioco"
15430 9eti$:ifi$<>"l"then15430
15440 ifi$="l"thenPrint"■":return
15970 rem**
15980 rem load (caricamento)
15990 rem**
16000 print"■■■Inserisci il nastro e ria
```

```
vvol9i:"
16010 input"Batti ■RETURN■:";q$
16020 open1,1,0,"planet"
16030 input#1,pr,cv,sv,ev,es,qv,dv,nv,fv
16040 input#1,z,n$,m,i,p,n,e,s,w,i$,pi,s
p
16050 input#1,ve$,no$,q,p2,np,na,rc,yy,h
h,hr
16060 fori=1toq:input#1,ob%(i):nexti
16070 fori=1toq:input#1,ob$(i):nexti
16080 fori=1toq:input#1,si$(i):nexti
16090 close1
16100 gosub35420
16110 print"■":pokebs,11:pokebc,11:goto4
020
16970 rem**
16980 rem save (salvare)
16990 rem**
17000 print"■■■■siete sicuri di voler sa
lvare":printtab(15)"il vostro gioco?"
17010 print"■premete ■s■ o ■n■:"
17020 geta$:ifa$<>"s"anda$<>"n"then17020
17030 ifa$="n"thenprint"■":goto4020
17040 input"■metti il nastro e premi ■RE
TURN:"q$:r$=chr$(13)
17050 open1,1,1,"planet"
17060 print#1,pr;r$;cv;r$;sv;r$;ev;r$;es
;r$;qv;r$;dv;r$;nv;r$;fv
17070 print#1,z;r$;n$;r$;m;r$;i;r$;p;r$;
n;r$;e;r$;s;r$;w;r$;i$;r$;pi;r$;sp
17080 print#1,ve$;r$;no$;r$;q;r$;p2;r$;n
p;r$;na;r$;rc;r$;yy;r$;hh;r$;hr
17090 fori=1toq:print#1,ob%(i):nexti
17100 fori=1toq:print#1,ob$(i):nexti
17110 fori=1toq:print#1,si$(i):nexti
17120 close1
17130 gosub35420
17140 print"■":goto4020
17470 rem**
17480 rem esce
17490 rem**
17500 pokebs,2:pokebc,2
17510 print"■":printtab(5)"■■■■■■■siete
```

```
sicuri ";n$;" ??":gosub19000
17520 printtab(5)"■■Batti ■c■ per ■c■ont
inuare:"
17530 printtab(5)"■■Batti ■u■ per ■u■esci
re dal gioco:"
17540 geta$:ifa$<>"c"anda$<>"u"then17540
17550 ifa$="u"then17570
17560 pokebs,11:pokebc,11:print"■":goto4
020
17570 print"■":print"■":poke53272,21:pok
ebs,14:pokebc,6:end
17970 rem**
17980 rem beep
17990 rem**
18000 forl=0to24:pokesc+l,0:nextl
18010 pokevo,15:pokea1,17:pokea2,17:poke
s1,64:pokes2,64
18020 pokew1,17:pokew2,17:pokeh1,17:poke
l1,195:pokel2,6
18030 fort=1to100:nextt:pokevo,0
18040 return
18970 rem**
18980 rem burp
18990 rem**
19000 forl=0to24:pokesc+l,0:nextl
19010 pokevo,15:pokea1,17:pokea2,64:poke
a3,17:pokes1,128:pokes2,128:pokes3,128
19020 pokew1,33:pokew2,33:pokew3,129
19030 pokeh1,189:pokel1,172:pokeh2,7:pok
el3,233:pokeh3,1:pokel3,19
19040 fort=1to200:nextt:pokevo,0
19050 return
19970 rem**
19980 rem luoghi
19990 rem**
20000 rem 1
20010 gosub 22050:n=0:e=2:s=11:w=0:goto4
150
20020 rem 2
20030 print"■■■Mentre avanzate ancora di
 Piu´ dei pezzetti di carta vi ";
20040 print"attraggono. In una fessura d
el muro c´e´ un messaggio con la ";
```

```
20050 print"9rafia di aurora. Nell'incer
ta luce potete ";
20060 print"riconoscere proprio le parol
e man9ia noci."
20070 n=0:e=3:s=12:w=1:9oto4150
20080 rem 3
20090 print"■■Avanzando sempre piu'profo
ndamente nel cuore del";
20100 print"tempio,diventa buio e umido.
  Le pareti sono coperte con ";
20110 print"una fetida melma mali9na che
 sembra pulsare con luce ";
20120 print"malefica..":n=0:e=4:s=13:w=2
:9oto4150
20130 rem 4
20140 print"■■E' proprio buio dentro il
 tempio.Il terreno e' scomesso ";
20150 print"e spesso cadete. Appare un p
assa99io ";
20160 print"verso ovest .":n=0:e=0:s=0:w
=3:9oto4150
20170 rem 11
20180 9osub 22050:n=1:e=12:s=0:w=0:9oto4
150
20190 rem 12
20200 9osub 22050:n=2:e=13:s=22:w=11:9ot
o4150
20210 rem 13
20220 print"■■Dentro al tempio e' troppo
buio per vedere bene. Passa99i ";
20230 print"sembrano mandare in parecchi
e direzioni. Qual'e' la mi9liore?"
20240 n=3:e=14:s=0:w=12:9oto4150
20250 rem 14
20260 print"■ Siete in un edificio che p
are un tempio. ";
20270 print"Sul pavimento fuori la stanz
a ci sono se9ni ";
20280 print"tracciati sul fan9o del corr
idoio. Sembra come una ";
20290 print"fraccia puntata ad ovest den
tro il tempio."
20300 n=0:e=15:s=24:w=13:9oto4150
```

```
20310 rem 15
20320 print"■■La ricerca delle tracce v
i rende un turbine - Pronti a ";
20330 print" tutto. Ma solo le Pietre re
stano insensibili - ovviamente ";
20340 print"alla vostra Paura. Forse non
 sono morte, dormono solo..."
20350 n=0:e=0:s=25:w=14:goto4150
20360 rem 22
20370 gosub 22050:n=12:e=0:s=0:w=0:goto4
150
20380 rem 24
20390 print"■■Avete l'imPressione che q
ualcuno vi osservi nascosto nelle ";
20400 print"ombre. Nel silenzio volete u
rlare, chiamare aiuto, fare ";
20410 print"qualsiasi cosa Per saPere se
 c'e' qualcuno di vivo qui in ";
20420 print"questo Posto morto.":n=14:e=
25:s=0:w=0:goto4150
20430 rem 25
20440 print"■■Il vento fischia deridendo
vi, sPettrale suono tra le Pareti ";
20450 print"roccIose, mentre Passate Per
 stanze vuote, sPalancando finestre. ";
20460 print"Solo questo rimane della Pas
sata civiltà?"
20470 n=15:e=26:s=0:w=24:goto4150
20480 rem 26
20490 print"■■Vorreste vedere dove siet
e, nella citta'! Se solo ";
20500 print"Poteste trovare una via; ma
i muri sbrecciati dell'oscura casa ";
20510 print"bloccano ogni via eccetto un
 intrico di cunicoli verso ovest."
20520 n=0:e=0:s=36:w=25:goto4150
20530 rem 28
20540 print"■■Siete sulla riva di un lag
o.  Le Pietre nell'acqua sembrano ";
20550 print"riflettere la luce del sole
come se avessero ";
20560 print"un Potere magico.":n=0:e=0:s
=38:w=0:goto4150
```

```
20570 rem 36
20580 print"■■da qui potete chiaramente
 vedere che le torri e le ";
20590 print"guglie sono ora le sbrecciat
e rovine di una fiera citta' ";
20600 print" abbandonata dagli abitanti.
"
20610 n=26:e=37:s=46:w=0:goto4150
20620 rem 37
20630 print"■■Infine la sabbia prende il
 posto dei ciottoli e sassi -";
20640 print"■dovrebbe essere il margine
del deserto. Ci sono ";
20650 print"alcuni alberi, simili a casp
ugli con frutti come noci di cocco ";
20660 print"pendono da essi.":n=0:e=38:s
=47:w=36:goto4150
20670 rem 38
20680 print"■■Siete fermi nel deserto.
A nord potete vedere un lago. ";
20690 print"Ad est il deserto tocca l'or
izzonte."
20700 n=28:e=0:s=38:w=37:goto4150
20710 rem 42
20720 gosub 22110:n=0:e=43:s=52:w=0:goto
4150
20730 rem 43
20740 gosub 22070:n=0:e=44:s=53:w=42:got
o4150
20750 rem 44
20760 gosub22110:n=0:e=0:s=54:w=43:goto4
150
20770 rem 46
20780 print"■■Anche se il sole sta comin
ciando a crearvi ";
20790 print"confusione nella mente - all
'orizzonte nord appaiono ";
20800 print"guglie e torri di una citta'
. O e' solo un miraggio ";
20810 print"che vi tortura ??":n=36:e=37
:s=56:w=0:goto4150
20820 rem 47
20830 print"■Mentre camminate rapidi sul
```

```
la sabbia rovente Potete vedere ";
20840 Print" i segni di dove la vostra a
stronave e' caduta atterrando. ";
20850 Print"Vi chiedete come farla riPar
tire."
20860 n=37:e=46:s=57:w=46:goto4150
20870 rem 48
20880 Print"■■Avete camminato Per ore e
Potete vedere";
20890 Print" che la vasta distesa di sab
bia non ha fine. Infatti saPete che ";
20900 Print"se non tornate Presto indiet
ro morirete a causa  ";
20910 Print"dell'ardore del sole che sem
bra fisso alto nel cielo."
20920 n=38:e=0:s=0:w=47:goto4150
20930 rem 52
20940 gosub 22070:n=42:e=53:s=62:w=0:got
o4150
20950 rem 53
20960 Print"■■Siete in una Piccola radur
a tra gli alberi. Un soffice muschio ";
20970 Print"ricoPre le rive di uno stagn
o azzurro e silenzioso, dalle ";
20980 Print"acque immote; rimanete Per u
n Po' Prima di ricominciare la ";
20990 Print"ricerca.":n=43:e=54:s=63:w=5
2:goto4150
21000 rem 54
21010 Print"■■Mentre lottate con rovi e
fogliame mandate occhiate ";
21020 Print"sPeranzose alla radura verso
 ovest, ma vi sembra di ";
21030 Print"non essere caPaci di raggiun
gerla !!":n=44:e=55:s=64:w=0:goto4150
21040 rem 55
21050 Print"■■■Siete proPrio nella fores
ta ora, con alberi grossi come case ";
21060 Print"slanciati al cielo, che impe
discono la vista del sole. ";
21070 Print"Animaletti balzano fuori e f
uggono davanti a voi."
21080 Print"Due sentieri si inoltrano -
```

```
uno a sud l'altro ad ovest."
21090 n=0:e=56:s=65:w=54:goto4150
21100 rem 56
21110 print"■■■Siete al limite di un des
erto - la sabbia cambia subito ";
21120 print"in erba di boscaglia.  Verso
 ovest ";
21130 print"un'oscura foresta riempie il
 cielo come uno spesso tappeto ";
21140 print" verde.":n=46:e=57:s=0:w=55:
goto4150
21150 rem 57
21160 print"■■■Ora siete all'esterno de
ll'astronave. Attorno ci sono tracce ";
21170 print"di bruciature ed i frammenti
 carbonizzati dell'astronave. Siete ";
21180 print"caduti in un arido deserto d
i sabbia rossastra."
21190 n=47:e=58:s=67:w=56:goto4150
21200 rem 58
21205 ifob%(2)=-1andob%(9)=-1then33280
21210 ifob%(2)<>-1then21240
21220 ifob%(2)=-1thenprint"■Magnifico !
Avete il cristallo.":cv=2
21230 print"■Mettetelo a riposare nella
teca nella cabina principale.":goto21380
21240 ifpr=2andob%(9)=-1then21330
21250 ifhh=2then21270
21260 ifpr=2andob%(9)<>-1then21350
21270 print"■■■Siete nella cabina."
21280 ifpr=2then21380
21290 print"■■Una delle tute spaziali ma
nca.   ";
21300 print"Sul pavimento vedete un bran
dello di vestito che ";
21310 print"riconoscete come un pezzetto
  della gonna che la principessa ";
21320 print"aurora   indossava l'ultima
volta che l'avevate vista...":goto21380
21330 print"■Indietro con aurora - Potre
i portarla nella cabina principale ";
21340 print"e offrirle una tazza di te'!
!":goto21380
```

```
21350 gosub19000:print"■■Dimenticate di ■
prendere aurora nel villaggio ";
21360 print"indigeno ";n$
21370 print"Un po' stupido - tutto cio'!
! Dovete tornare per lei."
21380 n=0:e=59:s=0:w=57:goto4150
21390 rem 59
21400 ifcv=2andpr=2then33280
21410 ifcv=2andpr<>2then33400
21420 ifcv=0andpr=2then33460
21430 print"■■■Siete nella cabina dell'a
stronave; la  caduta ha sparso ";
21440 print"il contenuto del vostro baul
e tutto sul pavimento."
21450 ifpr=2then21500
21460 print"Nonostante cio' scorgete i s
egni di una lotta."
21470 print"■■■***La Principessa e' spar
ita!***"
21480 print"■■■Il pannello di controllo
registra che  la cabina e' stata ";
21490 print"aperta dall'esterno e poi ri
chiusa..."
21500 n=0:e=0:s=0:w=58:goto4150
21510 rem 62
21520 gosub 22070:n=52:e=63:s=0:w=0:goto
4150
21530 rem 63
21540 gosub 22110:n=0:e=64:s=0:w=62:goto
4150
21550 rem 64
21560 print"■■La foresta e' assolutament
e troppo grande per trovare ";
21570 print"la via per uscire; state com
inciando a preoccuparvi ";
21580 print"chiedendovi cosa mai potrebb
e succedere."
21590 n=54:e=65:s=74:w=63:goto4150
21600 rem 65
21610 print"■■■Questo sentiero sembra co
ndurre via dal sole, dentro ";
21620 print"un'oscurita' che non pensava
te possibile nella luce ";
```

```
21630 Print"del sole. Dietro di voi ci s
ono dei rumori, come se qualche ";
21640 Print"animale colossale vi stesse
seguendo.":n=55:e=0:s=0:w=64:goto4150
21650 rem 67
21660 Print"■■Il deserto diventa all'imP
rovviso Pieno di vegetazione";
21670 Print" che sembra infoltirsi ad og
ni Passo. Guardando meglio intorno ";
21680 Print" - il terreno sembra farsi P
aludoso. ";
21690 n=57:e=68:s=0:w=0:goto4150
21700 rem 68
21710 Print"■■Siete in una Palude, cald
a, vischiosa e fastidiosa. ";
21720 Print"Potete sentire il rumore di
un fiume,immenso e veloce.";
21730 Print" Ad est c'e' il fiume - ad o
vest la Palude."
21740 n=0:e=69:s=0:w=67:goto4150
21750 rem 69
21760 Print"■■E' cosi' rinfrescante sed
ere sulla riva del fiume e ";
21770 Print" dondolare i Piedi bollenti
nella fresca corrente. Ma il fiume ";
21780 Print" e' largo, come un mare agit
ato all'orizzonte. Sarebbe ";
21790 Print"imPossibile da attraversare.
":n=0:e=0:s=0:w=68:goto4150
21800 rem 69
21810 Print"■■Siete sulla strada verso
il sud della foresta. Il manto ";
21820 Print" liscio suggerisce che sia s
tata fatta da un'antica civilta' ";
21830 Print"suPeriore a qualsiasi altra
nota. Ma la strada e' ancora usata...";
21840 Print" o e' solo un relitto di glo
ria Passata ?"
21850 n=64:e=0:s=84:w=0:goto4150
21860 rem 84
21870 Print"■■La strada si snoda in una
verde Pianura - indugiando a nord ";
21880 Print"e a sud come una corda getta
```

```
ta da una mano gigantesca. ";
21890 print"Dall'altra parte la vegetazi
one si avvicina viepPiu'...";
21900 n=74:e=0:s=94:w=0:goto4150
21910 rem 92
21920 print"■■Siete in una casa infangata. Pezzi di terracotta preistorica ";
21930 print" crepitano sotto i vostri Pi
edi.":n=0:e=93:s=0:w=0:goto4150
21940 rem 93
21950 print"■■Prendete la via per il vi
llaggio sperando che gli indigeni ";
21960 print" siano amichevoli.":n=0:e=94
:s=0:w=92:goto4150
21970 rem 94
21980 print"■■Siete fuori la cinta del
villaggio. Case sudicie ";
21990 print"sono sparse intorno alla rin
fusa."
22000 ifpr<>2then22020
22010 goto22040
22020 print"Il villaggio pare deserto, m
a sentite una donna che ";
22030 print"canta mestamente in una capa
nna."
22040 n=84:e=0:s=0:w=93:goto4150
22050 print"■■Siete in una specie di ded
alo di curve e incroci";
22060 print" temendo non poco' di esserv
i perduti.....":return
22070 print"■■Potete esservi perduti nel
la foresta. Date un'occhiata ";
22080 print"alla radura, un po' ad est,
un po' a sud...";
22090 print"e sempre il rumore di una qu
alche bestia terribile echeggia e ";
22100 print"vibra tra le foglie intorno
a voi.":return
22110 print"■Che foresta - questa ! Semb
rate definitivamente perduti."
22120 print"Forse avreste dovuto rimaner
e a casa, senza offrire ";
22130 print"ad Aurora passaggio sulla vo
```

```
stra nave !":return
29970 rem**
29980 rem serpente
29990 rem**
30000 print"■Fuori le rovine striscia un
 gigantesco serpente - il corpo ";
30010 print"grosso il doppio del vostro.
 Vi blocca L'unica uscita.":goto5000
30470 rem**
30480 rem anguilla
30490 rem**
30500 print"■E' molto profonda nell'acqu
a.":return
30510 ifob%(6)<>-1thengoto30540
30515 ifve=2thengosub18000:print"■Bel gi
orno per un tuffo! Cosa ora?":goto4320
30520 ifob%(4)<>-1thengoto30560
30530 goto30590
30540 gosub19000:print"■Non avete piu' l
'astronave, vero";n$
30550 print"Maledizione!! Siete annegato
!!":form=1to2000:nextm:goto190
30560 gosub18000:print"■Galleggiate come
 un sughero. Vi occorre qualcosa ";
30570 print"di pesante per tuffarvi   ";
30580 print"giu' per prendere il cristal
lo.":goto5000
30590 ifob%(2)=-1thengosub18000:print"■B
ella giornata per un tuffo!!!":goto4320
30610 es=1:print"■All'inizio scendete fi
no a prendere il cristallo. ";
30620 print"All'improvviso esce dal prof
ondo una gigantesca anguilla ";
30630 print"e vi si attorciglia intorno.
"
30640 print"Vi sta stritolando...":gosub
19000:rc=2:q=0:yy=1:goto5000
30650 gosub18000:print"■■L'animale vi h
a stretto cosi' forte che ";
30660 print"non potete fuggire !":goto50
00
30970 rem**
31000 print"■All'improvviso vi sentite s
```

```
offocare. L´atmosfera ";
31010 print"e´ velenosa. Se non farete n
ulla morrete presto..."
31030 form=1to5000:nextm
31040 print"■Cominciate a preoccuparvi,
";n$:print"*";
31050 forl=0to24:pokesc+l,0:nextl
31060 fori=1to30:print"*";
31070 pokevo,15:pokea1,41:pokes1,89:poke
w1,17:pokeh1,56:pokel1,99
31080 fort=1to200:nextt:pokevo,0:pokea1,
0:pokew1,0:pokes1,0
31090 nexti
31100 printchr$(13):print"■■troppo tardi
!!":print"troppo tardi!!":goto190
31470 rem**
31480 rem si toglie la tuta
31490 rem**
31500 ifp=58then12080
31510 ifp=59then12080
31520 iffv<>2then31000
31530 iffv=2then12080
31570 rem**
31580 rem se mangia le noci
31590 rem**
31600 ifno$="noci"orleft$(no$,5)="frutt"
then31620
31610 gosub19000:print"■Uuh! Perche´ vol
ete mangiare ■■■■■?":goto5000
31620 ifp=37thenprint"■Okay.Mm! buon sap
ore!":goto31650
31630 ifob%(3)=-1thenprint"■Okay.Mm! buo
n sapore!":goto31650
31640 ifob%(3)<>-1thengosub19000:print"■
Non avete nessun ";no$:goto5000
31650 ifz=1then31700
31660 gosub35100:print"■":print"■Fate un
a profonda inspirazione ";n$
31670 print"■■Potreste provare una stran
a sensazione percorrervi il corpo ";
31680 print" e potete respirare (felicem
ente) senza la vostra ";
31690 print"tuta. Potete toglierla.":ob%
```

```
(3)=0:z=1:fv=2:g=g-1:goto5000
31700 gosub18000:Print"■■Questa volta no
n sembra diverso. ";
31710 Print"Blocca ancora il vostro sens
o di fame.":ob%(3)=0:goto5000
31970 rem**
31980 rem cristallo
31990 rem**
32000 Print"■■Notate che il cristallo e'
 sparito dalla teca energetica. ";
32010 Print"Senza di esso la    nave non
 puo' volare!! Deve averlo ";
32020 Print"preso qualcuno! Dovete trova
rlo prima che    scappi."
32030 return
32040 rem ***
32050 rem non puo' prendere il cristallo
32060 rem ***
32070 ifno$="cristallo"then32090
32080 goto11700
32090 ifev<>2thengosub19000:Print"■Non p
otete...";:form=1to500:nextm
32100 ifev<>2thenPrint"...ancora!!":prin
t"E' ■sotto■ acqua.":goto4320
32110 ifev=2then11700
32500 gosub18000:Print"■A nord c'e' una
citta' in rovina - a sud molti ";
32510 Print"alberi arrivano all'orizzont
e. Discendete.":goto4320
32520 gosub18000:Print"■A sud potete ved
ere la vostra navicella - ovunque ";
32530 Print"alberi ed alberi ... Discend
ete.":goto4320
32540 gosub18000:Print"■A sud potete ved
ere del fumo salire al cielo. Puo' ";
32550 Print"esserci vita o cose simili.
Discendete.":goto4320
32570 rem**
32580 rem sabbie mobili
32590 rem**
32600 Print"■■■State sprofondando nelle
sabbie mobili. in breve vi sono alle ";
32610 Print"ginocchia.Bisogna far qualco
```

```
sa!!":q=0:goto5000
32670 rem**
32680 rem dinosauro
32690 rem**
32700 print"■Un colossale dinosauro esce
 dagli alberi e vi taglia la strada. ";
32710 print"Non potete passare.":goto500
0
32970 rem**
32980 rem indigeni
32990 rem**
33000 print"■una folla di indigeni ostil
i appare all'ingresso della ";
33010 print"capanna - bloccando la via.
Avanzano puntandovi le lance alla ";
33020 print"gola. Vogliono occuparsi di
";n$:q=0:goto5000
33030 rem**
33040 rem indietro alla capanna
33050 rem**
33060 print"■Indietro ancora!! Spero che
 abbiate pensato a come dare ad ";
33070 print"aurora il bacio che vi aiute
ra' a portarla via di qui!!"
33080 na=0:q=0:goto5000
33100 rem liberazione illegale di aurora
33120 ifnv<>2andno$="aurora"then33170
33130 ifnv<>2andno$="principessa"then331
70
33140 ifnv<>2andno$="principessa"then331
70
33150 goto11700
33170 gosub35100:print"■"
33180 print"■Spiacente ";n$:print"ma gli
 indigeni non amano i piloti !!";
33190 print"e e vi hanno impedito di ten
tare di liberarla."
33200 print"Vi hanno ucciso ! A questo v
i ha portato la via!!":goto190
33210 rem**
33220 rem niente aurora
33230 rem**
33240 print"■Perduta aurora non potete
```

```
completare la missione.":return
33250 rem**
33260 rem trovata aurora
33270 rem**
33280 gosub35100:print"■"
33290 print"■■■Congratulazioni!! Avete
compiuto la missione!!"
33300 print"■■Dopo solo poche ore potete
 decollare con l'astronave ";
33310 print"e tuffarvi nello spazio prof
ondo."
33320 print"■■E la principessa Aurora ha
 alla fine deciso di non sposare ";
33330 print"il crudele tiranno di Zen !!
":print" Preferisce voi ";n$
33340 print"■■Ce l'avete fatta eh ?":pok
ev+21,0:end
33350 gosub18000:print"■Aurora vi bacia
per la fortuna che avete avuto nel ";
33360 print"trovare il cristallo magico.
":hh=2:goto12090
33370 rem**
33380 rem trovato cristallo
33390 rem**
33400 gosub35100:print"■":print"■Okay -
cosi' avete il cristallo energetico.";
33410 print"Ora tutto cio' che dovete fa
re e' trovare la principessa!"
33420 print"Non dimenticate di lasciare
qui il cristallo.":goto21500
33430 rem**
33440 rem niente cristallo
33450 rem**
33460 gosub35100:print"■":print"■Okay. a
urora e' salva e voi potete ";
33470 print"lasciarla qui se volete.":go
to21500
35000 pokev+39,2:pokev+27,1:pokev+23,1:p
okev+29,1
35010 pokev+1,100:pokev,0:pokev+21,1
35020 forx=0to255:pokev,x:nextx:pokev+16
,1
35030 forx=0to95:pokev,x:nextx:pokev+16,
```

```
0:Pokev+21,0
35040 return
35100 forl=0to24:Pokesc+l,0:nextl
35110 Pokevo,15:Pokea1,190:Pokea2,190:Po
kea3,190:Pokew1,33:Pokew2,33:Pokew3,129
35120 Pokeh1,75:Pokel1,69:Pokeh2,212:Pok
el2,230:Pokeh3,1:Pokel3,123
35130 form=1to200stepP.25:nextm
35140 Pokev+40,4:Pokev+29,2:Pokev+23,2
35150 Pokev+2,80:Pokev+3,0:Pokev+21,2
35160 fory=0to190:Pokev+3,y:nexty
35170 Pokew1,0:Pokea1,0:Pokew2,0:Pokea2,
0:Pokew3,0:Pokea3,0
35180 return
35200 data 99999
35210 rem**
35220 rem data sprite 1
35230 rem**
35240 data 255,128,0,255,128,0,1,128,0,1
,128
35250 data 0,3,252,0,31,255,224,255,252,
32,63
35260 data 252,63,63,255,255,224,0,1,63,
255,240
35270 data 63,255,240,3,1,0,3,25,0,3,31
35280 data 0,227,0,0,255,0,0,0,0,0,0
35290 data 0,0,0,0,0,0,0,0,0
35330 data 0,112,0,0,248,0,1,252,0,1,172
35340 data 0,1,172,0,3,254,0,3,222,20,1
35350 data 116,20,1,4,28,1,252,8,0,112,8
35360 data 7,255,8,7,255,8,127,255,248,6
8,1
35370 data 0,229,253,0,255,253,0,4,1,0,7
35380 data 255,0,7,255,0,1,36,0
35420 restore:forll=1to1000:reada$:ifa$=
"99999"thenll=1016
35430 nextll
35440 Poke2041,14:Poke2040,13
35450 forll=0to126:read9:Poke832+ll,9:ne
xtll
35460 restore
35470 return
```

SUNSHINE

Adventure e COMMODORE 64 è un manuale per ideare e utilizzare programmi di Adventure basate esclusivamente sul testo. Il processo di adattamento di una trama concepita nella mente agli schemi della programmazione, viene semplificato dall'uso di diagrammi, mappe e cartine.
Una progettazione modulare del programma rende più facile la comprensione della struttura. Tutti i moduli sono spiegati in dettaglio, linea per linea, consentendo di assimilare i concetti di base utilizzati. L'abilità nella programmazione avrà modo di emergere durante la costruzione graduale dell'Adventure, per modificare il programma dimostrativo, o addirittura per costruirne uno nuovo.
Il libro è diviso in due parti:

1) una cornice semplificata per Adventure adatta per qualunque trama, che consente di capire come costruirne una da soli;
2) un'Adventure completa, "Il pianeta dell'Incubo", che introduce ai problemi e ai trabocchetti della programmazione di Adventure, con note ed anedotti per superarli.

L. 35.000

Cod. 571D ISBN 88-7056-172-O