Playing The Game

This program requires a slightly complex starting procedure because it runs separately on two Spectrums. A copy of the program must be loaded into each computer.

Both may be loaded from cassettes but it is much quicker to load one Spectrum from cassette (or Microdrive) and then transmit the program across the network to the other machine.

To do this, type LOAD *"n";0 at the receiving end and SAVE *"n";0 on the machine that has the program in memory. Next, the players should decide who is to shoot first. This player should RUN the program slightly before the second player. The program then assigns network numbers to both machines and works out which copy of the program is playing first and which second.

```
  10  REM Networked Battleships Game
  11  REM
  12  REM 2 Spectrums,Interface 1s
  13  REM June 84/Version 1.6
  15  REM *** init everything
  30  GO SUB 2000: REM right for net
  40  LET s$="                                        ":
 REM 34 spaces
  50  DIM s(8): REM ship types
  60  DIM n$(8,12): REM ship names
  70  DIM a(10,10): REM squared paper'
  80  FOR i=1 TO 8: READ s(i),n$(i): NEXT i
  90  DATA 1,"MTB",1,"MTB",2,"Cruiser",2,"Cruiser"
 100  DATA 3,"Battleship",3,"Battleship",4,"Destro
yer",5,"Carrier"
 110  LET sc=8
 120  GO SUB 3000: REM init screen
 130  GO SUB 4000: REM set up ships
 140  REM now jump depending which
 150  REM player we are.  #1 shoots first
 160  IF sta=2 THEN  GO TO 400
 200  REM *** Take a shot'
 210  LET m$="Your shot": GO SUB 6000
 220  GO SUB 7000: IF e=1 THEN  GO TO 210
 230  OPEN #4;"n";him
 240  PRINT #4;a$
 250  CLOSE #4
 260  REM wait & get results back
 270  OPEN #4;"n";him
 280  INPUT #4;a$
 290  CLOSE #4
 300  LET r=VAL (a$(1 TO 1)): LET x=VAL (a$(2 TO ))
 310  IF r=1 THEN  LET m$="Miss!": PRINT AT 6+q,18
+p;"O";: GO SUB 6000
 320  IF r>1 THEN  LET m$="Hit!": PRINT AT 6+q,18+
p;"X";: GO SUB 6000
 330  IF r>2 THEN  LET m$="You've sunk an enemy "+
n$(x): GO SUB 6000
 340  IF r=4 THEN  LET m$="Congratulations .... Yo
u win!!!": GO SUB 6000: STOP
 400  REM *** Enemy fire
 410  LET m$="Enemy firing": GO SUB 6000
 420  OPEN #4;"n";him
 430  INPUT #4;a$
 440  CLOSE #4
 450  LET p=VAL (a$(2 TO ))+1: LET q=CODE (a$)-64
 460  LET m$="Enemy firing at "+a$: GO SUB 6000
 470  LET x=a(p,q): LET a(p,q)=0
 480  IF x=0 THEN  LET r=1: GO TO 530
 490  LET r=2
 500  LET s(x)=s(x)-1
 510  IF s(x)=0 THEN  LET r=3: LET sc=sc-1
 520  IF sc=0 THEN  LET r=4
 530  LET a$=STR$ (r)+STR$ (x)
 540  OPEN #4;"n";him
 550  PRINT #4;a$
 560  CLOSE #4
 570  IF r=1 THEN  LET m$="It's a miss": PRINT AT
6+q,4+p;"O":
 580  IF r>1 THEN  LET m$=n$(x)+" damaged": PRINT
AT 6+q,4+p;"X";
 585  IF r>2 THEN  LET m$=m$+" and sunk"
 587  GO SUB 6000
 590  IF r=4 THEN  LET m$="Sorry .... you lose": G
O SUB 6000: STOP
 600  GO TO 210
2000  REM *** decide who's who
2005  CLOSE #4
2010  OPEN #4;"n";0
2020  PRINT #4;"1"
2030  CLOSE #4
2040  OPEN #4;"n";0
2045  INPUT #4;a$
2050  CLOSE #4
2060  IF a$="1" THEN  OPEN #4;"n";0: PAUSE 5:
PRINT #4;"2": LET sta=1
```

When the program begins, both players must set up the positions of their ships. This is done by specifying the location of one end of each craft on the 10 × 10 playing grid and saying whether the rest of the ship is up, down, left or right of that position. This sounds complicated but is convenient in practice. Each player has two MTBs (length 1 square), two Cruisers, (length 2 squares), two Battleships (length 3), a Destroyer (4) and an Aircraft Carrier (5).

The players then take turns to shoot at a square on each other's grid, and the program evaluates the result of each shot. A win is achieved by destroying all of one player's ships. Both players should type RUN to play again, remembering that whoever wants to start should RUN first

```
2070  IF a$="2" THEN  LET sta=2
2080  CLOSE #4
2090  FORMAT "n";sta: LET him=3-sta: RETURN
3000  REM *** set up screen
3010  LET col=0: IF sta=2 THEN  LET col=7
3020  PRINT ;:  BORDER 7-col: PAPER 7-col: INK col
: CLS
3030  PRINT TAB 8;"NET BATTLESHIPS"
3040  PRINT : PRINT "PLAYER #";sta
3050  PRINT : PRINT "     YOUR SHIPS     TARGET SHIP
S"
3060  PRINT : PRINT "     0123456789     0123456789
"
3070  FOR i=1 TO 10
3080  PRINT "    ";CHR$ (i+64);".........    ";CHR
$ (i+64);"........."
3090  NEXT i
3100  RETURN
4000  REM *** Set up ships
4010  LET m$="Please position your ships": GO SUB
6000
4020  FOR s=1 TO sc
4030  LET m$=STR$ s+", "+n$(s)+"length "+STR$ (s(s
)): GO SUB 6000
4050  GO SUB 7000: IF e=1 THEN  GO TO 4030
4055  IF s(s)=1 THEN  LET xd=0: LET yd=0: GO TO 41
30
4070  INPUT "U, D, L or R ? ";a$: LET xd=3: LET yd
=3
4080  IF a$="U"  OR a$="u" THEN  LET xd=0: LET yd=
-1
4090  IF a$="D"  OR a$="d" THEN  LET xd=0: LET yd=
1
4100  IF a$="L"  OR a$="l" THEN  LET xd=-1: LET yd
=0
4110  IF a$="R"  OR a$="r" THEN  LET xd=1: LET yd=
0
4120  IF xd=3 AND yd=3 THEN  GO TO 4070
4130  LET l=s(s): LET x=p: LET y=q
4140  IF x<1 OR x>10 OR y<1 OR y>10 THEN  LET m$="
Move the ship away from the edge": GO SUB 6000: GO
 TO 4030
4150  IF a(x,y)<>0 THEN  LET m$="Please reposition
the ship": GO SUB 6000: GO TO 4030
4160  LET x=x+xd: LET y=y+yd
4170  LET l=l-1: IF l>0 THEN  GO TO 4140
4180  LET l=s(s): LET x=p: LET y=q
4190  LET a(x,y)=s: INK s(s): PRINT AT 6+y,4+x;" "
;: INK col
4200  LET x=x+xd: LET y=y+yd
4210  LET l=l-1: IF l>0 THEN  GO TO 4190
4220  NEXT s
4230  LET m$="Prepare for action!!!": GO SUB 6000
4240  RETURN
6000  REM *** Print m$
6010  PRINT AT 20,0;s$;AT 20,0;m$: PAUSE 100: RETU
RN
7000  REM *** Validate co-ords
7010  LET e=0
7015  INPUT "Co-Ords ? ";a$
7020  IF LEN a$<>2 THEN  LET e=1: GO TO 7100
7030  FOR i=1 TO 2
7040  LET c=CODE (a$(i TO i)): IF c>=97 AND c<=122
 THEN  LET a$(i TO i)=CHR$ (c-32)
7050  NEXT i
7060  LET q=CODE (a$(1 TO 1)): LET p=CODE (a$(2 TO
 2))
7070  IF q<65 OR q>74 THEN  LET x=q: LET q=p: LET
p=x
7080  IF q<65 OR q>74 THEN  LET e=1
7090  IF p<48 OR p>57 THEN  LET e=1
7100  IF e=1 THEN  LET m$="Please re-enter co-ordi
nates": GO SUB 6000: RETURN
7110  LET q=q-64: LET p=p-47
7120  RETURN
```