Table of Contents

Type 0 - Player	
IF SCREEN = 25	Title screen
IF F = 99	If it’s time to show the instructions...
LET F 0	...reset indicator so that instructions are
COLOUR 68	only displayed when they are needed.
LET LINE 4	
LET COLUMN 18	
MESSAGE 8	
LET LINE 5	
LET COLUMN 18	
MESSAGE 9	
LET LINE 6	
LET COLUMN 18	
MESSAGE 10	
LET LINE 7	
LET COLUMN 18	
MESSAGE 11	
LET LINE 8	
LET COLUMN 14	
MESSAGE 12	
COLOUR 250	
LET LINE 11	
LET COLUMN 16	
MESSAGE 18	
LET LINE 12	
LET COLUMN 17	
MESSAGE 19	
LET LINE 16	
LET COLUMN 1	
COLOUR 7	
MESSAGE 13	
LET LINE 17	
LET COLUMN 1	
MESSAGE 14	
LET LINE 18	
LET COLUMN 1	
MESSAGE 15	
LET LINE 19	
LET COLUMN 1	
MESSAGE 16	
LET LINE 20	
LET COLUMN 1	
NESSAGE 17	
ENDIF	
IF KEY 4	If fire key pressed, start the game...
LET SCREEN 0	
LET X 160	
LET Y R	
SUBTRACT 32 TO Y	
LET J 13	
LET F 0	
RESTART	
EXIT	
ENDIF	
ENDIF	

```
IF J = 0                                  If game state is “play game”
   IF SCREEN = 25                         Title screen
      ADD 1 TO P                          P is a “stage” counter; Different things happen
      IF P = 30                              at different stages
         LET J 254                        Take off rocket number 1
      ENDIF
      IF P = 31
         LET E 10                         Set rocket to number 2
         LET J 251                        Land rocket number 2
      ENDIF
      IF P = 61
         LET J 254                        Take off rocket number 2
      ENDIF
      IF P = 62
         LET E 13                         Set rocket to number 3
         LET J 251                        Land rocket number 3
      ENDIF
      IF P = 92
         LET J 254                        Take off rocket number 3
      ENDIF
      IF P = 93
         LET E 16                         Set rocket to number 4
         LET J 251                        Land rocket number 4
      ENDIF
      IF P = 123
         LET J 254                        Take off rocket number 4
      ENDIF
      IF P = 124
         LET E 19                         Set rocket to number 5
         LET J 251                        Land rocket number 5
      ENDIF
      IF P = 154
         LET J 254                        Take off rocket number 5
      ENDIF
      IF P = 155
         LET E 7                          Set rocket to number 1
         LET J 251                        Land rocket number 1
      ENDIF
      IF P = 185
         LET P 0                          Repeat take off/land cycle
      ENDIF
      EXIT
   ELSE                                   If not on title screen:
      IF I = 1                            If ready to spawn enemy sprite
         IF C <= 11                       If there are sprites available
            LET F Q                       store rate of enemy appearance
            IF F > 7                      If F > 7, find the modulo
               DIVIDE F BY 7
               MULTIPLY F BY 7
               LET OPT Q
               SUBTRACT F TO OPT
               LET F OPT
            ENDIF
            IF F = 1
               SPAWN 1 2                  1=Meteor; 2=blank sprite
            ENDIF
```

```
            IF F = 2
               SPAWN 2 2          2=Bubble; 2=blank sprite
            ENDIF
            IF F = 3
               SPAWN 3 2          3=Octagon; 2=blank sprite
            ENDIF
            IF F > 3
               IF F <= 6
                  SPAWN 4 2       4=Spinning balls, UFO and Alien; 2=blank sprite
               ENDIF
            ENDIF
            IF F = 7
               SPAWN 5 2          5=Happy/Evil face; 2=blank sprite
            ENDIF
         ENDIF
         LET I 0
      ENDIF
      IF B = 6                     If all of the fuel has been collected
         IF COLLISION 8              and Bob is in collision with a rocket part
            LET FRAME 0                reset sprite frame
            LET IMAGE 2                change sprite image to blank sprite
            LET J 254                  change game state for rocket take-off
            EXIT
         ENDIF
      ENDIF
      IF KEY 0
         LET IMAGE 1               Bob facing right
         IF CANGORIGHT
            SPRITERIGHT
            IF CANGORIGHT
               SPRITERIGHT
            ENDIF
         ENDIF
      ENDIF
      IF KEY 1
         LET IMAGE 0               Bob facing left
         IF CANGOLEFT
            SPRITELEFT
            IF CANGOLEFT
               SPRITELEFT
            ENDIF
         ENDIF
      ENDIF
      IF KEY 3
         IF CANGOUP
            SPRITEUP
            IF CANGOUP
               SPRITEUP
            ENDIF
         ENDIF
      ELSE
         IF CANGODOWN              Automatically “fall” if not moving up
            SPRITEDOWN
            IF CANGODOWN
               SPRITEDOWN
            ENDIF
```

```
         ENDIF
      ENDIF
      IF KEY 4                    Fire the laser
         IF L <= 1                If the fire key isn't already being pressed
            ADD 1 TO L            allow one laser fire per key press
            ADD 4 TO X            position the laser
            LASER IMAGE           fire the laser
            SUBTRACT 4 TO X       reposition
            BEEP 10
         ENDIF
      ELSE
         LET L 0                  Reset key press indicator
      ENDIF
      SPRITEINK 6                 Yellow
      IF A = 0
         ANIMATE
      ENDIF
   ENDIF
ENDIF
IF J > 100
   IF J <= 200
      IF IMAGE = 20
         IF A = 0
            ANIMATE
         ENDIF
         IF FRAME = 5
            LET FRAME 0
            LET IMAGE 2
         ENDIF
      ENDIF
      IF J = 150
         LET X 160
         LET Y R
         SUBTRACT 32 TO Y
         LET FRAME 0
         LET IMAGE 0
         LET J 0
         LET O 0
         EXIT
      ENDIF
      IF J = 200
         LET FRAME 0
         LET IMAGE 20
      ENDIF
      SUBTRACT 1 TO J
   ELSE
      IF J = 250
         LET J 13
         EXIT
      ENDIF
      IF J = 251                  Land the current rocket
         IF SCREEN = 0            On these screens (0,5,10,15,20) the rocket needs
            LET J 0               to be put back together, therefore don't show
         ENDIF                    it ready built by setting J to 0.
         IF SCREEN = 5
            LET J 0
```

```
ENDIF
IF SCREEN = 10
   LET J 0
ENDIF
IF SCREEN = 15
   LET J 0
ENDIF
IF SCREEN = 20
   LET J 0
ENDIF
IF J = 0
   LET M X
   LET N Y
   LET Y R
   LET X 144
   PUT 0
   SUBTRACT 16 TO X
   PUT 1
   LET J 0
   LET X M
   LET Y N
ENDIF
IF J = 251
   IF E = 7
      SPAWN 8 7
   ENDIF
   IF E = 10
      SPAWN 8 10
   ENDIF
   IF E = 13
      SPAWN 8 13
   ENDIF
   IF E = 16
      SPAWN 8 16
   ENDIF
   IF E = 19
      SPAWN 8 19
   ENDIF
   SPAWNED
   LET X 48
   LET Y R
   ORIGINAL
   IF E = 7
      SPAWN 8 6
   ENDIF
   IF E = 10
      SPAWN 8 9
   ENDIF
   IF E = 13
      SPAWN 8 12
   ENDIF
   IF E = 16
      SPAWN 8 15
   ENDIF
   IF E = 19
      SPAWN 8 18
```

Draw the rocket on screen using the correct sprites. Start with the bottom sprite and work upwards towards the top. This is to ensure that each rocket part above "covers" the one below for when fuel is added and the colour of each part changes (the colour of the top parts are not overwritten by changing any below).

```
            ENDIF
            SPAWNED
            LET X 32
            LET Y R
            ORIGINAL
            IF E = 7
               SPAWN 8 5
            ENDIF
            IF E = 10
               SPAWN 8 8
            ENDIF
            IF E = 13
               SPAWN 8 11
            ENDIF
            IF E = 16
               SPAWN 8 14
            ENDIF
            IF E = 19
               SPAWN 8 17
            ENDIF
            SPAWNED
            LET X 16
            LET Y R
            ORIGINAL
            SPAWN 8 4
            SPAWNED
            LET X 70
            LET Y R
            ORIGINAL
         ENDIF
         LET X 160
         LET Y R
         SUBTRACT 32 TO Y
         LET J 252
         EXIT
      ENDIF
      IF J = 253                  Time to set up a few other things...
         LET IMAGE 0              Set player image to left Bob
         LET D 21                 Shields at max strength
         LET H 99
         LET J 0                  On these screens (0,5,10,15,20), the rocket is
         IF SCREEN = 0            not complete so do not spawn the fuel sprite.
            LET J 1
         ENDIF
         IF SCREEN = 5
            LET J 1
         ENDIF
         IF SCREEN = 10
            LET J 1
         ENDIF
         IF SCREEN = 15
            LET J 1
         ENDIF
         IF SCREEN = 20
            LET J 1
         ENDIF
```

```
         IF SCREEN = 25
            LET J 1
         ENDIF
         IF J = 0
            SPAWN 7 2                Fuel sprite
         ENDIF
         LET J 0
         EXIT
      ENDIF
   ENDIF
ENDIF
LET M X
LET N Y
```

Type 1 - Meteor

```
IF J = 0
   IF DIRECTION = 0
      LET DIRECTION PARAMA
      IF CANGODOWN
         ADD 1 TO X
      ELSE
         LET TYPE 6                  Explode
         EXIT
      ENDIF
   ELSE
      SUBTRACT 1 TO DIRECTION
   ENDIF
   IF IMAGE = 22
      IF Y <= 10                     Off screen left
         LET FRAME 0
         LET IMAGE 2
         SUBTRACT 1 TO C
         REMOVE
         EXIT
      ENDIF
      IF FRAME = 7
         IF CANGOLEFT
            SPRITELEFT
         ELSE
            LET TYPE 6               Explode
            EXIT
         ENDIF
      ELSE
         ANIMATE
      ENDIF
   ELSE
      IF Y > 228                     Off screen right
         LET FRAME 0
         LET IMAGE 2
         SUBTRACT 1 TO C
         REMOVE
         EXIT
      ENDIF
      IF FRAME = 7
         IF CANGORIGHT
```

```
            SPRITERIGHT
         ELSE
            LET TYPE 6                  Explode
            EXIT
         ENDIF
      ELSE
         ANIMATE
      ENDIF
   ENDIF
   IF COLLISION 10                      Laser
      SCORE 1
      LET TYPE 6                        Explode
      EXIT
   ENDIF
   IF COLLISION 0                       Player
      OTHER
      LET OPT IMAGE
      ORIGINAL
      IF OPT <> 2                       If not transparent player sprite
         ADD Q TO O                     Increment hit points
         LET TYPE 6                     Explode
         EXIT
      ENDIF
   ENDIF
   SPRITEINK PARAMB
ELSE
   SUBTRACT 1 TO C
   REMOVE
ENDIF
```

```
Type 2 - Bubble
IF J = O
   IF PARAMA = 0
      IF CANGOLEFT
         SPRITELEFT
      ELSE
         LET PARAMA 1                   Bounce off barrier
         LET FRAME 0                    Start “wobble animation
      ENDIF
   ELSE
      IF CANGORIGHT
         SPRITERIGHT
      ELSE
         LET PARAMA 0                   Bounce off barrier
         LET FRAME 0                    Start “wobble animation
      ENDIF
   ENDIF
   IF DIRECTION = 0
      IF CANGOUP
         SPRITEUP
      ELSE
         LET DIRECTION 1                Bounce off barrier
         LET FRAME 0                    Start “wobble animation
      ENDIF
   ELSE
```

```
      IF CANGODOWN
         SPRITEDOWN
      ELSE
         LET DIRECTION 0          Bounce off barrier
         LET FRAME 0              Start “wobble animation
      ENDIF
   ENDIF
   IF A = 0
      IF FRAME <= 7               If playing “wobble” animation
         IF A = 0
            ANIMATE
         EBDIF
      ENDIF
   ENDIF
   IF COLLISION 10                Laser
      SCORE 5
      LET TYPE 6                  Explode
      EXIT
   ENDIF
   IF COLLISION 0                 Player
      OTHER
      LET OPT IMAGE
      ORIGINAL
      IF OPT <> 2                 If not transparent player sprite
         ADD Q TO O               Increment hit points
         LET TYPE 6               Explode
         EXIT
      ENDIF
   ENDIF
   SPRITEINK PARAMB
ELSE
   SUBTRACT 1 TO C
   REMOVE
ENDIF
```

Type 3 - Hunter #1 (Octagon)

```
IF J = 0
   IF DIRECTION = 0
      IF CANGOUP
         SPRITEUP
      ELSE
         IF N <= Y                Change hunter direction depending on where the
            LET DIRECTION 2       player sprite is in relation to this sprite.
         ELSE
            LET DIRECTION 3
         ENDIF
      ENDIF
   ENDIF
   IF DIRECTION = 1
      IF CANGODOWN
         SPRITEDOWN
      ELSE
         IF N <= Y                Change hunter direction depending on where the
            LET DIRECTION 2       player sprite is in relation to this sprite.
         ELSE
```

```
            LET DIRECTION 3
         ENDIF
      ENDIF
   ENDIF
   IF DIRECTION = 2
      IF CANGOLEFT
         SPRITELEFT
      ELSE
         LET DIRECTION 3
      ENDIF
      GETRANDOM 4
      IF RND = 0
         IF M <= X                    Change hunter direction depending on where the
            LET DIRECTION 0           player sprite is in relation to this sprite.
         ELSE
            IF CANGODOWN
               LET DIRECTION 1
            ENDIF
         ENDIF
      ENDIF
   ENDIF
   IF DIRECTION = 3
      IF CANGORIGHT
         SPRITERIGHT
      ELSE
         LET DIRECTION 2
      ENDIF
      GETRANDOM 4
      IF RND = 0
         IF M <= X                    Change hunter direction depending on where the
            LET DIRECTION 0           player sprite is in relation to this sprite.
         ELSE
            IF CANGODOWN
               LET DIRECTION 1
            ENDIF
         ENDIF
      ENDIF
   ENDIF
   IF COLLISION 10                    Laser
      SCORE 7
      LET TYPE 6                      Explode
      EXIT
   ENDIF
   IF COLLISION 0                     Player
      ADD Q TO O                      Increment hit points
      LET TYPE 6                      Explode
      EXIT
   ENDIF
   SPRITEINK PARAMAB
ELSE
   SUBTRACT 1 TO C
   REMOVE
ENDIF
```

Type 4 - Hunter #2 (Spinning balls and others tracking sprites)	
IF J = 0	

```
   SPRITEINK K
   IF IMAGE > 25
      ADD 16 TO X
      SPRITEINK K
      ADD 16 TO Y
      SPRITEINK K
      SUBTRACT 16 TO X
      SUBTRACT 16 TO Y
   ENDIF
   IF M <= X                    Change hunter direction depending on where the
      IF CANGOUP                player sprite is in relation to this sprite.
         SUBTRACT 1 TO X
      ENDIF
   ELSE
      IF CANGODOWN
         ADD 1 TO X
      ENDIF
   ENDIF
   IF N <= Y                    Change hunter direction depending on where the
      IF CANGOLEFT              player sprite is in relation to this sprite.
         SUBTRACT 1 TO Y
      ENDIF
   ELSE
      IF CANGORIGHT
         ADD 1 TO Y
      ENDIF
   ENDIF
   IF COLLISION 10              Laser
      SCORE 6
      LET TYPE 6                Explode
      EXIT
   ENDIF
   IF COLLISION 0               Player
      ADD Q TO O                Increment hit points
      LET TYPE 6                Explode
      EXIT
   ENDIF
   IF IMAGE = 25
      SPRITEINK PARAMAB
   ENDIF
   IF IMAGE = 26
      SPRITEINK 66
      ADD 8 TO X
      SPRITEINK 71
      ADD 8 TO Y
      SPRITEINK 71
      SUBTRACT 8 TO Y
      SUBTRACT 8 TO X
   ENDIF
   IF IMAGE = 27
      SPRITEINK 68
      ADD 8 TO X
      SPRITEINK 69
      ADD 8 TO Y
      SPRITEINK 69
      SUBTRACT 8 TO Y
```

```
      SUBTRACT 8 TO X
   ENDIF
   IF A = 0
      ANIMATE
   ENDIF
ELSE
   SUBTRACT 1 TO C
   REMOVE
ENDIF
```

```
Type 5 - Happy / Evil Ball
IF J = 0
   SPRITEINK K
   LET PARAMA 0
   LET OPT 0
   IF X > M                          Change hunter direction depending on where the
      IF CANGOUP                     player sprite is in relation to this sprite.
         SUBTRACT 1 TO X
      ENDIF
      LET OPT X
      SUBTRACT M TO OPT
   ELSE
      IF CANGODOWN
         ADD 1 TO X
      ENDIF
      LET OPT M
      SUBTRACT X TO OPT
   ENDIF
   IF OPT <= 64                      Is there 64 or less pixels between this sprite
      LET PARAMA 1                   and the player?
   ENDIF
   LET PARAMB 0
   LET OPT 0
   IF Y > N                          Change hunter direction depending on where the
      IF CANGOLEFT                   player sprite is in relation to this sprite.
         SUBTRACT 1 TO Y
      ENDIF
      LET OPT Y
      SUBTRACT N TO OPT
   ELSE
      IF CANGORIGHT
         ADD 1 TO Y
      ENDIF
      LET OPT N
      SUBTRACT Y TO OPT
   ENDIF
   IF OPT <= 64                      Is there 64 or less pixels between this sprite
      LET PARAMB 1                   and the player?
   ENDIF
   LET FRAME 0
   IF PARAMA = 1                     If this sprite is close to the player, change
      IF PARAMB = 1                  its image to be more sinister!
         LET FRAME 1
      ENDIF
   ENDIF
```

```
   IF COLLISION 10                  Laser
      SCORE 10
      LET TYPE 6                    Explode
      EXIT
   ENDIF
   IF COLLISION 0
      ADD Q TO O                    Increment hit points
      LET TYPE 6                    Explode
      EXIT
   ENDIF
   IF FRAME = 0
      SPRITEINK 71
   ELSE
      GETRANDOM 2
      IF RND = 0
         SPRITEINK 66
      ELSE
         SPRITEINK 70
      ENDIF
   ENDIF
ELSE
   SUBTRACT 1 TO C
   REMOVE
ENDIF
```

```
Type 6 - Explosion
IF IMAGE <> 20
   LET FRAME 0
   LET IMAGE 20
   GETRANDOM 5
   ADD 2 TO RND
   LET PARAMA RND
   LET LINE 1
   LET COLUMN 24
   COLOUR 79
   SHOWSCORE
   EXIT
ELSE
   SPRITEINK PARAMA
   IF FRAME = 1
      BEEP 20
   ENDIF
   IF FRAME = 5
      LET FRAME 0
      LET IMAGE 2
      SPRITEINK K
      SUBTRACT 1 TO C
      REMOVE
      EXIT
   ENDIF
   IF A = 0
      ANIMATE
   ENDIF
ENDIF
```

```
Type 7 - Fuel and Rocket Parts
IF PARAMB > 253
   LET G 0
   LET DIRECTION 0
   IF PARAMB = 255
      GETRANDOM 28
      ADD 1 TO RND
      MULTIPLY RND BY 8
      LET DIRECTION 1
      LET Y RND
      LET X 16
      LET IMAGE 3
   ENDIF
   LET PARAMB 0
ENDIF
IF IMAGE = 5
   IF GOT 0
   ELSE
      SPRITEINK 7
      EXIT
   ENDIF
ENDIF
IF IMAGE = 8
   IF GOT 0
   ELSE
      SPRITEINK 7
      EXIT
   ENDIF
ENDIF
IF IMAGE = 11
   IF GOT 0
   ELSE
      SPRITEINK 7
      EXIT
   ENDIF
ENDIF
IF IMAGE = 14
   IF GOT 0
   ELSE
      SPRITEINK 7
      EXIT
   ENDIF
ENDIF
IF IMAGE = 17
   IF GOT 0
   ELSE
      SPRITEINK 7
      EXIT
   ENDIF
ENDIF
IF G = PARAMA                    Is this sprite in collision with player?
   IF M <= X                     Drop the fuel/rocket if too far away from player
      LET RND X
      SUBTRACT M TO RND
```

```
      IF RND > 30
         LET G 0
         BEEP 8
         EXIT
      ENDIF
   ENDIF
   LET RND 0
   IF Y > N                    Drop the fuel/rocket if too far away from player
      LET RND Y
      SUBTRACT N TO RND
   ELSE
      IF N > Y
         LET RND N
         SUBTRACT Y TO RND
      ENDIF
   ENDIF
   IF RND > 30
      LET G 0
      BEEP 8
      EXIT
   ENDIF
   IF Y > N                    Move sprite with player
      IF CANGOLEFT
         SPRITELEFT
         IF CANGOLEFT
            SPRITELEFT
         ENDIF
      ENDIF
   ELSE
      IF Y <= N
         IF Y = N
         ELSE
            IF CANGORIGHT
               SPRITERIGHT
               IF CANGORIGHT
                  SPRITERIGHT
               ENDIF
            ENDIF
         ENDIF
      ENDIF
   ENDIF
   ADD 20 TO M
   IF X > M                    Move sprite with player
      IF CANGOUP
         SPRITEUP
         IF CANGOUP
            SPRITEUP
         ENDIF
      ENDIF
   ELSE
      IF X <= M
         IF X = M
         ELSE
            IF CANGODOWN
               SPRITEDOWN
               IF CANGODOWN
```

```
                  SPRITEDOWN
               ENDIF
            ENDIF
         ENDIF
      ENDIF
   ENDIF
   SUBTRACT 20 TO M
   SUBTRACT 4 TO R
   IF Y > R                    Is fuel/rocket over rocket base?
      ADD 4 TO R
      IF Y <= R
         LET G 99              If so, drop it
         BEEP 8
      ENDIF
      SUBTRACT 4 TO R
   ENDIF
   ADD 4 TO R
ELSE
   IF COLLISION 0
      IF G <> 99               If fuel/rocket not over base
         LET G PARAMA          Grab it
         LET RND X
         DIVIDE RND BY 2
         MULTIPLY RND BY 2
         LET X RND
         BEEP 8
      ELSE
         IF CANGODOWN          If not being held, fall
            ADD 1 TO X
         ENDIF
      ENDIF
   ELSE
      IF IMAGE > 4             This section controls rocket placement when
         IF IMAGE <= 19        building the rocket
            DETECT OBJ
            IF IMAGE = 6
               IF OBJ = 0
                  GET OBJ
               ENDIF
            ENDIF
            IF IMAGE = 5
               IF OBJ = 1
                  GET OBJ
               ENDIF
            ENDIF
            IF IMAGE = 9
               IF OBJ = 0
                  GET OBJ
               ENDIF
            ENDIF
            IF IMAGE = 8
               IF OBJ = 1
                  GET OBJ
               ENDIF
            ENDIF
            IF IMAGE = 12
```

```
      IF OBJ = 0
         GET OBJ
      ENDIF
   ENDIF
   IF IMAGE = 11
      IF OBJ = 1
         GET OBJ
      ENDIF
   ENDIF
   IF IMAGE = 15
      IF OBJ = 0
         GET OBJ
      ENDIF
   ENDIF
   IF IMAGE = 14
      IF OBJ = 1
         GET OBJ
      ENDIF
   ENDIF
   IF IMAGE = 18
      IF OBJ = 0
         GET OBJ
      ENDIF
   ENDIF
   IF IMAGE = 17
      IF OBJ = 1
         GET OBJ
      ENDIF
   ENDIF
   ADD 1 TO X
   IF COLLISION 8
      LET G 1
      IF IMAGE = 5
         LET G 0
      ENDIF
      IF IMAGE = 8
         LET G 0
      ENDIF
      IF IMAGE = 11
         LET G 0
      ENDIF
      IF IMAGE = 14
         LET G 0
      ENDIF
      IF IMAGE = 17
         LET G 0
      ENDIF
      IF G = 0
         GET 1
         LET X 128
         SPAWN 7 2
         SCORE 50          Add 500 to score
         ADD 1 TO LIVES
         COLOUR 79
         LET LINE 1
         LET COLUMN 3
```

```
                DISPLAY LIVES
                BEEP 10
             ELSE
                GET 0
                LET X 144
                SCORE 25                Add 250 to score
                BEEP 10
             ENDIF
             COLOUR 79
             LET LINE 1
             LET COLUMN 24
             SHOWSCORE
             IF SCREEN = 10
                SUBTRACT 8 TO X
             ENDIF
             LET G 0
             LET TYPE 8                 Change to "rocket" type
             EXIT
          ENDIF
          SUBTRACT 1 TO X
       ENDIF
    ENDIF
    IF COLLISION 8                      If touching the rocket
       IF CANGODOWN
          ADD 1 TO X
       ELSE
          ADD 1 TO B
          SCORE 15                      Add 150 to score
          COLOUR 79
          LET LINE 1
          LET COLUMN 24
          SHOWSCORE
          BEEP 8
          LET PARANB 255
          IF B = 6
             BEEP 30
             SUBTRACT 1 TO C
             REMOVE
             EXIT
          ENDIF
       ENDIF
    ELSE
       IF CANGODOWN
          ADD 1 TO X
       ENDIF
    ENDIF
 ENDIF
ENDIF
IF IMAGE = 3
 SPRITEINK 3
ENDIF
IF IMAGE > 4
 IF IMAGE <= 19
    SPRITEINK 7
 ENDIF
ENDIF
```

```
Type 8 - Rocket and Flames
IF J = 250
   SUBTRACT 1 TO C
   REMOVE
   EXIT
ENDIF
IF IMAGE = 4
   SPRITEINK K
ELSE
   IF B = 6
      ADD 1 TO PARAMA
      IF PARAMA = 5
         LET PARAMA 0
         IF PARAMB = 1
            LET PARAMB 0
         ELSE
            LET PARAMB 1
         ENDIF
      ENDIF
   ELSE
      LET PARAMA 0
      LET PARAMB 0
   ENDIF
   IF PARAMB = 0
      LET RND 7
      LET OPT 3
   ELSE
      LET RND 3
      LET OPT 7
   ENDIF
   IF IMAGE = E
      IF B > 1
         SPRITEINK OPT
      ELSE
         IF B > 0
            SPRITEINK OPT
            SUBTRACT 8 TO X
            SPRITEINK RND
            ADD 8 TO X
         ELSE
            SPRITEINK RND
         ENDIF
      ENDIF
   ENDIF
   SUBTRACT 1 TO E
   IF IMAGE = E
      IF B > 3
         SPRITEINK OPT
      ELSE
         IF B > 2
            SPRITEINK OPT
            SUBTRACT 8 TO X
            SPRITEINK RND
            ADD 8 TO X
```

```
         ELSE
            SPRITEINK RND
         ENDIF
      ENDIF
   ENDIF
   SUBTRACT 1 TO E
   IF IMAGE = E
      IF B > 5
         SPRITEINK OPT
      ELSE
         IF B > 4
            SPRITEINK OPT
            SUBTRACT 8 TO X
            SPRITEINK RND
            ADD 8 TO X
         ELSE
            SPRITEINK RND
         ENDIF
      ENDIF
   ENDIF
   ADD 2 TO E
ENDIF
IF J = 252
   IF CANGODOWN
      ADD 1 TO X
      BEEP 5
   ELSE
      IF IMAGE = 4
         SUBTRACT 1 TO C
         REMOVE
         EXIT
      ELSE
         IF SCREEN = 25
            LET J 0
         ELSE
            LET J 253
         ENDIF
      ENDIF
   ENDIF
ENDIF
IF J = 254                          Launch rocket
   IF CANGOUP
      SUBTRACT 1 TO X
      BEEP 5
   ELSE
      IF SCREEN = 25
         LET J 0
         LET F 99
         RESTART
      ELSE
         IF SCREEN <= 23
            SCREENRIGHT
         ELSE
            LET SCREEN 0
         ENDIF
         ADD 1 TO Q
```

```
         LET J 250
         RESTART
      ENDIF
      EXIT
   ENDIF
   IF DIRECTION = 0
      IF IMAGE = E
         IF X <= 138
            LET DIRECTION 1
            SPAWN 8 4
            SPAWNED
            LET FRAME 0
            LET X 160
            LET Y R
            ORIGINAL
         ENDIF
      ENDIF
   ENDIF
ENDIF
IF IMAGE = 4
   SPRITEINK 66
   IF A = 0
      ANIMATE
   ENDIF
ENDIF
```

```
Initialise sprite
ADD 1 TO C
IF TYPE = 0
   LET FRAME 0
   LET IMAGE 2
ENDIF
IF TYPE = 1
   GETRANDOM 4
   ADD 2 TO RND
   LET PARAMB RND
   GETRANDOM 2
   LET DIRECTION RND
   GETRANDOM 144
   ADD 16 TO RND
   LET X RND
   IF DIRECTION = 0
      LET IMAGE 22
      LET Y 232
   ELSE
      LET IMAGE 21
      LET Y 8
   ENDIF
   LET FRAME 0
   GETRANDOM 10
   LET PARAMA RND
   LET DIRECTION RND
ENDIF
IF TYPE > 1
   IF TYPE <= 5
```

```
      GETRANDOM 2
      IF RND = 0
         LET Y 8
      ELSE
         LET Y 232
      ENDIF
      GETRANDOM 2
      IF RND = 0
         LET X 16
      ELSE
         LET X 160
      ENDIF
   ENDIF
ENDIF
IF TYPE > 1
   IF TYPE <= 4
      GETRANDOM 6
      ADD 65 TO RND
      LET PARAMB RND
   ENDIF
ENDIF
IF TYPE = 2
   LET FRAME 0
   LET IMAGE 23
   LET FRAME 8
ENDIF
IF TYPE 3
   LET FRAME 0
   LET IMAGE 24
ENDIF
IF TYPE = 4
   LET FRAME 0
   IF F = 4
      LET IMAGE 25
      GETRANDOM 3
      LET FRAME RND
   ENDIF
   IF F = 5
      LET IMAGE 26
   ENDIF
   IF F = 6
      LET IMAGE 27
   ENDIF
ENDIF
IF TYPE = 5
   LET FRAME 0
   IF F = 5
      LET IMAGE 26
   ELSE
      LET IMAGE 28
   ENDIF
ENDIF
IF TYPE = 7
   LET PARAMA C
   LET PARAMB 255
   IF IMAGE > 4
```

```
      IF IMAGE <= 19
         LET PARAMB 254
      ENDIF
   ENDIF
ENDIF
IF TYPE = 8
   LET DIRECTION 0
ENDIF
```

```
Main loop 1
IF SCREEN = 26
   ENDGAME
ENDIF
IF LIVES = 0
   LET SCREEN 26
   RESTART
ENDIF
IF SCREEN <= 24
   IF J = 0
      IF A = 0                    Update shield status every other frame
         IF D = 0
            SUBTRACT 1 TO LIVES
            COLOUR 79
            LET LINE 1
            LET COLUMN 3
            DISPLAY LIVES
            LET D 21
            LET H 99
            LET J 200
            EXIT
         ELSE
            IF O > 0
               IF O > D
                  LET O D
               ENDIF
               SUBTRACT 1 TO D
               SUBTRACT 1 TO O
            ENDIF
         ENDIF
      ENDIF
      IF I = 0
         LET I 30
         IF Q <= 28
            SUBTRACT Q TO I
         ELSE
            LET I 2
         ENDIF
         GETRANDOM I
         LET I RND
      ELSE
         IF I > 1
            SUBTRACT 1 TO I
         ENDIF
      ENDIF
```

```
      IF A = 1                                  Update shield status every other frame
         IF H > 0
            IF H = 99
               LET OPT 1
               REPEAT 21
                  ADD 1 TO OPT
                  IF OPT <= 12
                     COLOUR 68
                  ELSE
                     COLOUR 66
                  ENDIF
                  LET LINE OPT
                  LET COLUMN 0
                  MESSAGE 6
                  LET LINE OPT
                  LET COLUMN 31
                  MESSAGE 6
               ENDREPEAT
               LET H D
            ELSE
               IF H > D
                  LET OPT 21
                  SUBTRACT H TO OPT
                  ADD 2 TO OPT
                  COLOUR 0
                  LET LINE OPT
                  LET COLUMN 0
                  MESSAGE 6
                  LET LINE OPT
                  LET COLUMN 31
                  MESSAGE 6
                  SUBTRACT 1 TO H
               ENDIF
            ENDIF
         ENDIF
      ENDIF
   ENDIF
ENDIF
IF A = 0
   ADD 1 TO A
ELSE
   LET A 0
ENDIF
```

```
Main loop 2
IF J > 0
   IF J <= 13
      IF SCREEN = 0
         LET R 19
         LET E 7
         LET K 4
         COLOUR 79
         LET LINE 1
         LET COLUMN 1
         MESSAGE 4
```

LET LINE 1	
LET COLUMN 3	
DISPLAY LIVES	
LET LINE 1	
LET COLUMN 24	
SHOWSCORE	
LET LINE 1	
LET COLUMN 30	
MESSAGE 7	
LET LINE 23	
LET COLUMN 1	
COLOUR 0	
MESSAGE 3	
ENDIF	
IF SCREEN = 1	
LET R 11	Column to place rocket
ENDIF	
IF SCREEN = 2	
LET R 16	Column to place rocket
ENDIF	
IF SCREEN = 3	
LET R 26	Column to place rocket
ENDIF	
IF SCREEN = 4	
LET R 6	Column to place rocket
ENDIF	
IF SCREEN = 5	
LET R 15	Column to place rocket
LET E 10	Change rocket
LET K 5	Scenery colour
ENDIF	
IF SCREEN = 6	
LET R 11	Column to place rocket
ENDIF	
IF SCREEN = 7	
LET R 19	Column to place rocket
ENDIF	
IF SCREEN = 8	
LET R 6	Column to place rocket
ENDIF	
IF SCREEN = 9	
LET R 15	Column to place rocket
ENDIF	
IF SCREEN = 10	
LET R 6	Column to place rocket
LET E 13	Change rocket
LET K 2	Scenery colour
ENDIF	
IF SCREEN = 11	
GETRANDOM 3	
IF RND = 0	
LET R 6	Column to place rocket
ENDIF	
IF RND = 1	
LET R 15	Column to place rocket
ENDIF	

```
         IF RND = 2
            LET R 24                    Column to place rocket
         ENDIF
      ENDIF
      IF SCREEN = 12
         LET R 27                       Column to place rocket
      ENDIF
      IF SCREEN = 13
         LET R 15                       Column to place rocket
      ENDIF
      IF SCREEN = 14
         LET R 15                       Column to place rocket
      ENDIF
      IF SCREEN = 15
         LET R 12                       Column to place rocket
         LET E 16                       Change rocket
         LET K 3                        Scenery colour
      ENDIF
      IF SCREEN = 16
         LET R 8                        Column to place rocket
      ENDIF
      IF SCREEN = 17
         LET R 19                       Column to place rocket
      ENDIF
      IF SCREEN = 18
         LET R 19                       Column to place rocket
      ENDIF
      IF SCREEN = 19
         LET R 12                       Column to place rocket
      ENDIF
      IF SCREEN = 20
         LET R 18                       Column to place rocket
         LET E 19                       Change rocket
         LET K 5                        Scenery colour
      ENDIF
      IF SCREEN = 21
         LET R 9                        Column to place rocket
      ENDIF
      IF SCREEN = 22
         LET R 16                       Column to place rocket
      ENDIF
      IF SCREEN = 23
         LET R 21                       Column to place rocket
      ENDIF
      IF SCREEN = 24
         LET R 6                        Column to place rocket
      ENDIF
      IF SCREEN = 25                    Title screen
         LET R 29                       Column to place rocket
         LET K 7                        Scenery colour
      ENDIF
      MULTIPLY R BY 8
      LET A 0
      LET J 251                         Land current rocket
   ENDIF
ENDIF
```

Game Initialisation	
LET LIVES 3	
COLOUR 6	
BORDER 0	
CLS	
LET B 0	Fuel collected counter
LET C 0	Sprite counter
LET D 24	Shield strength
LET E 7	Rocket number 1
LET F 99	
LET G 0	
LET I 0	
LET J 13	
LET K 0	
LET L 0	
LET O 0	
LET Q 1	
LET LINE 0	
LET COLUMN 0	
COLOUR 78	
MESSAGE 1	
LET LINE 1	
LET COLUMN 0	
COLOUR 77	
MESSAGE 3	
LET LINE 1	
LET COLUMN 9	
MESSAGE 2	
LET LINE 23	
LET COLUMN 5	
COLOUR 1	
MESSAGE 5	

Restart screen	
LET B 0	
LET C 1	
LET G 0	

Variables	
A	Animation flip-flop
B	Fuel collected counter
C	Sprite counter
D	Shield strength
E	Current rocket image (bottom sprite no.)
F	Sound count down
G	Fuel collection status - 0 = Not collected Sprite number = Being carried by Bob 99 = Falling to rocket
H	Draw shield strength bar when > 0
I	Enemy populate count down
J	Game state - 0 = Playing the game

	200 = Player dead, hide all sprites 199 - 100 = Wait for player to spawn 251 = Draw rocket on screen 250 = Remove rocket 252 = Rocket landing 254 = Rocket taking off
K	Scenery colour
L	Currently firing laser if > 0
M	Bob’s X co-ordinate
N	Bob’s Y co-ordinate
O	Enemy attack points
P	
Q	Rate of enemy appearance
R	Rocket x position (y co-ord)