

```

1 ｲｲﾁｺﾞｼﾞﾓｸｼﾞｼﾞ
10 CLS:CLP:CLV:H=#900:L=18:M=10:X=19:Y=17
20 POKE#780,255,4,63,127,255,127,18,63,224,0,6,254,224,0:DRAW0,37,63,37:BEEP
30 Y=Y-BTN(30)*(Y>1)+BTN(31)*(Y<22):LCX,Y:GSB500:IFC?"**":GOTO600ELSE?"▲△"
40 IFBTN(32)AND!BB=1:U=Y
50 IFBU=U+1:LCX,U:GSB500:IFCS=S+(C=230):BEEP(C=230)*9:"*":B=0ELSE?"."
100 K=RND(3)-1:IFL>18K=RND(2)-1ELSEIFL<7K=RND(2)
110 IF!KLC0,L:?CHR#(#91+RND(2)*85):GOTO400
120 LC0,L:?CHR#(#98+(K<0)*2)
130 IFRND(2)L=L+KELSELC0,L+K:"|":L=L+K*2
140 IF!RND(5):LC0,RND(L-3)+1:">"
150 LC0,L:?CHR#(#9B-(K<0)*2)
200 P=N*4:W=P+2:V=P+3:[P+1]=[P+1]-1
210 IF[P+1]<0AND[V]>0[W]=[W]+1:[V]=[V]-1:LC[W],[V]:GSB500:IFC=240?"**":GOTO600ELSE?"/"
220 IF![V][P]=0
230 N=N+1:IFN=MN=0ELSEGOTO200
300 WAIT5:FORA=0TOM-1:IF[A*4+1]<0LC[A*4+2],[A*4+3]:?CHR#(0)
310 NEXT
320 POKEH+X+Y*32,0,144:SCROLL1:GOTO30
400 P=N*4:IF![P]AND!RND(3)[P]=1:[P+1]=RND(80/L)+6:[P+2]=[P+1]-1:[P+3]=L:N=0:GOTO200
410 N=N+1:IFN=MN=0:GOTO200ELSEGOTO400
500 C=PEEK(H+POS()):RTN
600 VIDEO2:BEEP5,60:WAIT60:VIDEO1:LC0,0:?S
610 IFINKEY( )=10RUNELSECONT

```

## ■ 変数 ■

- A : 汎用 (FOR~NEXT ループなど)
- B : 爆撃発動中フラグ
- C : 指定座標のキャラクターのコード
- H : 画面左上端 VRAM アドレス
- K : 地形の起伏量
- L : 新規に書き出す地形の Y 座標
- M : 1画面に表示できる地对空ミサイルの最大量
- N : 地对空ミサイル用カウンター
- P : N の計算式省メモリ用 ([P]:スタンバイフラグ、[P+1]:発射カウンター、[P+2]:X 座標、[P+3]:Y 座標)
- S : スコア
- U : 爆撃弾の Y 座標
- V : 地对空ミサイルの Y 座標カウンター省メモリ用
- W : 地对空ミサイルの X 座標カウンター省メモリ用
- X : へりの X 座標
- Y : へりの Y 座標

## ■ 処理の内訳 ■

- 10~20 行 : 初期設定 (変数リセット、へりのキャラクター定義、平地描画)
- 30~50 行 : へり関連 (操作、当たり判定、描画、爆撃弾)
- 100~150 行 : 地形、空対空ミサイル、敵基地描画
- 200~230 行 : 地对空ミサイル描画、当たり判定
- 300~320 行 : 地对空ミサイル、へり消去、スクロール
- 400~410 行 : 地对空ミサイルセッティング
- 500 行 : 指定座標のキャラクターコード取得
- 600~610 : ゲームオーバー処理

★ プログラムは IchigoJam BASIC 1.5 系準拠で組まれています。1.4 系でも動きますが動作はかなり遅くなります。  
 対応策として 10 行の地对空ミサイル量 M の値を 4~6 に、300 行の WAIT を削除し、ゲームバランスとして地对空ミサイルが減った分 130 行の RND(5) の値を 2~3 に変えて空対空ミサイルの出現率を増やすなどすると良いでしょう。

※ 30, 320 行を少々変更し、へり後ろの“・”が正しく出るようにいたしました。[20250209]