

5 Program Library

In this chapter we will deal with slightly longer comprehensive programs. The first is a sorting program in which data are rearranged. Array variables are used in an interesting manner in this program. The second is an exciting game program. Please note the usage of the various commands from the examples of these programs.

5-1 Rearrangement of Data (Sorting)

Sorting of disorderly data often becomes important in the creation of practical programs. Here we will introduce a program using a method called "bubble sort".

Consider a program to display the ranking of eight examinees from No. 101 to 108 by sorting their points supplied.

• Program List

```
10 REM bubble sort
20 INPUT "Number of data",C
30 REM dimension
40 DIM D(3,C)
50 REM data set
60 FOR A=1 TO C
70 READ D(1,A),D(2
  ,A)
80 NEXT A
90 PRINT "sorting"
  ;
100 REM sort
110 FOR A=1 TO C-1
120 FOR B=A+1 TO C
130 IF D(2,A) > D(2,B
  ) THEN 160
140 D=D(1,A):D(1,A)
  =D(1,B):D(1,B)=
  D
150 D=D(2,A):D(2,A)
  =D(2,B):D(2,B)=
  D
160 NEXT B
170 NEXT A
180 FOR A=1 TO C:D(
  3,A)=A: NEXT A
190 FOR A=1 TO C-1
200 IF D(2,A) > D(2,A
  +1) THEN 220
210 D(3,A+1)=D(3,A)
220 NEXT A
300 REM output
310 FOR A=1 TO C
320 BEEP : PRINT
330 PRINT D(3,A): C
  SR5:"No":D(1,A)
  ; CSR15;">":D(2
  ,A)
340 NEXT A
350 PRINT "OVER"
360 END
1000 DATA 101,40
1010 DATA 102,60
1020 DATA 103,80
1030 DATA 104,60
1040 DATA 105,60
1050 DATA 106,80
1060 DATA 107,20
1070 DATA 108,95
```

501 bytes

- Variables List

Variable	Role	Variable	Role
A	} Loop control variables	D (2, 1)	} Scores
B		:	
C	Number of data	:	
D	Used for conversion	D (2, C)	} Ranking
D (1, 1)	} Examinees' numbers	D (3, 1)	
:		:	
:		:	
D (1, C)		D (3, C)	

- Program Description

Line Nos. 20 – 40 Inputs number of data and declares array.

Line Nos. 50 – 80 Loads data in line No. 1000 and after into array.

Line Nos. 100 – 170 . . . Main routine for sorting. Compares scores successively and swaps the score and the examinee's number with line number 140 and 150 if the large and small scores are reversed.

Line Nos. 180 – 220 . . . Assigns ranking to data sorted according to sequence of scores.

Correspondence of array variables will be as shown below in relation to a person of ranking A

Examinees' Nos. Scores Ranking
 $D(1, A) \longleftrightarrow D(2, A) \longleftrightarrow D(3, A)$

Line Nos. 300 – 360 . . . Displays sorted data in the order of ranking, examinee's No. and score.

Execution example:

Operation	Display
R U N EXE	Number of data?
8	8_
EXE	sortina
	1 No 108 ÷ 95
EXE	2 No 106 ÷ 80
EXE	2 No 103 ÷ 80
EXE	4 No 105 ÷ 60
EXE	4 No 102 ÷ 60
EXE	4 No 104 ÷ 60
EXE	7 No 101 ÷ 40
EXE	8 No 107 ÷ 20
EXE	OVER

- Since the sorted data remain in array D (1, 1)~ D (3, C), they can be used in a different program.

5-2 Horse Race Game

Place your bet by considering the odds on 4 horses in this horse race game. Start with \$20.00 and make a fortune by picking the dark horse.

• Program List

```

10 REM Horse Race
20 CLEAR : DIM A(3
,4)
30 R=1:$="♦♦♦♦"
40 PRINT "< Horse
Race >";
50 FOR J=1 TO 5: B
EEP : BEEP 1: M
EXT J
60 PRINT
70 PRINT "HORSE ";
80 FOR J=1 TO 4: P
RINT J: MID$(J,
1):; NEXT J
90 GOSUB 1000: GOS
UB 1000
100 PRINT : BEEP
110 INPUT "How many
players ",P
120 IF P>5 THEN 110
130 IF P<1 THEN 110
140 PRINT "ALL PLAY
ERS HAVE $20";
150 GOSUB 1000
160 DIM X(2,P),Y$(P
)
170 FOR J=1 TO P:X(
2,J)=20: NEXT J
180 REM initialize
190 G=0
200 FOR J=1 TO 4
210 A(1,J)=0:A(2,J)
= RAN#:A(3,J)=1
+ INT(10*(1.2-A
(2,J)))
220 NEXT J
230 PRINT : PRINT "
<RACE":R:">";:
GOSUB 1000
240 REM bet money
250 FOR J=1 TO P
260 PRINT :X(1,J)=0
:Y$(J)=" "
270 IF X(2,J)=0 THE
N 450
280 PRINT "PLAYER";
J:" HAS $":X(2,
J);
290 GOSUB 1000
300 PRINT : PRINT "
RATE ";
310 FOR K=1 TO 4
320 PRINT CSRK*5: M
ID$(K,1);A(3,K)
;
330 NEXT K
340 GOSUB 1000: BEE
P
350 PRINT CSR0:"P";
J:">";
360 A$= KEY$: IF A$
=" " THEN 360
370 IF A$<"0" THEN
360
380 IF A$>"4" THEN
360
390 N= VAL(A$): IF
N=0 THEN 450
400 A$= MID$(N,1):Y
$(J)=A$
410 BEEP : PRINT :
PRINT "PLAYER";
J:" ":A$;
420 INPUT " MONEY "
,X(1,J)
430 IF X(2,J)<X(1,J
) THEN 410
440 X(2,J)=X(2,J)-X
(1,J)
450 NEXT J
460 PRINT
470 PRINT " < START
! >";
480 FOR K=1 TO 10:
BEEP : NEXT K
490 PRINT
500 REM main loop
510 IF G=2 THEN 600
520 FOR J=1 TO 4
530 IF G<1 THEN 560
540 PRINT CSR A(1,J)
;" ";
550 IF RAN#*(0.9+A(
2,J)/10)>0.7 TH
EN A(1,J)=A(1,J
)+1
560 IF A(1,J)=23 TH
EN G=G+1
570 PRINT CSR A(1,J)
; MID$(J,1);
580 NEXT J
590 GOTO 500
600 REM goal
610 PRINT CSR0:"GOA
L!";
620 FOR J=1 TO 7: B
EEP : BEEP 1: N
EXT J
630 GOSUB 1000
640 FOR J=1 TO 4
650 IF A(1,J)=23 TH
EN H=A(3,J):A$=
MID$(J,1)
660 NEXT J
670 F=0

```

```

680 FOR J=1 TO P
690 M=0: IF X(1,J)=
    0 THEN 730
700 IF Y$(J)=A$ THE
    N M=X(1,J)*H
710 PRINT : BEEP
720 PRINT "PLAYER":
    J;" *PRIZE $":M
    ;
730 X(2,J)=X(2,J)+M
    : GOSUB 1000
740 PRINT : BEEP :
    IF X(2,J)=0 THE
    N F=F+1
750 PRINT "PLAYER":
    J;" HAS $":X(2,
    J):: GOSUB 1000
760 NEXT J
770 PRINT : BEEP :
    IF F=P THEN 830
780 PRINT "REPLAY [
    Y/N] ?":
790 A$= KEY$: IF A$
    =" " THEN 790
800 IF A$="Y" THEN
    R=R+1: GOTO 180
810 IF A$="N" THEN
    830
820 GOTO 790
830 PRINT : PRINT "
    GAME OVER"
840 END
1000 REM timer sub
1010 FOR K=1 TO 150:
    NEXT K
1020 RETURN
1338 bytes
    
```

• Variables List

Variable	Role	Variable	Role
A\$	For keys and characters	K	Loop control variable
A (1, 1)	Position of the spade (♠)	M	For calculating prize
A (1, 2)	Position of the heart (♥)	N	For horse number
A (1, 3)	Position of the diamond (♦)	P	Number of players
A (1, 4)	Position of the club (♣)	R	Race number
A (2, 1)	Random number of (♠)	X (1, 1)	Player 1's bet
A (2, 2)	Random number of (♥)	X (1, 2)	Player 2's bet
A (2, 3)	Random number of (♦)	⋮	⋮
A (2, 4)	Random number of (♣)	⋮	⋮
A (3, 1)	Odds on (♠)	X (1, P)	Player P's bet
A (3, 2)	Odds on (♥)	X (2, 1)	Player 1's holdings
A (3, 3)	Odds on (♦)	X (2, 2)	Player 2's holdings
A (3, 4)	Odds on (♣)	⋮	⋮
F	For determining game over	⋮	⋮
G	For determining goal	X (2, P)	Player P's holdings
H	Odds on winning horse	\$	For selecting a horse's character (♠, ♥, ♦ or ♣)
J	Loop control variable		

• Game Description

One to five persons can play this game with each player starting with \$20.00. There are 4 horses numbered from 1 to 4 with 1 being the spade ♠, 2 the heart ♥, 3 the diamond ♦, and 4 the club ♣. Select a horse from 1 to 4. A player selecting 0 passes a race since there is no horse numbered 0.

The odds on each horse are displayed for each race. If the horse selected by a player wins, the player receives an amount equal to the odds times his bet. If the selected horse does not win, the player loses his bet. If a player's holdings drop to 0, he must drop out of the game. Game is over when all player's lose their holdings.

Following is a description of the sequential displays and key operations in a sample game.

1) Start game.

R **U** **N** **EXE** (Displays title.)

(Describes horse.)

```
<Horse Race>
```

```
HORSE 1♠ 2♥ 3♦ 4♣
```

2) Input number of players.

2 (Enters 2 for two players.)

```
How many players ?
```

```
2_
```

EXE (Initial holding.)

(First race.)

```
ALL PLAYERS HAVE $20
```

```
<RACE 1>
```

3) Input horse and bet.

(Holdings of player 1)

(Odds displayed.)

```
PLAYER 1 HAS $ 20
```

```
RATE ♠ 12 ♥ 3 ♦ 7 ♣ 12
```

(Inputs horse of player 1.)

2 (Selects ♥ horse.)

1 **2** (Bets \$10.00)

EXE (Holdings of player 2)

(Odds displayed.)

(Inputs horse of player 2.)

3 (Selects ♦ horse.)

5 (Bets \$5.00)

```

      ①      ②      ③      ④
      ↓      ↓      ↓      ↓
P 1→ ♠ 12 ♥ 3 ♦ 7 ♣ 12

```

```
PLAYER 1 ♥ MONEY ?
```

```
10_
```

```
PLAYER 2 HAS $ 20
```

```
RATE ♠ 12 ♥ 3 ♦ 7 ♣ 12
```

```
P 2→ ♠ 12 ♥ 3 ♦ 7 ♣ 12
```

```
PLAYER 2 ♦ MONEY ?
```

```
5_
```

4) Race starts.

EXE

(Exciting race is being taken.)

```
< START! >
# # # #
```

5) Race ends.

(Placing decided. # wins.)

(Player 1 loses.)

(Holdings of player 1)

(Player 2 wins \$35.00.)

(Holdings of player 2)

(Do you wish to play again?)

```
GOAL!           #   #   #
PLAYER 1 +PRIZE $ 0
PLAYER 1 HAS $ 10
PLAYER 2 +PRIZE $ 35
PLAYER 2 HAS $ 50
REPLAY [Y/N] ?
```

6) Press Y to advance to the next race and press N for "GAME OVER".

- This game is programmed so the lower the odds the easier to win and horses with high odds are difficult to win. Good luck!