

Dopo un gioco "meditativo" come il Mastermind che vi abbiamo presentato il mese passato è ora la volta di un gioco d'azione e di simulazione. Riuscirà il nostro eroico comandante di un'astronave ad effettuare un atterraggio su un pianeta sconosciuto senza schiantarsi sulle scoscese montagne e senza lasciarsi vincere dalla terribile forza di gravità?

Cimentandovi in questo tentativo meditate su quanto siete fortunati: a differenza di un reale ed autentico comandante che rischia la sua vita e quella dell'equipaggio voi non perderete comunque la vostra "pellaccia"; male che vada potrete sempre premere il tasto A ed avrete un'astronave ed un equipaggio nuovi di zecca.

Atterraggio

di Paolo Di Francescantonio — Brescia

Questo programma trae origine da un noto videogame molto di moda anni fa nelle sale da gioco; lo scopo è quello di riuscire ad effettuare un perfetto atterraggio con una astronave. Appena eseguito il RUN, sul video appariranno, sul fondo di un cielo stellato (la posizione delle stelle è casuale e varia di volta in volta), alcune montagne con qualche zona pianeggiante su cui è possibile atterrare. Le piste di atterraggio sono sempre cinque e si riconoscono facilmente perché sono piatte. Appare inoltre, nella parte superiore dello schermo, un'astronave alla quale è stata assegnata una velocità iniziale casuale. La velocità verticale è visualizzata in alto a sinistra dello schermo mentre a destra viene mostrata la quantità di carburante an-

cora disponibile (inizialmente 400 litri).

Lo scopo del gioco è quello di riuscire ad atterrare su una di queste zone pianeggianti con velocità verticale minore o uguale a 15 m/s consumando la minor quantità possibile di carburante. Al termine della partita, se l'atterraggio è riuscito, viene assegnato un punteggio dipendente dalla velocità di atterraggio, dal carburante rimasto e dalla difficoltà della pista su cui si è atterrati (ognuna delle cinque piste ha un diverso coefficiente di difficoltà).

Il programma ha memorizzati al suo interno cinque diversi sfondi per l'atterraggio, alcuni facili, altri più difficili, che vengono scelti casualmente; ognuno di questi sfondi prevede cinque piste di atterraggio per un totale di venticinque possibilità.

Se non viene impartito nessun comando l'astronave è soggetta ad una accelerazione di 10 m/s. Ad ogni spostamento dell'astronave si sente un rumore acuto. L'uso dei

```

10 REM *** ATTERAGGIO *****
20 REM *** DI PAOLO FRANCESCANTONIO ***
30 CALL CLEAR
40 CAR=400
50 L=1
60 P=1
70 DIM A(35)
80 FOR I=1 TO 8
90 CALL COLOR(1,16,1)
100 NEXT I
110 CALL SCREEN(2)
120 CALL COLOR(16,7,1)
130 CALL COLOR(13,10,1)
140 CALL COLOR(14,16,1)
150 CALL COLOR(12,2,1)
160 CALL COLOR(15,12,1)
170 CALL CHAR(152,"FFFFFFFFFFFFFF")
180 CALL CHAR(144,"0000000507023F")
190 CALL CHAR(140,"0000001")
200 CALL CHAR(145,"0000000F0F0207E")
210 CALL CHAR(146,"7F27222272060001")
220 CALL CHAR(147,"FFF2A2A2A7B0B0C0")
230 CALL CHAR(128,"80C0E0E0F0C0FEFF")
240 CALL CHAR(129,"01030F0F1F7FFF")
250 CALL CHAR(130,"80C0C0E0E0F0F0B")
260 CALL CHAR(131,"8F0CFCFCFCFEFF")
270 CALL CHAR(132,"010307070F0F0F")
280 CALL CHAR(133,"1F3F7F7F7FFF")
290 CALL CHAR(134,"00000000000000FF")
300 CALL CHAR(135,"FFFFFFFFFFFFFF")
310 DIM V(16)
320 RANDOMIZE
330 TER=INT(5*RAND)+1
340 ON TER GOTO 2440,2690,2940,3190,3440
350 FOR I=1 TO 32
360 READ A(I),C
370 R=(75-A(I))*3
380 CALL SOUND(R,A(1)*20,2)
390 GOSUB 1530
400 NEXT I
410 CALL HCHAR(1,1,86,1)
420 CALL HCHAR(1,2,69,1)
430 CALL HCHAR(1,3,76,1)
440 CALL HCHAR(1,4,61,1)
450 CALL HCHAR(1,26,67,1)
460 CALL HCHAR(1,27,65,1)
470 CALL HCHAR(1,28,82,1)
480 CALL HCHAR(1,29,61,1)
490 X=16
500 Y=1
510 W=0
520 D=X
530 D=Y
540 T=0
550 CALL SCREEN(2)
560 VEL=(INT(30*RAND)+10)*5
570 S=INT(3*RAND)-1
580 FORZA=0
590 CALL SOUND(100,500,3)
600 CALL HCHAR(D,0,127,2)
610 CALL HCHAR(D+1,0,127,2)
620 CALL HCHAR(Y,X,144,1)
630 CALL HCHAR(Y,X+1,145,1)
640 CALL HCHAR(Y+1,X,146,1)
650 CALL HCHAR(Y+1,X+1,147,1)
660 D=X
670 D=Y
680 CALL HCHAR(1,30,32,3)
690 CALL HCHAR(1,3,32,3)
700 VEL=STR$(VEL)
710 LEN=LEN(VEL)
720 FOR I=1 TO LEN
730 AS=ASC(SEG$(VEL$,IE,1))
740 CALL HCHAR(1,IE+4,AS,1)
750 NEXT IE
760 CAR=STR$(CAR)
770 LEN=LEN(CAR)
780 FOR I=1 TO LEN
790 AS=ASC(SEG$(CAR$,IE,1))
800 CALL HCHAR(1,IE+29,AS,1)
810 NEXT IE
820 ON T+1 GOTO 830,1380
830 CALL KEY(3,KEY,STATUS)
840 IF KEY=69 THEN 1030
850 IF KEY=88 THEN 1080
860 IF KEY=85 THEN 1130
870 IF KEY=68 THEN 1180
880 VEL=VEL+10-FORZA
890 IF VEL=0 THEN 1230
900 IF VEL<0 THEN 1250
910 V=Y+1
920 U=10000/VEL
930 S=S+S
940 FOR K=1 TO U
950 NEXT K
960 IF Y<1 THEN 1290
970 IF X<31 THEN 3680
980 IF X<1 THEN 3700
990 IF (Y+1)=A(X) THEN 1360
1000 U=X+1
1010 IF (Y+1)=A(U) THEN 1360
1020 GOTO 580
1030 IF CAR(40) THEN 880
1040 CAR=CAR+0
1050 FORZA=40
1060 CALL SOUND(200,200,3,-8,3)
1070 GOTO 880
1080 IF CAR(5) THEN 880
1090 CAR=CAR-5
1100 FORZA=5
1110 CALL SOUND(200,200,2,-8,2)
1120 GOTO 880
1130 IF CAR(10) THEN 880
1140 CAR=CAR-10
1150 S=S-1
1160 CALL SOUND(200,200,1,-8,1)
1170 GOTO 880
1180 IF CAR(10) THEN 880
1190 CAR=CAR-10
1200 S=S-1
1210 CALL SOUND(200,200,1,-8,1)
1220 GOTO 880
1230 U=50
1240 GOTO 920
1250 Y=Y-1
1260 U=(10000/VEL)
1270 GOTO 920
1280 CALL SOUND(500,200,3)
1290 RESTORE 3860
1300 FOR I=1 TO 32
1310 READ X
1320 CALL HCHAR(24,I,X,1)
1330 NEXT I
1340 CALL SOUND(500,200,3)
1350 GOTO 1910
1360 T=1
1370 GOTO 580
1380 ON -(X*VAR(11)+(X*VAR(21))+1) GOTO 1390,1390,1430
1390 ON -(X*VAR(21)+(X*VAR(41))+1) GOTO 1400,1400,1450
1400 ON -(X*VAR(51)+(X*VAR(61))+1) GOTO 1410,1410,1470
1410 ON -(X*VAR(71)+(X*VAR(81))+1) GOTO 1420,1420,1490
1420 ON -(X*VAR(91)+(X*VAR(101))+1) GOTO 2040,2040,1510
1430 V=V*(11)
1440 V=V*(14)
1450 V=V*(12)
1460 SOTC 1710
1470 V=V*(13)
1480 SOTC 1710
1490 V=V*(14)
1500 GOTO 1710
1510 V=V*(15)
1520 GOTO 1710
1530 CALL HCHAR(A(1),1,C,1)
1540 T=0
1550 IF C=130 THEN 1660
1560 IF C=133 THEN 1690
1570 IF A(1)=24 THEN 1650
1580 V=A(1)+1+2
1590 IF Y=24 THEN 1650
1600 CALL HCHAR(Y,1,135,25-Y)
1610 COL=INT(32*RAND)+1
1620 RIG=INT(13*RAND)+1
1630 NU=INT(2*RAND)+1
1640 CALL HCHAR(RIG,COL,140,NU)
1650 RETURN
1660 CALL HCHAR(A(1)+1,1,131,1)
1670 Z=1
1680 GOTO 1580
1690 CALL HCHAR(A(1)-1,1,132,1)
1700 GOTO 1580
1710 IF VEL>15 THEN 2050
1720 S=INT((15-VEL)*5+CAR/10)+V
1730 GOSUB 3720
1740 RESTORE 3780
1750 CALL HCHAR(24,1,32,32)
1760 FOR I=1 TO 15
1770 READ X
1780 CALL HCHAR(24,1,X,1)
1790 NEXT I
1800 IF D<MAX THEN 3800
1810 PUM=STR$(P)
1820 LEN=LEN(PUM)
1830 FOR I=1 TO LEN
1840 AS=ASC(SEG$(PUM$,I,1))
1850 CALL HCHAR(24,1+18,AS,1)
1860 NEXT I
1870 FOR I=1 TO 5
1880 READ X
1890 CALL HCHAR(24,1+23,X,1)
1900 NEXT I
1910 FOR I=1 TO 4
1920 READ X
1930 CALL HCHAR(1,1+12,X,1)
1940 NEXT I
1950 MAX=STR$(MAX)
1960 LEN=LEN(MAX)
1970 FOR I=1 TO LEN
1980 AS=ASC(SEG$(MAX$,I,1))
1990 CALL HCHAR(1,1+16,AS,1)
2000 NEXT I
2010 CALL KEY(3,KEY,STATUS)
2020 IF STATUS=0 THEN 2010
2030 IF KEY=65 THEN 30 ELSE 2010
2040 W=1
2050 D=Y
2060 F=X+1
2070 CALL COLOR(15,11,7)
2080 CALL SOUND(1500,110,3,-7,3)
2090 FOR G=1 TO 400
2100 NEXT G
2110 CALL HCHAR(Y+1,X,127,2)
2120 CALL SOUND(3500,-7,0)
2130 FOR G=1 TO 10
2140 CALL HCHAR(Y,X,127,1)
2150 CALL SCREEN(12)
2160 CALL HCHAR(D,F,127,1)
2170 X=X-1
2180 Y=Y+1
2190 F=F+P
2200 D=D-1
2210 ON -(X(31)+X(32))+1 GOTO 2370,2370,2220
2220 ON -(F(1)+F(32))+1 GOTO 2400,2400,2230
2230 CALL SCREEN(2)
2240 CALL HCHAR(Y,X,144,1)
2250 CALL HCHAR(D,F,145,1)
2260 NEXT G
2270 IF W=1 THEN 2340
2280 RESTORE 3820
2290 CALL HCHAR(24,1,32,32)
2300 FOR I=1 TO 32
2310 READ X
2320 CALL HCHAR(24,1,X,1)
2330 NEXT I
2340 CALL SOUND(500,200,3)

```

tasti è il seguente: E per decelerare di 40 m/s con un consumo di carburante di 40 litri, X per decelerare di 5 m/s con un consumo di carburante di 5 litri, S per accelerare a sinistra di 10 m/s con relativo consumo di 10 litri, D per accelerare a destra di 10 m/s sempre con un consumo di 10 litri di carburante. Da notare che i comandi non vengono ovviamente eseguiti se il carburante disponibile è minore di quello richiesto e che alla decelerazione ottenibile con i tasti E e X va aggiunta l'accelerazione di gravità pari a 10 m/s ottenendo, in definitiva, una decelerazione effettiva pari a 30 e -5 m/s rispettivamente.

Bisogna inoltre fare attenzione al fatto che una volta che l'astronave ha assunto una certa velocità laterale questa viene mantenuta fino a quando non si interviene nuovamente sui tasti S e D; supponiamo per esempio che l'astronave stia scendendo verticalmente; se premiamo il tasto S essa acquista una velocità laterale di 10 m/s (cioè ad ogni ciclo si sposta lateralmente di una colonna); a questo punto per far andare l'astronave a destra non è più sufficiente premere una volta sola D ma occorrerà premere due volte. Per essere sicuri che il comando venga eseguito bisogna mante-

nere premuto il tasto desiderato fino a quando non si sente un rumore grave che segnala che il calcolatore ha recepito l'ordine.

Precisazioni sulla velocità verticale e laterale

La parola velocità è stata usata fino ad ora in modo alquanto improprio. Infatti quella che abbiamo chiamato "velocità dell'astronave" non corrisponde all'effettiva velocità, così come comunemente viene intesa. Ciò è dovuto al fatto che l'elaboratore non calcola la posizione dell'astronave nell'istante successivo in base alla posizione attuale e alla velocità; se si fosse fatto così si sarebbe perso quel minimo d'illusione sulla continuità del moto dell'astronave. Il calcolatore, in realtà, opera nel modo che segue: supponiamo che attualmente l'astronave si trovi in una certa riga, nell'istante successivo essa si sposterà qualunque sia la sua velocità discendente (o ascendente) alla riga immediatamente inferiore (o superiore); così facendo si riesce a dare un'illusione di continuità del moto dell'astronave. A questo punto però per non rendere completamente fittizia la "velocità dell'astronave" si è fatto uso di un

ciclo di ritardo per far sì che il tempo che essa impiega per spostarsi di una riga vari a seconda della velocità reale. Quindi se è vero che qualunque sia la velocità dell'astronave questa si sposta sempre di una riga, è anche vero che l'intervallo di tempo che intercorre tra due spostamenti è funzione della velocità stessa. A differenza di altri giochi di simulazione questo programma prosegue (ossia l'astronave continua a scendere) anche se non viene premuto nessun tasto; ne consegue il fatto che il giocatore non ha a disposizione tutto il tempo che vuole per scegliere quale tasto premere, ma che deve agire in tempo reale. Per ottenere tale effetto si è fatto uso di una CALL KEY passante, ossia non seguita dalla solita istruzione di controllo della variabile dello status, ed è quindi ovvio che il comando viene recepito solo se il tasto è premuto durante l'esecuzione della CALL KEY; a tale scopo un suono grave indica il momento in cui il comando stesso viene eseguito.

Terminiamo la descrizione di "Atterraggio" facendo notare che se l'astronave esce lateralmente dal video (senza essersi prima schiantata su di una montagna) essa riappare dalla parte opposta. Al termine di una manche potrete iniziarne un'altra semplicemente premendo il tasto A.

Analisi del listato

Ci limiteremo a descrivere il modo in cui sono memorizzati ed ottenuti i cinque sfondi di gioco poiché il resto del programma pur essendo alquanto lungo non presenta eccessive difficoltà di interpretazione. Gli sfondi sono definiti a partire dalla linea 2430 fino alla 3670; i dati che permettono la loro visualizzazione sullo schermo sono inseriti nei quindici elementi del vettore VA e nelle istruzioni DATA immediatamente successive. Le cinque zone in cui si può atterrare sono stabilite tramite i registri da 1 a 10 accoppiati due a due, ossia VA (1) e VA (2) delimitano la colonna di partenza e di arrivo della prima pista, VA (3) e VA (4) quelle della seconda e così via. Gli elementi da VA (11) a VA (15) contengono invece i coefficienti di difficoltà di ognuna delle cinque zone di atterraggio in ordine progressivo da sinistra a destra. Le relative istruzioni DATA permettono di memorizzare le forme delle montagne, vediamo come: i caratteri impiegati sono quelli aventi il codice ASCII compreso tra 128 e 134, nella DATA vengono posti in ordine sequenziale per ognuna delle 32 colonne i seguenti dati:

a) Riga in cui deve essere stampato il carattere (il programma provvede automaticamente a riempire tutti i caratteri sottostanti).

b) Codice ASCII del carattere che si vuole stampare.

Da notare infine che i caratteri 130-131 e 132-133 vengono sempre usati accoppiati e che sarà sufficiente indicare il codice 130 per la prima combinazione e 133 per la seconda.

```

2350 RESTORE 3850
2360 GOTO 1910
2370 L=0
2380 X=1
2390 GOTO 2220
2400 P=0
2410 F=32
2420 GOTO 2230
2430 REM *** SFONDO N.1 ***
2440 RESTORE 2610
2450 VA(1)=2
2460 VA(2)=4
2470 VA(3)=8
2480 VA(4)=10
2490 VA(5)=15
2500 VA(6)=18
2510 VA(7)=23
2520 VA(8)=25
2530 VA(9)=28
2540 VA(10)=30
2550 VA(11)=4
2560 VA(12)=4
2570 VA(13)=1
2580 VA(14)=2
2590 VA(15)=3
2600 GOTO 350
2610 DATA 16,133,14,129,13,134,13,134,14,130
2620 DATA 16,130,18,130,20,130,21,134,21,134
2630 DATA 21,133,19,134,20,130,22,130
2640 DATA 24,128,24,134,24,134,24,134,24,133
2650 DATA 22,129,21,129,20,129,19,133,17,134
2660 DATA 17,134,18,130,20,128,21,128
2670 DATA 21,134,21,134,22,128,23,128
2680 REM *** SFONDO N.2 ***
2690 RESTORE 2860
2700 VA(1)=4
2710 VA(2)=8
2720 VA(3)=13
2730 VA(4)=15
2740 VA(5)=16
2750 VA(6)=18
2760 VA(7)=22
2770 VA(8)=26
2780 VA(9)=30
2790 VA(10)=32
2800 VA(11)=1
2810 VA(12)=3
2820 VA(13)=2
2830 VA(14)=1
2840 VA(15)=4
2850 GOTO 350
2860 DATA 17,130,19,130,21,130,23,128,23,134
2870 DATA 23,134,23,134,23,134,23,129,22,133
2880 DATA 20,129,19,134,20,130,21,134
2890 DATA 21,134,21,133,19,134,19,134,19,129
2900 DATA 18,129,17,129,16,129,15,134,15,134
2910 DATA 15,134,15,134,16,130,18,130
2920 DATA 20,130,22,130,23,134,23,134
2930 REM *** SFONDO N.3 ***
2940 RESTORE 3110
2950 VA(1)=4
2960 VA(2)=7
2970 VA(3)=12
2980 VA(4)=14
2990 VA(5)=17
3000 VA(6)=19
3010 VA(7)=21
3020 VA(8)=23
3030 VA(9)=29
3040 VA(10)=32
3050 VA(11)=2
3060 VA(12)=4
3070 VA(13)=2
3080 VA(14)=3
3090 VA(15)=2
3100 GOTO 350
3110 DATA 19,133,17,129,16,129,16,128,16,134
3120 DATA 16,134,16,134,17,128,18,130,20,130

```

```

3130 DATA 22,130,24,128,24,134,24,134
3140 DATA 24,133,22,133,20,133,17,134,17,134
3150 DATA 18,128,19,130,20,134,20,134,21,128
3160 DATA 22,130,23,129,22,129,21,133
3170 DATA 19,129,18,134,18,130,18,134
3180 REM *** SFONDO N.4 ***
3190 RESTORE 3360
3200 VA(1)=0
3210 VA(2)=3
3220 VA(3)=6
3230 VA(4)=8
3240 VA(5)=11
3250 VA(6)=13
3260 VA(7)=19
3270 VA(8)=22
3280 VA(9)=29
3290 VA(10)=1
3300 VA(11)=4
3310 VA(12)=2
3320 VA(13)=5
3330 VA(14)=5
3340 VA(15)=4
3350 GOTO 350
3360 DATA 24,134,24,134,24,134,24,133,22,133
3370 DATA 20,133,16,134,16,134,17,130,19,128
3380 DATA 20,128,20,134,20,134,20,133
3390 DATA 18,133,16,133,14,134,15,130,17,128
3400 DATA 17,134,17,134,17,134,17,129,16,129
3410 DATA 15,129,15,128,16,130,18,128
3420 DATA 19,130,20,134,20,134,20,133
3430 REM *** SFONDO N.5 ***
3440 RESTORE 3610
3450 VA(1)=3
3460 VA(2)=8
3470 VA(3)=12
3480 VA(4)=15
3490 VA(5)=17
3500 VA(6)=22
3510 VA(7)=25
3520 VA(8)=27
3530 VA(9)=30
3540 VA(10)=32
3550 VA(11)=2
3560 VA(12)=2
3570 VA(13)=1
3580 VA(14)=3
3590 VA(15)=4
3600 GOTO 350
3610 DATA 20,130,22,130,24,128,24,134,24,134
3620 DATA 24,134,24,134,24,129,23,129,22,133
3630 DATA 20,133,18,133,16,134,16,134
3640 DATA 16,134,17,128,18,128,18,134,18,134
3650 DATA 18,134,18,134,18,134,19,130,21,130
3660 DATA 23,130,24,134,24,134,24,133
3670 DATA 22,133,20,129,19,134,19,134
3680 X=X-31
3690 GOTO 970
3700 X=X+31
3710 GOTO 980
3720 CALL SOUND(300,290,3)
3730 CALL SOUND(200,264,3)
3740 CALL SOUND(200,290,3)
3750 CALL SOUND(300,310,3)
3760 CALL SOUND(500,264,3)
3770 RETURN
3780 DATA 66,82,65,86,79,44,72,65,73,32,70,65
3790 DATA 84,84,79,80,85,78,84,73,77,65,88,61
3800 MAX=P
3810 GOTO 1810
3820 DATA 83,69,73,32,65,84,84,69,82,82,65,84
3830 DATA 79,32,84,82,79,80,80,79,32,86,69,76
3840 DATA 79,67,69,77,69,78,84,69
3850 DATA 77,65,88,61
3860 DATA 72,65,73,32,83,85,80,69,82,65,84,79
3870 DATA 32,73,76,32,76,73,77,73,84,69,32,83
3880 DATA 85,80,69,82,73,79,82,69,77,65,88,61

```