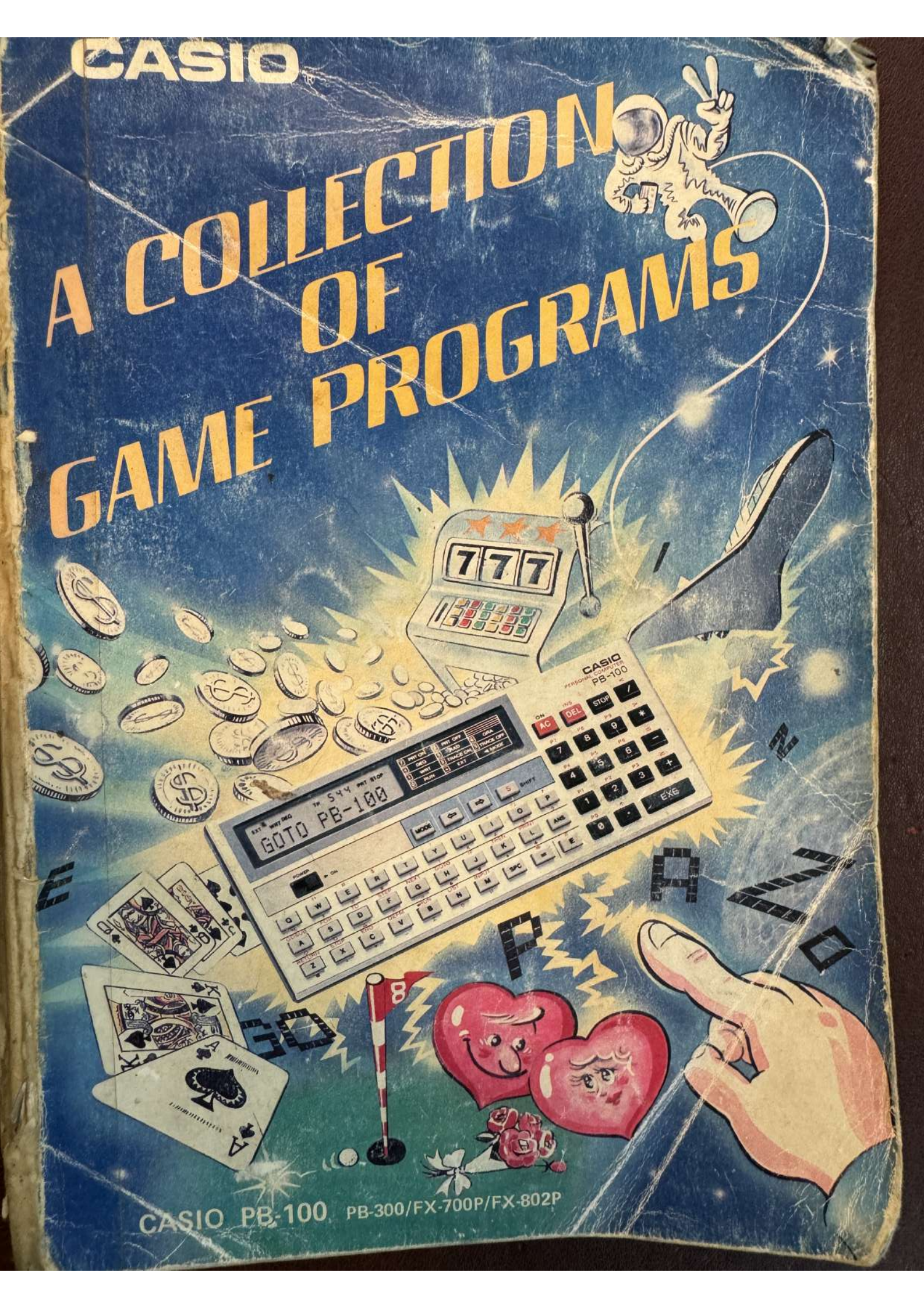
CASIO

A COLLECTION OF GAME PROGRAMS



CASIO PB-100 PB-300/FX-700P/FX-802P

Test your speed and skill against a hard-swinging challenger!

Are you ready to take on a powerful challenger in an exciting game of luck and skill? The Casio personal computer is waiting to take you on with a whole gallery of games – from Royal Sevens and War of Numbers to a thrilling Moon-Landing simulation. Unlike everyday TV games, with the computer you carry out all steps from key-in to operation on your own. It's not only fun, but educational too!

Programming can all be easily performed using the computer keyboard. Just clear the program area and the computer starts winking at you. You then write your program. When that's finished, you start the RUN. If an error message appears, you call up the incorrect number and correct it. Thanks to BASIC, even program debugging is fun.

Once you've got it all set, the computer is ready to do battle. Take up your position and prepare for a fast and exciting contest.

Ready! Set! Go!

CONTENTS

HOW TO USE THIS BOOK. .1
SPOT AND STOP 4
GOPHER TRAP. 8
COMPATIBILITY TEST . .12
NUMBER-ATTACK 15
NUMBER-RECALL 19
KNOCK-A-BLOCK 23
COCKROACH ATTACK . .27
CRAZYBALLOON. 31
NERVE-RACKER. 35
PICKPOCKET. 39
SPACE LANDING. 43
ZERO ATTACK. 48
MOON-LANDING. 52
ALPHABET ATTACK. . . .57
CHANCE IT. 63
ROYAL SEVENS. 67
WAR OF NUMBERS. 72
DRAGON-SLAYING 77
POISON BERRIES. 82
BASEBALL GAME 86
GOLF GAME. 90
FLASHY FIVE 95
BOMBARDMENT GAME .102
SLOT-MACHINE GAME .107
BLACKJACK GAME. . . .113

HOW TO USE THIS BOOK

Each game description is divided into the following sections:

1. **TITLE and GENERAL DESCRIPTION**
   Read this section to get a general idea of the game.

2. **RULES and KEY OPERATION**
   In this section, the rules of the game are described in detail and the keys used to play the game are listed. Read this section carefully to understand how the game is played.

3. **DISPLAY**
   This section is mainly divided into the following 3 displays:

   (1) **Starting Display**: When the program is started.
   (2) **Play Display**: During the game
   (3) **End of Game**: When the game is over.

   Easy-to-understand explanations on how to read each of the above displays are given in this section.

4. **HOW TO PLAY (EXAMPLE)**
   A step-by-step illustration of how a game is played is given in this section.

5. **VARIABLES**
   This section lists the variables used in the program and their descriptions.

6. **PROGRAM**
   The program list is given in this section. Input the program in your computer to play the game.

* Parts of the display may sometimes become too dim to read during a game. If this happens, adjust the contrast.

NOTES

1. Program writing is performed in line units. Up to 62 characters, including the line number, can be written in each line. The blinking cursor (—) is displayed until 55 characters have been input. From the 56th character on, a blinking ■ will be displayed instead. Press EXE after inputting each line.

2. Some of the program lines given here will have more than 62 characters if one-key commands are used. In such cases, type out the commands using the alphabet keys, leaving no space between letters.

3. However, blank spaces enclosed in quotation marks (" ") within a character string must be retained. Input the same number of blank spaces by pressing key SPC.

Example: **"PICKPOCKET"** (see page 42)

```
[LIST 210] PRINT CSR 0;"AR
           REST!      ":S=S
                      ↑
                 5 spaces (press SPC)
```

4. To start a game, press S P9 . Key S is the red-lettered shift key. Do not mistake it with the alphabet "S" key.

5. Several game programs given in this book have more than 544 steps. With the PB-100, use the OR-1 optional RAM pack (maximum 1,568 steps) to enjoy these games.

6. Remember to press the following keys before and after you input a program into the computer.

BEFORE PROGRAM INPUT

Before you input the program, press the following keys:

WRT 1 . . . . . .(Write mode)
CLEAR A EXE . . . . . .(Program area clear)
S , ALPHA , (~ . . .) EXE
↑ Shift key — (or a number which is indicated at the start of the program list)

PROGRAM INPUT

Press V A C EXE after completing input.

AFTER PROGRAM INPUT

Don't forget to press the following keys after completing your program input:

WRT ⟲ . . . . . .(run mode)

* In this book, keys which are used are indicated for PB-100 key arrangement. For the PB-300, FX-700P and FX-802P key positions vary slightly.

SPOT AND STOP

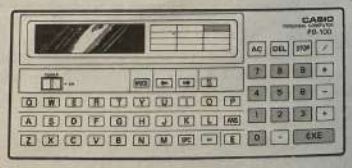
There's a sneaky old snake hidden in the grass. It shows itself for an instant – but you have to look carefully, because it's camouflaged. Your goal is to find the snake and note its position, all in a very short time.

RULES AND KEY OPERATION

The snake and its clump of grass appear on the display for a very short instant and then disappear. Immediately after this, numbers 0–9 are displayed. You input the number corresponding to the position that you think the snake was in. If you're right, you score 20 points. If you miss, half your score will be subtracted. The game ends after 20 tries.

The keys used are the numerical keys [0] – [9].

These are used to input the snake's position. After pressing one of these keys, always press [EXE].



DISPLAY

• Starting display

The game title, the highest score so far, and the name of the highest scorer are displayed, after which the play dispaly appears.

SPOT AND STOP

⇩

HIGHEST=0

⇩

BY------

↑ Name of highest scorer

• Play display

::::;::::

⇩ Grass ↑ Snake ⇩ Grass

0123456789?

⇩ Indicate position with number

When you're right

GOOD

When you're wrong

MISSED

• End of game

SCORE=40

If your score is the highest so far

NEW HIGH

NAME?

Name input await condition

HOW TO PLAY (EXAMPLE)

Start by pressing [S] [P0]

1 SPOT AND STOP

[EXE]

2 HIGHEST=0

[EXE]

3 BY

[EXE]

4 : : : : : : : : : :

Where's the snake? Oops, it's gone!

5 0 1 2 3 4 5 6 7 8 9 ?

Was it 6?

Press [6] [EXE]

6 MISSED

Wrong!

12 ; : : : : : : : : : :

13 0 1 2 3 4 5 6 7 8 9 ?

[0] [EXE] This was easy

14 GOOD

[EXE]

15 : : : : : : : : : :

16 0 1 2 3 4 5 6 7 8 9 ?

[6] [EXE] This snake sure is sneaky

17 GOOD

[EXE]

38 SCORE=72

[EXE]

39 NEW HIGH

[EXE]

40 NAME?

[L][I] NIDIA [EXE]

- Variables

Name of variable	Description	Name of variable	Description
A	Random number for selecting grass display pattern	L	Score this time
C	Input data	$	Name of highest scorer
G	Number of tries (count) (up to 20)	W	Loop counter
H	Highest score so far		

PROGRAM

[MODE] [1]

CLEAR A [EXE]

[S] [OPTN] [0] [EXE]

```
 5 L=0
10 PRINT "SPOT AND
   STOP"
20 PRINT "HIGHEST=
   ";H
30 PRINT "BY ";$
40 FOR G=1 TO 20
50 A=INT (RAN#*10)
60 GOSUB 300
70 INPUT C
80 IF A=C:L=L+20:P
   RINT "GOOD":GOT
   O 100
90 L=L/2:PRINT "MI
   SSED"
100 NEXT G
110 L=INT L:PRINT :
   PRINT "SCORE=";
   L
120 IF L>H:H=L:PRIN
   T "NEW HIGH":IN
   PUT "NAME",$
130 END
300 PRINT ":::::::
   ::";
310 PRINT CSR A;":"
   ;:FOR W=1 TO 40
   :NEXT W
315 PRINT
320 PRINT "01234567
   89";
330 RETURN
```

VAC [EXE]

[MODE] [0]

Total 258 steps

GOPHER TRAP

This game gives you a chance to see how fast your reflexes are. As soon as a gopher pops out of its hole, you have to trap it within the time limit to prevent its escape. The time it takes you to do this will be displayed after each successful trap. If you miss and the gopher escapes, you don't score any points.

RULES AND KEY OPERATION

There are 26 gophers, each represented by a letter in the alphabet, hidden in their respective holes. You try to trap a gopher as it pops out of its hole and prevent its escape. The faster you trap it, the higher your score. The time it takes you to trap a gopher will be displayed after each successful move. To trap a gopher, you have to press the alphabet key which corresponds to the letter which appeared on the display. (This is the gopher that popped out of its hole.) If you press the wrong key or fail to press the right key within the time limit (about 2 seconds), the gopher escapes, and you don't score any points. The game is over when 5 gophers have popped out of their holes.

The keys used are the 26 alphabet keys, [A] – [Z].

DISPLAY

- **Starting display**
  When the program is started, first the highest score so far is displayed, followed by the actual play display.

```
HIGHEST: 300
```

- **Play display**

```
R
```

↑ Gopher

When a gopher is trapped

```
TIME : 24
```

↑ Lapse of time before a key is pressed

```
SCORE: 60
```

↑ Score

When a gopher escapes
(Due to wrong input or end of time limit)

```
MISSED : 0
```

- **Score display ....**
  Game over

```
FINAL : 220
```

↑ Final score

HOW TO PLAY (EXAMPLE)

Start by pressing [S] [P6]

1 HIGHEST:330

[EXE]

2 D

Press [D] Out pops gopher D!

3 TIME:10

4 SCORE:60

5 W

Press [W] This time it's gopher W!

6 TIME:7

7 SCORE:90

8 S

Time is up, and gopher S disappears from the display

9 MISSED:0

10 B

Out pops goher B!

Oops! Press the wrong key.
Press [G]

11 MISSED:0

12 P

Press [P] Out pops gopher P!

13 TIME:9

14 SCORE:70

15 FINAL:300

Can you beat this score next time?

- Variables

Name of variable	Description	Name of variable	Description
A	Random number generation for gopher display	G	Score
B	Gopher display position	H	Highest score so far
C	Number of gophers displayed (loop counter)	D$	Gopher
E	Time (count)	F$	Key input data

PROGRAM

[MODE] [1]
CLEAR A[EXE]
[S] [OPTN] [CL] [EXE]

```
10 PRINT "HIGHEST:
   ";H:G=0:$="ABCD
   EFGHIJKLMNOPQRS
   TUVWXYZ"
20 FOR C=0 TO 4
30 PRINT CSR 0;"
                "
40 A=INT (RAN#*26+
   1):B=INT (RAN#*
   12)
50 D$=MID$(A,1)
60 FOR E=0 TO 25:P
   RINT CSR B;D$;:
   F$=KEY:IF F$=""
   :NEXT E
70 IF D$=F$:PRINT
   :PRINT "MISSED:
   0";:GOSUB 200:N
   EXT C:GOTO 100
80 PRINT :PRINT CS
   R 0;"TIME :";E;
   :GOSUB 200
90 S=(27-E)*10:PRI
   NT CSR 0;"SCORE
   :";S;:G=G+S:GOS
   UB 200:NEXT C
100 IF H<G:H=G
110 PRINT :PRINT CS
   R 0;"FINAL:";G;
   " "
120 END
200 FOR W=1 TO 200:
   NEXT W:RETURN
```

VAC[EXE]
[MODE] [CL]
Total 310 steps

COMPATIBILITY TEST

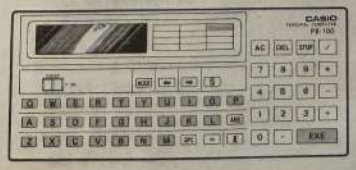
This checks the compatibility of a boy and girl.
First, all the letters in their names are converted into numbers; then, the numbers are used to calculate their compatibility. By manual calculation, this takes at least 3 minutes, but by computer it takes only about 10 seconds.

RULES AND KEY OPERATION

Input the names of the 2 persons whose compatibility you want to check, in this order: the boy's last name, his first name, the girl's last name, and her first name. The total number of letters input must not exceed 30. After completing this input, the computer asks you to wait while it goes through the calculations. After about 10 seconds, the compatibility rate is displayed.

The keys used are the alphabet keys and EXE .

The alphabet keys are used to input the two names. EXE is pressed after inputting the names.



DISPLAY

- **Starting display**

NAMES(MAX30)?
Awaiting input of names

WAIT
Wait about 10 secs.

- **Result**

47 percent
↑
Compatibility rate

HOW TO PLAY (EXAMPLE)

Start by pressing [S] [PRGM]

1 NAMES(MAX30)
Display moves leftward

2 AMES(MAX30)?
Input the two names

3 YOUNGCLIFFAD
Input Cliff Young and Helen Adams, last names first, consecutively

4 AMSHELEN
EXE

5 WAIT

6 97 percent
Highly compatible!
You're ideal for each other!

- **Variables**

Name of variable	Description	Name of variable	Description
A	Counter	E	Number of letters in the names input
B	Position of the letters in the name (loop counter)	F	Compatibility rate
C	Loop counter	G$	A letter in the names input
D	Counter	I(D)	Number corresponding to a letter in the names input

PROGRAM

MODE 1

CLEAR A EXE

S DEFM 1 3 EXE

```
10 VAC
20 INPUT "NAMES(MA
   X30)",$
30 PRINT "WAIT";
40 E=LEN($)
50 FOR B=1 TO E
60 G$=MID(B,1)
70 IF G$="A";I(A)=
   1:A=A+1
80 IF G$="I";I(A)=
   2:A=A+1
90 IF G$="U";I(A)=
   3:A=A+1
100 IF G$="E";I(A)=
   4:A=A+1
110 IF G$="O";I(A)=
   5:A=A+1
120 NEXT B
130 $=""
140 FOR C=A TO 3 ST
   EP -1
150 FOR D=0 TO C-2
160 I(D)=10*FRAC ((
   I(D)+I(D+1))/10
   )
170 NEXT D:NEXT C
190 F=I(0)*10+I(1)
200 IF F=10;IF I(2)
   =0;F=100
210 PRINT CSR 0;F;"
   percent ";END
```

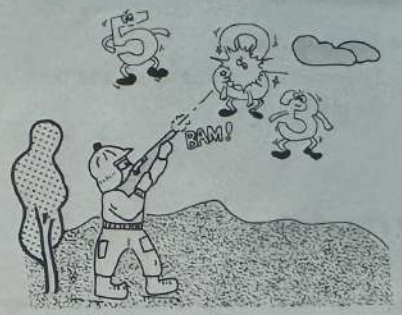
VAC EXE

MODE 0

Total 312 steps

NUMBER-ATTACK

Here your mission is to attack the numbers that appear in quick succession on the display. When you succeed in hitting a number, it moves to the left, and you score a point. This game requires quite a bit of manual dexterity, a quick eye and fast reflexes.

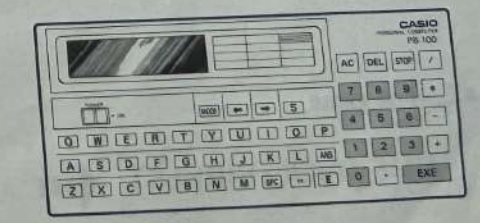


RULES AND KEY OPERATION

Random numbers between 0 and 9 appear and disappear on the display one after another in quick succession. You try to hit each number by pressing the corresponding numerical key before the number disappears. If you succeed in hitting a number, it moves to the left, and you score a point. If you press the wrong key or take too long to attack, the number disappears. 50 numbers appear on the display in one game.

The keys used are the numerical keys (0 – 9).

They are used to attack the numbers which appear on the display. You press the key that corresponds to the number displayed.



DISPLAY

- **Starting display**
  The game title is displayed, and then the play display.

NUMBER–ATTACK

- **Play display**

5

↑ The number to attack

- **Score** (when a number is hit)

#5

↑ Hit

⇩

#5

⇩ Moves to the left

#5

⇩

#5

⇩

#5

＊1＊

↑ Score

- **End of game**

TIME OVER ; HIGHEST ⟵ 25 YOUR SCORE 8

Display moves leftward

HOW TO PLAY (EXAMPLE)

Start by pressing S P0

1 NUMBER–ATTACK

Display moves leftward

2 K

3 8

Press ⊟

4 #8

Hit

5 #8

9 #8

The number moves leftward

13 #8

14 ＊1＊

Score

15 5

Press ⊟ — Too bad, it's the wrong number

16 1

Continue in the same way until 50 numbers have been attacked

120 TIME OVER ; HI

Display moves leftward

121 GHEST 31 YOUR SCORE 21

Press ↵ (To start the next game)

122 NUMBER–ATTACK

- **Variables**

Name of variable	Description	Name of variable	Description
A	Number to be displayed	S	Score for this game
H	Highest score	K	Loop counter
I	Loop Counter	K$	Numbers of variable A
M	Count on number of attacks	J$	Number input

PROGRAM

MODE 1

```
CLEAR A
```

S DEFM

```
10 H=0
11 PRINT "NUMBER-A
   TTACK";:FOR K=1
    TO 100:NEXT K
15 S=0:M=0:$="0123
   456789"
20 A=INT (RAN#*10)
   :IF A>10;A=10
30 PRINT :PRINT CS
   R 9;A;" ";
40 J$=MID(A+1,1)
50 M=M+1
60 FOR I=1 TO 20:K
   $=KEY
80 IF K$=J$ THEN 2
   00
90 NEXT I
100 IF M=50 THEN 14
   0
110 GOTO 20
140 IF S>H;H=S
150 PRINT :PRINT "T
   IME OVER;HIGHES
   T";H;"YOUR SCOR
   E";S;
170 STOP :GOTO 15
200 S=S+1:PRINT :PR
   INT CSR 9;"*";K
   $;
220 FOR I=1 TO 9
230 PRINT " ";
240 NEXT I
250 PRINT :PRINT CS
   R 6;"*";S;"* ";
   :FOR K=1 TO 50:
   NEXT K
260 GOTO 100
```

```
VAC
```

MODE

```
Total 317 steps
```

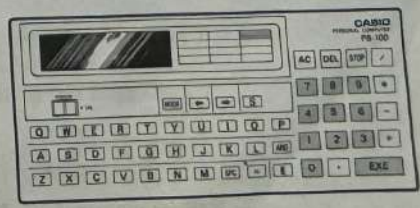
NUMBER-RECALL

In this game, you test your powers of concentration and recall. You have to correctly recall the numbers which appear on the display. Each number is shown only for an instant. Immediately after the number disappears, you have to recall and input the number. The computer will indicate whether you are right or wrong. Some numbers are short and easy to recall, but some are so long that you have to be really quick even to read the entire number before it disappears. In addition, the time that each number is displayed shortens as the game progresses.

RULES AND KEY OPERATION

A number randomly chosen by the computer is displayed for an instant and then disappears. Recall the number and input it. If your answer is correct, "RIGHT!" will be displayed; if incorrect, "WRONG!". There are 10 numbers to recall in Group 1. If your rate of correct recall is 70% or over, you're allowed to go on to challenge the next group. With each succeeding group, you have 2 more numbers to recall and less time to read each number. See how many groups you can challenge.

The keys used are 0 – 9 **and** EXE .



DISPLAY

- **Starting display**
  Group number

```
GROUP 1
```

- **Play display**

The number to be recalled

```
8025637
```

```
NUMBER?
```

```
ANSWER=8025637
```

Display moves leftward if more than 12 digits are used

Correct recall

```
RIGHT!
```

Incorrect recall

```
WRONG!
```

- **Score**

8 correct out of 10

```
RIGHT=8
```

Rate of correct recall

```
(80%)
```

2 incorrect out of 10

```
WRONG= 2
```

Rate of incorrect recall

```
(20%)
```

- **When the rate of correct recall is below 70%**

```
GAME OVER
```

- **When Group 10 is completed successfully**

```
GREAT!
```

HOW TO PLAY (EXAMPLE)

Start by pressing [S] [P0]

1 `GROUP 1` [EXE]

2 `97953`
The number to be recalled

3 `NUMBER?`
97953 [EXE] (The number recalled is input)

4 `ANSWER=97953` [EXE]

5 `RIGHT!` [EXE]

6 `768721045` [EXE]

7 `NUMBER?`
768721043 [EXE]

8 `ANSWER=768721045`
(Display moves leftward)

9 `WRONG!`

38 `562`
The number to be recalled

39 `NUMBER?`
562 [EXE]

40 `ANSWER=562` [EXE]

50 `RIGHT!` [EXE]

51 `RIGHT=8`
8 correct out of 10
[EXE]

52 `(80%)`
80% correct recall
(You can go on to Group 2 since it's over 70%)

```
53 WRONG=2
   2 incorrect
54 (20%)
   Rate of incorrect recall
55 GROUP 2
56 326710
```

```
106 RIGHT=22
    22 answers correct out of 36
107 (61.1%)
    Rate of correct recall
108 WRONG=14
109 (38.9%)
110 GAME OVER
    The game ends because your rate of
    correct recall is below 70%.
```

- **Variables**

Name of variable	Description	Name of variable	Description
A	Loop counter	E	Answer input
B	Number of digits	J	Number of correct answers
C	Number to be recalled	M	Rate of correct recall
D	Display time		

PROGRAM

```
MODE 1
CLEAR A
S DEFM
```

```
5 VAC
8 T=3:Y=10
10 J=0:L=L+1
20 PRINT "GROUP";L
30 FOR A=1 TO Y
40 B=INT (RAN#*10)
50 C=INT (RAN#*10^
   B)
60 IF C=0 THEN 40
70 PRINT C;
80 FOR D=1 TO T*50
   :NEXT D
90 PRINT CSR 0;"NU
   MBER  ";
100 INPUT E
110 PRINT "ANSWER="
   ;C
120 IF C=E;PRINT "R
   IGHT!":J=J+1:GO
   TO 140
130 PRINT "WRONG!"
140 NEXT A
150 M=RND(J/Y*100,-
   2)
160 PRINT "RIGHT=";
   J,"(";M;" %)"
170 PRINT "WRONG=";
   Y-J,"(";100-M;"
   %)"
180 IF M<70:PRINT "
   GAME OVER":END
190 T=T-.3:IF T<0;P
   RINT "GREAT!":E
   ND
200 Y=Y+2:GOTO 10
```

```
VAC
MODE
```

Total 326 steps

KNOCK-A-BLOCK

In Knock-a-Block, you start with a wall of 12 blocks. Your goal is to destroy the wall by knocking off the blocks. But it's not as easy as it sounds. You'd almost think the wall has a mind of its own!

RULES AND KEY OPERATION

The wall starts as 12 blocks arranged 6 across and 2 high. It's divided into 3 sections, located over the numbers 4, 5 and 6 as follows:

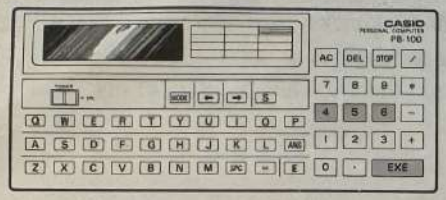
```
= =  = =  = =
4    5    6
```

To eliminate a block, you press either the number 4, 5 or 6, and if luck is on your side, you score a hit in the section you've selected and a block disappears.

However, you cannot choose which block to hit; the computer makes that decision. You win 2 points for every block you eliminate. If you miss, in other words, if you hit a space in a section where the block has already been knocked off, you lose 2 points. When 10 blocks are knocked off, you receive a bonus of 10 points, and you can continue with a new game. On the other hand, when you miss four times, the game is over. If you score the highest number of points won so far, you may enter your name in the computer using the alphabet keys.

The keys used are: 4 , 5 and 6 .

These keys specify the section of blocks from which a block is to be knocked off. The alphabet keys are used when entering the name of the highest scorer in the computer.



DISPLAY

- **Starting display**

```
HIGHEST 0
```
Score

```
by
```
Name of highest scorer

```
======: 0
```
blocks — Your score (lose 2 points for each miss)

- **End of game**

```
SCORE 96
```
Your final score

- **If you score the highest number of points**

```
HIGHEST
```

```
NAME?
```

- **When playing another game**

```
AGAIN(EXE)
```

HOW TO PLAY (EXAMPLE)

Start by pressing S P0

1 `HIGHEST 92` EXE

2 `by` EXE Name of highest scorer

3 `======:0`
Press 4

4 `=-====:2` ↑ Score
Press 6

5 `=-====-:4`
Press 5

6 `=-=-=-:6`

16 `- - -:16`
Press 6

17 `- - :18`

18 `======:28`
Press 5 — Knocked off 10 blocks, Bonus points added

19 `==-===:30`

28 `SCORE 100` EXE

29 `HIGHEST` EXE

30 `NAME?`
K A T H Y EXE — Enter your name

31 `AGAIN(EXE)` EXE

32 `HIGHEST 100` EXE

33 `by KATHY`
Return to Step 3

- **Variables**

Name of Variable	Description	Name of Variable	Description
G$	Key input	A$(0)-A$(5)	Blocks
I	Score	W	Loop counter
J	0 or 1	N	Loop counter
K	Number of misses	M	Highest score
L	Number of blocks knocked off	$	Highest scorer

PROGRAM

MODE 1

```
CLEAR A
S DEFM EXE
```

```
10 I=0:K=0:L=0
20 PRINT "HIGHEST"
   ;M
30 PRINT "by ";$
40 FOR N=0 TO 5:A$
   (N)="=":NEXT N
50 FOR N=0 TO 5:PR
   INT CSR N;A$(N)
   ;:NEXT N:PRINT
   ":";I;
51 IF I<10;PRINT "
   ";
60 G$=KEY:IF G$≠"4
   ";IF G$≠"5";IF
   G$≠"6" THEN 60
70 J=(VAL(G$)-4)*2
   +INT (RAN#*2)
80 IF A$(J)=" ";K=
   K+1:I=I-2:GOTO
   130
90 IF A$(J)="=";A$
   (J)="-":GOTO 11
   0
100 IF A$(J)="-";A$
    (J)=" "
110 L=L+1:I=I+2
120 IF L=10;I=I+10:
    L=0:GOTO 40
130 IF K>3 THEN 150
140 GOTO 50
150 PRINT CSR 0;"SC
    ORE";I;"   "
160 IF I>M;M=I:PRIN
    T "HIGHEST":INP
    UT "NAME",$
170 PRINT "AGAIN(EX
    E)  ":GOTO 10
```

VAC EXE

MODE

Total 374 steps

COCKROACH ATTACK

You're trying to get rid of cockroaches with an insecticide spray, but the exasperating little buggers won't keep still long enough for you to take careful aim. As you chase after the scuttling creatures, if you happen to be pressing the spray nozzle just when one crosses your path, you've succeeded in killing it. If you miss, it escapes.

RULES AND KEY OPERATION

Using your insecticide spray, you chase after a cockroach as it scuttles right and left on the display. If you happen to be pressing either of the 2 keys used to move the spray when the cockroach crosses your path, you succeed in killing it with the insecticide, and you score a point. But if you fail to press either of such keys just when the cockroach crosses your path, it escapes.

The keys used are ◻ and ⊡. These are used to move the insecticide spray.

Key ◻ is for moving the spray leftward.
Key ⊡ is for moving the spray rightward.

DISPLAY

- Starting display

```
COCKROACH ¥

HIGHEST: 0
          ↑
   Highest score so far
```

- Play display

```
¥      <■>      0
↑       ↑       ↑
Cockroach Spray  Score
```

- Score

```
<¥>      1
 ↑       ↑
Killed  Score
```

- End of game

```
Game Over !!

Killed ¥−10
        ↑
Number of cockroaches killed

<EXE> AGAIN
Press [EXE] to start the next game
```


CRAZYBALLOON

In this game you chase after and try to bust a floating balloon with an arrow. Busting a balloon may be easy, but it is pretty hard to catch a balloon that's floating about. In this game the balloon and the arrow may seem to move in a somewhat awkward manner, but you will fully feel a sense of impatience while pursuing the balloon and a sense of satisfaction when you succeed in making a hit.

RULES AND KEY OPERATION

The balloon sways right and left so restlessly that you drive it to bay. By busting the balloon, you win up to 300 points. The sooner the balloon is popped, the higher your score. You start with a time limit (countdown) of 60, which diminishes by one number each time the arrow is moved. When you hit the balloon, the time limit increases by 0 to 7. If you fail to bust the balloon after a long time (before the countdown reaches zero), the game is over.

The keys used are [1], [2], and [−]. These are used to move the arrow.

Key [1] is for moving the arrow leftward.
Key [2] is for moving the arrow rightward.
Key [−] is for moving the arrow in an indefinite direction.

DISPLAY

- **Starting Display**

The following display appears on the display at the start of the game:

```
( 1 ) / ( 2 ) / ( − )
```

The game can be started by pressing any of the keys, [1], [2] and [−]. The choice is up to you. Even after hitting the balloon during the game, continue to press one of the keys.

- **Play Display**

```
↑        O          55
↑ Arrow  ↑ Balloon  ↑ Countdown
```

- **When the arrow hits the balloon**

```
*
⇩
200
↑ Score
```

- **End of Game**

```
GAME OVER
⇩
SCORE 1540 HIGHEST 3760
↑ Score this time  ↑ Highest score so far
```

HOW TO PLAY (EXAMPLE)

1 Start by pressing [S] [P0]

```
( 1 ) / ( 2 ) / ( − )
```

2 Start by pressing key [1]

```
↑  O        59
```

3 Press key [2] to move the arrow rightward

```
↑  O        58
```

4 Press key [2] to move the arrow rightward

```
↑O          57
```

5 Press key [2] to move the arrow rightward

```
*
```

6 Score display after a hit

```
200
```

7 Continue with key [−]

```
↑O          61
```

8 Press key [2] to move the arrow rightward

```
O↑          60
```

9 Press key [1] to move the arrow leftward

```
↑O          59
```

10 Jump the arrow by key [−]

```
O  ↑        58
```

11 Jump the arrow by key [−]

```
↑O          57
```

12 Press key [2] to move the arrow rightward

```
*
```

13 Score display after a hit

```
170
```

14 Continue with key [1]

```
↑O          48
```

25

```
O↑          0
```

26 Game over

```
GAME OVER
```

27 Score display

```
SCORE 1540HI
```

28 Display moves leftward

```
GHEST 3760
```

- Variables

Name of Variable	Description	Name of Variable	Description
B	Balloon position	S	Total score
H	Highest score	T	Score
I	Time limit	Y	Arrow (↑) position
J	Loop counter	A$	Key input command
L	Loop count	K	Loop counter

PROGRAM

MODE 1

```
CLEAR A EXE
S DEFM ⊡ EXE
```

```
10 S=0:I=60
15 PRINT "(1)/(2)/
   (-)";
20 B=INT (RAN#*10)
   :L=4+RAN#*5
30 FOR J=1 TO L:I=
   I-1
40 IF I<0 THEN 400
50 A$=KEY:IF A$=""
   THEN 50
60 IF A$="1" THEN
   160
70 IF A$="2" THEN
   170
80 IF A$="-" THEN
   180
90 GOTO 50
160 Y=Y-INT (RAN#*3
   )-1:GOTO 200
170 Y=Y+INT (RAN#*3
   )+1:GOTO 200
180 Y=INT (RAN#*10)
200 IF Y<0;Y=8
210 IF Y>8;Y=0
220 IF Y=B THEN 300
230 PRINT :PRINT CS
   R 9;I;
232 PRINT CSR B;"O"
   ;
235 PRINT CSR Y;"↑"
   ;
240 NEXT J
250 GOTO 20
300 T=10*INT (RAN#*
   10):I=I+INT (RA
   N#*6)
310 PRINT :PRINT CS
   R B;"*";
320 FOR N=1 TO 50:N
   EXT N
330 PRINT CSR 0;"";
   CSR B;T;:S=S+T
340 GOTO 20
400 IF S>H;H=S
410 PRINT :PRINT "G
   AME OVER";:FOR
   K=1 TO 150:NEXT
   K
420 PRINT :PRINT "S
   CORE";S;"HIGHES
   T";H:FOR K=1 TO
   100:NEXT K
```

VAC EXE

MODE ⊡

Total 428 steps

NERVE-RACKER

You need a good memory to play this game, and good guesswork, too. 10 cards, placed face-down, hide 5 pairs of matching cards (numbers). Revealing 2 cards at a time, you must find the 5 matching pairs. When the 2 cards you turn over don't match, they are returned to their original face-down positions. The fewer turns you require, the higher your score.

RULES AND KEY OPERATION

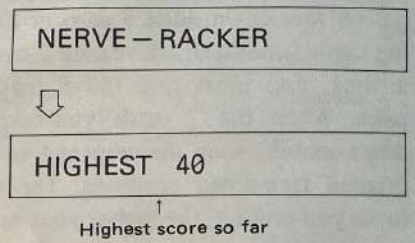
When the program is started, the computer takes a while to deal out 10 cards face-down on the display. When this is completed, you turn over a card at random. Then you turn over another one, trying to find the one with the same number. If the 2 cards match, they are left open. If they don't, they are returned to their original face-down positions. Remember what their numbers were so you can later come back to them when their matching partners appear. Continue in the same way, always revealing 2 cards at a time. The game ends when you've matched all 10 cards. The fewer turns you require, the higher your score.

The keys used are the numerical keys ⊡ – 9 and EXE.

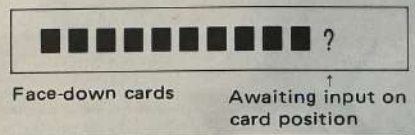
Press a numerical key to input the position (counted from the left) of the card you want to see. Press key ⊡ to reveal the card at the far left, key 1 to reveal the 2nd card from the left, key 2 for the 3rd, and so on. Key 9 is pressed to see the 10th card. After pressing a numerical key, press EXE.

DISPLAY

- **Starting display**
  The game title and highest score so far are displayed before the play display

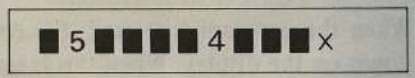


- **Play display**

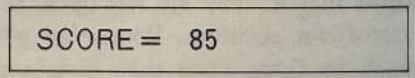


- **When the cards match**

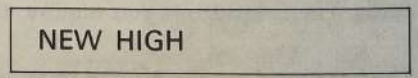
- **When the cards don't match**



- **End of game**



- **When your score is the highest so far**



HOW TO PLAY (EXAMPLE)

Start by pressing [S] [P0]

1 NERVE-RACKER [EXE]

2 HIGHEST 42 [EXE]

3 ■■■■■■■■■■? [0] [EXE] Wonder what the first card is ...

4 2■■■■■■■■■? [5] [EXE] Try to guess right

5 2■■■■3■■■■X [EXE] Too bad ·····

6 ■■■■■■■■■■? [7] [EXE] What's the 7th card?

7 ■■■■■■■2■■? [0] [EXE] Do you remember where the other 2 is?

8 2■■■■■■2■■O Right! ·····

[EXE]

9 2■■■■■■2■■?

28 213144■23■? [6] [EXE]

29 2131445 23■? [9] [EXE]

30 21314452350 [EXE]

31 SCORE= 85 [EXE]

32 NEW HIGH

• Variables

Name of variable	Description	Name of variable	Description
S	Card data	S	Number of matches made
N	Loop counter	A(0)~ A(9)	Cards lined up at random
M	Loop counter	X	Number of first card from left
W	Loop counter	Y	Number of 2nd card from left
U	Highest score so far	T	Score
Q	Number of times cards are selected	O	Working area

PROGRAM

MODE 1

```
CLEAR A EXE
S DEFM ⊘ EXE
10 PRINT "NERVE-RA
   CKER":$="12345"
   :FOR N=0 TO 9:A
   (N)=0:NEXT N
20 FOR M=1 TO 5:FO
   R M=1 TO 2
30 O=INT (RAN#*10)
   :IF A(O)<>0 THEN
    30
40 A(O)=-M:NEXT M:
   NEXT N
50 PRINT "HIGHEST
   ";U
60 Q=0:S=0
70 X=-1:Y=-1:Q=Q+1
80 GOSUB 200:INPUT
    X:IF X>9:X=-1:
   GOTO 80
90 GOSUB 200:INPUT
    Y:IF Y>9:Y=-1:
   GOTO 90
100 GOSUB 200
110 IF X<>Y:IF X≥0:I
   F Y≥0:IF A(X)=A
   (Y):IF A(X)<0 T
   HEN 130
120 PRINT "x":GOTO
   70
130 PRINT "o"
140 A(X)=ABS (A(X))
   :A(Y)=ABS (A(Y)
   ):S=S+1:IF S≥5
   THEN 160
150 GOTO 70
160 T=100-(Q-5)*5:P
   RINT "SCORE=";T
170 IF T>U:U=T:PRIN
   T "NEW HIGH"
180 END
200 PRINT CSR 0;:FO
   R N=0 TO 9:O=AB
   S (A(N))
210 IF O<>A(N):IF X<
   N:IF Y<>N:PRINT
   "■";:GOTO 230
220 PRINT MID(O,1);
230 NEXT N:RETURN

VAC EXE
MODE ⊘
Total 433 steps
```


PICKPOCKET

Madam X is shopping at a department store. A pickpocket is snooping around her, looking for a chance to steal her wallet. Madam X, completely unaware, moves from one counter to another busily shopping. You, the store detective, have to go after the pickpocket and catch him before he manages to get away with the wallet.

RULES AND KEY OPERATION

First, the pickpocket (P), Madam X (M) and you (I) are displayed in that order from left to right. When you input your move (1–2 space(s) leftward or 1–2 space(s) rightward), you move to the position indicated. But be careful, as both the pickpocket and Madam X also change positions at the same time, and you have no way of knowing where they will move next. Keep chasing the pickpocket until you either catch him (ARREST) or he outsmarts you and makes his getaway with Madam X's wallet (ESCAPED). When you succeed in arresting the pickpocket, you score 10 points. When the pickpocket escapes with the wallet, the game ends.

The keys used are 4 , 1 , · , 3 and 6 (5 keys).

These are used to move "I" leftward or rightward, as you try to catch the pickpocket. P and M move at random.

Key 4 is for moving 2 spaces leftward.
Key 1 is for moving 1 space leftward.
Key · is for remaining in the same position.
Key 3 is for moving 1 space rightward.
Key 6 is for moving 2 spaces rightward.

DISPLAY

- **Starting display**
  When the program is started, the highest score so far is displayed, after which the starting play is displayed.

```
HIGHEST : 10
```

- **Play display**
  When only P and I are displayed, you (I) and Madam X (M) are in the same position.

```
P   M   I
↑       ↑
Pickpocket  Madam X  You
```

- **When the pickpocket escapes with the wallet, appears**

```
ESCAPED
```

- **When you arrest the pickpocket, appears**

```
ARREST!
```

- **Score and end of game**

```
SCORE : 10
    ⇩
(EXE) AGAIN
```

(If you press EXE . you can start another game.)

HOW TO PLAY (EXAMPLE)

Start by pressing S P0

1 `HIGHEST : 10`

EXE

2 `P  M  I`

Press 4 to move 2 spaces leftward

3 `P M  I`

There's the pickpocket!

Press 4 to move 2 spaces leftward

4 `M  PI`

Press 1 to move 1 space leftward.

5 `M  I`

6 `ARREST!`

EXE

The pickpocket has been arrested

7 `P  M  I`

Press 4 to move 2 spaces leftward

8 `PM  I`

Here he comes again!

Press 1 to move 1 space leftward

9 `P M I`

Press 4 to move 2 spaces leftward

10 `M  PI`

Press · to remain in position.

11 `P MI`

Stop!

Press 4 to move 2 spaces leftward

12 `PI`

Press 1 to move 1 space leftward

13 `MI P`

Press 6 to move 2 spaces rightward

14 `M  I`

I've caught you red-handed

15 `ARREST!`

EXE

16 `P  M  I`

⋮

22 `I  P  M`

Press 3 to move 1 space rightward

23 `I  P`

24 `ESCAPED`

The wallet's gone!

25 Game Over : 20

26 (EXE)AGAIN

(Press EXE to start another game)

- Variables

Name of variable	Description	Name of variable	Description
P	Pickpocket's position	J	Move
H	Highest score so far	M	Madam X's position
I	Your position	S	Score

PROGRAM

```
MODE 1
CLEAR A EXE
S DEFM EXE

 5 VAC
10 S=0:$="41.36"
20 PRINT "HIGHEST:
   ";H
30 P=2:M=5:I=8
40 GOSUB 300
50 M=M+INT (RAN#*5
   )-2
60 IF M<0:M=0
70 IF M>11:M=11
80 P=P+INT ((RAN#*
   5)-2+(M-P)/2)
90 IF P<0:P=0
100 IF P>11:P=11
110 A$=KEY:IF A$=""
    THEN 110
120 FOR N=1 TO 5:IF
    A$=MID$(N,1):J=
    N-3:GOTO 140
130 NEXT N:GOTO 110
140 I=I+J
150 IF I<0:I=0
160 IF I>11:I=11
170 GOSUB 300
180 IF I=P THEN 210
190 IF P=M THEN 220
200 GOTO 50
210 PRINT CSR 0;"AR
    REST!      ":S=S
    +10:GOTO 30
220 PRINT CSR 0;"ES
    CAPED      ":PRI
    NT "Game Over";
    S
230 IF S>H:H=S
240 PRINT "(EXE) AG
    AIN":GOTO 10
300 PRINT CSR X;" "
    ;CSR Y;" ";CSR
    Z;" ";
305 PRINT CSR M;"M"
    ;CSR P;"P";CSR
    I;"I";
310 X=M:Y=P:Z=I:RET
    URN

VAC EXE
MODE
Total 436 steps
```


SPACE LANDING

A spaceship lander – just detached from its main unit – is now free-falling, tail-first, towards an unknown planet. To avoid a headlong fall onto the planet's surface, you have to ignite the lander's rockets and slow down its descent. The amount of fuel you burn determines how much its speed is reduced – but decide quickly! Unless you manage to reduce the lander's speed to 100 or less by the time it reaches a height of 100, it makes a crash (!) landing.

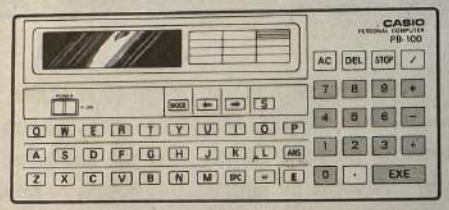
RULES AND KEY OPERATION

When the lander starts descending, its height is at 6000, its speed 200, and its fuel supply 35. As it free-falls towards the planet's surface, its speed accelerates by 5. In other words, its height decreases while its speed increases. You must control – slow down – the lander's descent by igniting its rockets to produce counterthrust. To do this, input the quantity of fuel you want to burn. Once the rockets are ignited, the lander's fuel supply continues to decrease by the amount input and its speed continues to change in accordance with the amount of fuel being burned. For example, if you input 1.0, the speed decelerates by 5. If you input 0.3, the speed accelerates by 2. While the lander's fuel supply lasts, you can change the amount of fuel being burned anytime. You have guided the lander to a safe landing if at height 100 its speed is 100 or less. Otherwise, the lander makes a crash landing. Your score depends on the landing conditions of the lander (height, speed, fuel supply).

The keys used are the numerical keys [0] – [9], and [+], [−], [×] and [EXE].

These are used to input the amount of fuel to burn, each key designating a different amount, as follows:

	(Quantity of Fuel Burned)	(Speed)	(Change in Speed)
Key [0]	0.0	+5	(acceleration)
Key [1]	0.1	+4	(acceleration)
Key [2]	0.2	+3	(acceleration)
Key [3]	0.3	+2	(acceleration)
Key [4]	0.4	+1	(acceleration)
Key [5]	0.5	0	(constant)
Key [6]	0.6	-1	(deceleration)
Key [7]	0.7	-2	(deceleration)
Key [8]	0.8	-3	(deceleration)
Key [9]	0.9	-4	(deceleration)
Key [+]	1.0	-5	(deceleration)
Key [−]	1.2	-7	(deceleration)
Key [×]	1.5	-10	(deceleration)



DISPLAY

- **Starting display**
  The game title and start condition are displayed before the play display.

```
SPACE LANDING
START<EXE>
```

- **Play display**

```
4330-211  33
↑Height  ↑Speed  ↑Remaining fuel supply
```

- **End of game**
  Successful landing

```
GREAT!
69   85   27
↑Height  ↑Speed  ↑Remaining fuel supply
SCORE 1200
```

Crash landing

```
C-R-A--A--S-H!!
Display moves leftward
YOU'RE DEAD!
69-193   19
↑Height  ↑Speed  ↑Remaining fuel supply
```

HOW TO PLAY (EXAMPLE)

Start by pressing [S] [P0]

1 SPACE LANDING

[EXE]

2 START(EXE)

[EXE]

3 6000-200 35

Height Speed Fuel supply

4 5798-205 35

Press [0] to ignite rockets

5 5595-201 34

12 4460-178 28

Press [8] to change amount of fuel burned

13 4284-175 28

15 3940-169 26

Press [7] to change amount of fuel burned

16 3772-167 25

30 1630-139 15

Press [*] to change amount of fuel burned

31 1496-129 14

35 1060- 89 10

Press [-] to change amount of fuel burned

36 974- 82 8

41 652- 47 2

Note remaining fuel supply

42 608- 40 1

43 566- 45 1

Nearing the planet's surface but running out of fuel — will it land safely?

51 46- 85 0

52 GREAT!

53 42- 89 0

54 SCORE 2900

- **Variables**

Name of variable	Description	Name of variable	Description
G	Acceleration	S	Score
H	Height	V	Speed
P	Amount of fuel burned	A$	Key input data
Q	Remaining fuel supply	W	Loop counter

PROGRAM

[MODE] [1]

CLEAR A [EXE]

[S] [DEFM] [0] [EXE]

```
10 PRINT "SPACE LA
   NDING":SET F0
20 V=-200:H=6000:Q
   =35:G=5:P=0
30 PRINT "START(EX
   E)"
40 PRINT CSR 0;"
          ";
50 PRINT CSR 9;Q;C
   SR 5;V;CSR 0;H;
60 FOR Z=1 TO 30:A
   $=KEY:NEXT Z
70 IF Q<=0 THEN 140
80 IF A$>="0";IF A$
   <="9" THEN 130
90 IF A$="+";P=1
100 IF A$="-";P=1.2
110 IF A$="*";P=1.5
120 GOTO 140
130 P=VAL(A$)/10
140 IF P>Q:P=Q
150 Q=Q-P:G=P*10-G:
    H=H+V+G/2:V=V+G
160 IF H>0 THEN 40
170 IF ABS H<=100:IF
     ABS V<=100:PRIN
    T CSR 0;"GREAT!
          ":GOTO 19
    0
180 PRINT CSR 0;"C-
    R-A--A--S-H!!";
    "YOU'RE DEAD!":
    GOSUB 210:END
190 GOSUB 210:S=(10
    0-ABS H)*(100-A
    BS V)*(Q*10+G)
200 PRINT "SCORE";S
    :END
210 PRINT CSR 9;Q;C
    SR 5;V;CSR 0;H;
    RETURN
```

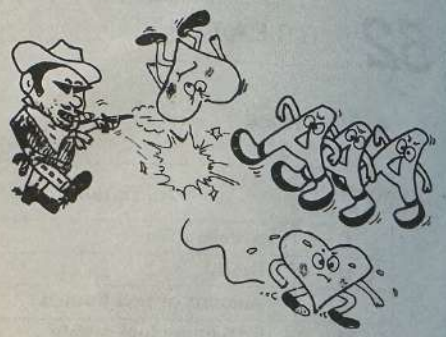
VAC [EXE]

[SHIFT] [0]

Total 437 steps

ZERO ATTACK

In this game, displayed letters are eliminated one by one. Complete elimination has to be achieved within a specified number of operations. There are eight successive patterns and each pattern has to be overcome before you can proceed to the next pattern.

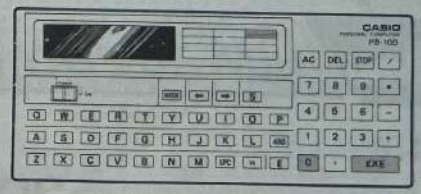


RULES AND KEY OPERATION

The moment you start the game, ten A's will be displayed. The symbol " - " will move rightward from the leftmost end. Attack (press key ▣) the moment it is superimposed on an A, and the asterisk mark " * " will appear at the point attacked and the point will become blank. An attack on a blank space is considered ineffective. Eliminate the ten A's in this way. The number of attacks is limited to 15. (In other words, the allowable number of ineffective attacks is 5.) If an A or A's remain after 15 attacks, the game is over. If you succeed in eliminating all the A's, not more than eight ♠'s will be displayed at random points on the display. Attack these ♠'s in the same way as the A's. In this case the number of attacks is limited to 10.

The pattern display changes from A's to ♠, ♥, ◆, ♣, ○, □'s and △'s. The same rules apply to attack on ♠'s and the subsequent patterns. When all △'s are eliminated, the word "GREAT!!" is displayed and the game is over.

The Keys used are ▣ and EXE.
Each press of the ▣ key constitutes one attack.



DISPLAY

- **Play Display (A pattern)**

```
AAAAAAAAAA
----AAAAAA
-----*AAAA
A  AA  AAAA
```

- **♠ ~ △ Pattern Display**

♠ ♠ ♠ ♠

♥♥ ♥ ♥♥

◆◆ ◆ ◆ ◆

♣ ♣♣ ♣

○○○ ○ ○

□□ □□ □ □

△ △△ △ △

- **When all patterns have been eliminated**

```
GREAT!!
```

HOW TO PLAY (EXAMPLE)

Start by pressing [S] [P0]

1 AAAAAAAAAA

2 ---AAAAAAA

Attack by pressing [▭]

3 --*AAAAAAA

4 AA AAAAAAA

Attack by pressing [▭]

5 -----*AAAA

6 AA AA AAAA

Attack by pressing [▭]

7 -* AA AAAA

8 A AA AAAA

30 -------A A

Attack by pressing [▭]

31 ------* A

32 A

33 ----------A

Attack by pressing [▭]

34 ---------*

35 ♠ ♠ ♠ ♠

36 ---♠ ♠ ♠

- Variables

Name of Variable	Description	Name of Variable	Description
L	Loop count	$	Character data
J	Number of patterns	B$	Attack
K	Loop count	C$	Character specification
N	Loop count	W	Loop count
P$(0)~P$(9)	Character data		

NOTE

In the program, allow 10 spaces within the " " (quotation marks) in lines 140 and 180.

PROGRAM

[MODE] [1]

```
CLEAR A[EXE]
[S][DEFM][▭][EXE]
10 N=15:J=0
20 FOR I=0 TO 9:P$
   (I)="A":NEXT I
30 GOSUB 500
40 FOR K=1 TO N
60 B$="1":PRINT $;
70 FOR I=0 TO 9
80 PRINT CSR I;"-"
   ;:FOR Z=1 TO 10
   :NEXT Z
90 IF B$="0" THEN
   140
100 B$=KEY:IF B$="0
   " THEN 200
110 NEXT I
120 PRINT CSR 0;
130 GOSUB 500
140 IF $="
    " THEN 230
150 NEXT K
160 FOR I=0 TO 10:P
    RINT P$(I);:NEX
    T I
170 FOR Z=1 TO 100:
    NEXT Z
180 IF $*"
     ":PRINT CSR 0;
    "GAME OVER ":EN
    D
200 P$(I)=" "
210 PRINT CSR I;"*"
    ;
220 GOTO 120
230 J=J+1
240 IF J=1:C$="♠"
250 IF J=2:C$="♥"
260 IF J=3:C$="♦"
270 IF J=4:C$="♣"
280 IF J=5:C$="○"
290 IF J=6:C$="□"
300 IF J=7:C$="△"
310 IF J=8 THEN 400
320 FOR I=1 TO 8
330 P$(INT (RAN#*5)
    *2+1)=C$
340 NEXT I
350 N=10
360 GOTO 30
400 PRINT "GREAT !!
    "
410 END
500 $="":FOR I=0 TO
    9:$=$+P$(I):NE
    XT I
510 RETURN
```

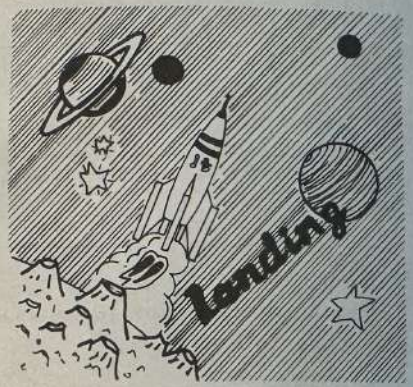
VAC[EXE]

[MODE] [▭]

Total 490 steps

MOON-LANDING

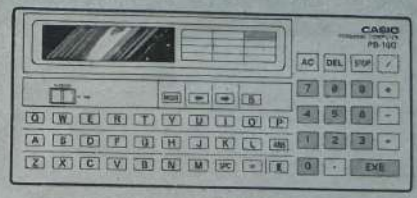
Your task is to land a spaceship successfully on the moon. The spaceship is descending towards the moon's surface, but unless you reduce the speed of descent, you'll crash-land. To slow your descent, you have to ignite your retrorockets to achieve counter-thrust. For a safe landing, your velocity should be between -2 and 2 km/sec by the time the spaceship touches down. The less fuel you burn and the less time it takes to land, the higher your score.



RULES AND KEY OPERATION

The game starts with the spaceship descending at a speed of 20 km/sec from a height of 100 km. The descending speed accelerates by 2 km/sec (not true in real life, by the way) due to the pull of gravity. To avoid crash-landing, you must reduce the speed of your descent by igniting your retrorockets. To do this, input the quantity of fuel you want burned (any quantity from 0 to 40 kl) after the "ROCKET?" display. Each kl of fuel burned reduces your velocity by 0.5 km/sec. You start with a fuel supply of 180 kl, and if you run out of fuel before the spaceship reaches the surface, you'll free-fall the rest of the way and end in disaster. Your landing is successful if your velocity is between -2 and 2 km/sec when the spaceship touches the moon's surface (when HEIGHT = 0). The less fuel you burn and the less time it takes to land, the higher your score.

The keys used are the numerical keys [0] – [9] and [.].



DISPLAY

- **Starting display**

```
MOON-LANDING
```

- **Play display**

```
h 100 s 20 f 180
```

Height | Speed | Fuel └Fuel at game start
Starting height Speed at game start

```
)                .
```

Moon's surface 1 space = 10 km Spaceship

```
ROCKET?
```

Awaiting fuel quantity input

- **End of game**

Successful landing

```
LANDING
```

```
TIME 23.5
```

Time required

```
SPEED 1
```

Landing speed

FUEL 8

SCORE 220

Crash landing

CRASH

GAME OVER

When height falls below 10 km

) ÷

↑ Moon's surface 1 space = 1km ↑ Spaceship

When fuel is used up

EMPTY

h 150 s 50 f 0

↑ no fuel

CRASH

Crash-landing

GAME OVER

HOW TO PLAY (EXAMPLE)

Start by pressing S P0

1 MOON–LANDING

EXE Starting data

2 h 100 s 20 f 180

↑ Starting height ↑ Speed at game start ↑ Fuel at game start

EXE

3) .

↑ Moon's surface ↑ Spaceship

EXE

4 ROCKET?

Input fuel quantity

1 0 EXE (10)

5 h 82 s 17 f 170

EXE

6) .

Repeat 3–6

20 CRASH

Crash landing

21 GAME OVER

- **Variables**

Name of variable	Description	Name of variable	Description
F	Fuel remaining	T	Time
H	Spaceship position	V	Speed
K	Score	W	Quantity of fuel burned
Q	Speed	Z	Spaceship position
S	Moon's gravity		

PROGRAM

```
MODE 1
CLEAR A
S DEFM
```

```
10 PRINT "MOON-LAN
   DING":SET F0:T=
   0:V=20:F=180:H=
   100:S=2
20 GOSUB 200
30 INPUT "ROCKET",
   W:IF W*(W-40)>0
    THEN 30
40 IF W>F;W=F
50 Q=V-W/2+2:F=F-W
   :H=H-(V+Q)/2
60 IF H<0 THEN 110
70 T=T+1:V=Q
80 IF F>0 THEN 20
90 IF W<>0;PRINT "E
   MPTY":W=0
100 GOSUB 200:GOTO
   50
110 H=H+(V-Q)/2:IF
   H<0;H=0
120 IF W>4;R=W/V:GO
   TO 140
130 R=(-V+SQR (V*V+
   H*(4-W)))/(2-W/
   2)
140 Q=V+(2-W/2)*R:I
   F ABS Q<S THEN
   160
150 PRINT "CRASH","
   GAME OVER":END
160 PRINT "LANDING"
   :K=100*F-INT (T
   +R)*10
170 PRINT "TIME";T+
   R,"SPEED";V,"FU
   EL";F,"SCORE";K
   :END
200 PRINT CSR 8;F;C
   SR 8;"f";CSR 5;
   V;CSR 4;"s";CSR
    8;H;CSR 8;"h"
210 C=INT H:0$="÷":
   IF C>10;C=INT (
   H/10)+1:0$=".":
   IF C>11;C=11
220 PRINT CSR 8;")"
   ;CSR C;0$
230 RETURN
```

```
VAC
MODE
```

Total 497 steps

ALPHABET ATTACK

In this game, you attack each letter that appears on the display in quick succession by pressing the corresponding key within the time limit (countdown). When your attack is successful, the letter disappears; if you fail, the letter remains. Before starting, you choose the desired game level and speed. Whatever speed you choose, however, it will accelerate as the game progresses. You can consider yourself a wizard at alphabet key operation once you've obtained a high score in a game played at the highest level and fastest speed.

RULES AND KEY OPERATION

First, select your game level. There are 3 levels, depending on how many alphabet keys are to be used in the game:

Level 1: Bottom row of alphabet keys, Z–M (10 keys)
Level 2: Bottom and middle rows, Z–M and A–L (19 keys)
Level 3: Bottom, middle and top rows, Z–M, A–L and Q–P (29 keys)

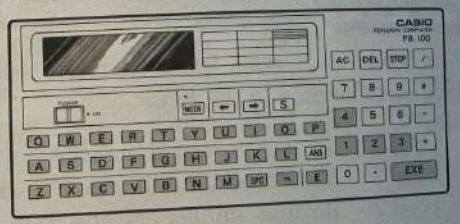
Next, select your game speed. There are 4 speeds, depending on how fast you have to attack each letter. For example, if you choose speed 1 at the start of the game, you have 46 counts in which to attack each of the 6 letters in Line 1. If you choose speed 4, your time limit at the start of the game is 22 counts. When the count is up, you lose your chance to attack. Remember, whatever speed you choose, it will accelerate as the game progresses. The smallest number of counts is 8.

Once you've selected a game level and speed, you're ready to start. When a letter appears on the display, attack it by pressing the corresponding key. If the attack is successful, the letter turns to + and you score one or more points, depending on the game level and speed. If you press the wrong key or fail to press the right key within the time limit, the letter remains and you've lost your chance of attacking that letter. You then go on to attack the next letter that appears on the display.

There are 6 letters to attack in a line, and when you've finished attacking all 6 letters, you move on to the next line – unless your attacks on all 6 letters failed. When this happens, the game is over.
The higher the game level and speed, the higher your score. In addition, each time you successfully attack all 6 letters of a line, you score the number of points arrived at by multiplying the line number times 10 points, and if you successfully attack =, you score 50 additional points.

The keys used are [1], [2], [3], [4], [EXE], [SPC], [=], [E] (exponent key) and the alphabet keys.

[1], [2], [3] and [4] are used to specify the game level and speed. Press [EXE] after pressing one of these keys.
[SPC], [=], [E] and the alphabet keys are used to attack the letters (or characters). Press [SPC] when no letter is indicated on the display.



DISPLAY

- **Starting display**

First, game level/speed input await condition is displayed. After you select the game level and speed, the highest score so far is displayed.
Then the game display appears.

```
LEVEL: 1,2,3  ?
```
⇩ Waiting for game level input

```
SPEED: 1,2,3,4  ?
```
⇩ Waiting for game speed input

```
HI-SCORE = 0
```
↑ Highest so far

- **Play display**

```
------        1  0
```
Not yet indicated ↑ Line ↑ Score

```
++B +S *      1  0
```
↑ Successful attack ↑ Prepare to attack / Failed attack

Score display when a line is completed.

```
LINE = 1:SCORE = 17
```
Score display for Line 1

- **End of game**

```
GAME OVER!
```
⇩

```
LINE = 5:SCORE = 180
```
↑ Line number at game end ↑ Final score

HOW TO PLAY (EXAMPLE)

Start by pressing S P0

1 `LEVEL:1,2,3?`

Press 1 ↵ (Level 1 selected)

2 `SPEED:1,2,3,`

Display moves leftward

3 `EED:1,2,3,4?`

Press 2 ↵ (Speed 2 selected)

4 `HI-SCORE=30`

Display moves leftward

5 `I-SCORE=305`

6 `------ 1 0`

↑ Level ↑ Score

7 `*----- 1 0`

Attack quickly and accurately

8 `Z----- 1 0`

Press Z to attack

9 `+----- 1 0`

↑

The attack was successful

10 `+*---- 1 0`

The faster you attack, the higher your score

11 `+=---- 1 0`

Press = to attack

12 `++---- 1 0`

13 `++*--- 1 0`

14 `++W--- 1 0`

Press V to attack

15 `++W--- 1 0`

↑

The attack failed

16 `++W*-- 1 0`

17 `++WZ-- 1 0`

26 `++W+++ 1 0`

27 `LINE=1:SCOR`

Display moves leftward.

28 `1:SCORE=60`

29 `------ 2 60`

30 `*----- 2 60`

31 `V----- 2 60`

Press V to attack

32 `+----- 2 60`

33 `+*---- 2 60`

34 `+=---- 2 60`

Press = to attack

35 `++---- 2 60`

60 `XY=WZV 6 185`

61 `GAME OVER!`

62 `LINE=6:SCOR`

63 `=6:SCORE=185`

- **Variables**

Name of variable	Description	Name of variable	Description
A	Speed (loop) variable	P	Number of lines
C$	Key input	S	Game speed
D$	Character selection	X	Number of successful attacks in a line
I	Loop		
J	Loop	Y	Score
K	Game level	Z	Highest score so far
L	Beginning of character string	$	QWE . . . .= E
M	Conversion value	N	Loop

PROGRAM

MODE 1

```
CLEAR A
S DEFM
```

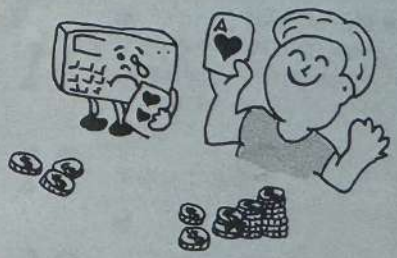
```
10 $="QWERTYUIOPAS
   DFGHJKL=ZXCVBNM
   =E":Y=0:P=1
20 INPUT "LEVEL:1,
   2,3",K:L=(3-K)*
   10+1:M=99/(30-L
   )
30 INPUT "SPEED:1,
   2,3,4",S:FOR N=
   1 TO 40:NEXT N
40 PRINT "HI-SCORE
   =";Z;
50 FOR N=1 TO 100:
   NEXT N:X=0
60 A=(6-S)*4/P+3:I
   F A<4:A=4
70 PRINT :PRINT CS
   R 0;"------";P;
   Y;
80 FOR I=0 TO 5
90 PRINT CSR I;"*"
   ;:FOR N=1 TO 50
   :NEXT N
100 D$=MID$(INT (INT
    (RAN#*100)/M)+
    L,1)
110 PRINT CSR I;D$;
120 FOR J=0 TO A*2
130 C$=KEY
140 IF C$<>D$ THEN 1
    80
150 IF C$=D$:PRINT
    CSR I;"+";:X=X+
    1:Y=Y+K*(6-INT
    (A/4))
160 IF D$="=":Y=Y+5
    0
170 GOTO 190
180 NEXT J
190 NEXT I
200 IF X=6:Y=Y+10*P
210 PRINT "";:P=P+1
220 IF X=0:PRINT :P
    RINT "GAME OVER
    !";
225 FOR N=1 TO 100:
    NEXT N
230 PRINT :PRINT "L
    INE=";P-1;":SCO
    RE=";Y;
240 IF X<>0 THEN 50
250 IF Z<Y:Z=Y
260 STOP :GOTO 10
```

```
VAC
MODE
```

Total 502 steps

CHANCE IT

This is a very simple game of chance. You and the computer each draw a card, and the one who draws the larger number wins the round. Since you must place a bet before you draw your card, the chance factor is greater. Try to guess when you'll draw a high card and bet accordingly.



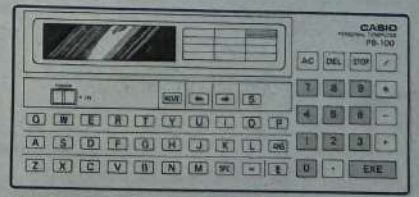
RULES AND KEY OPERATION

The game starts with the dealer (computer) holding 300 chips and you, 200. Between 10 and 50 chips may be staked in a single bet. After placing a bet, the dealer and you each draw a card, and – naturally – the one with the larger number wins. The loser must hand over the number of chips staked to the winner. When an Ace is drawn – !!! – the winner receives twice the number of chips staked in the bet. If both players draw the same number (a tie), the chips go to the winner of the next round. The game ends when 25 rounds are concluded or when either player loses all his chips.

* Card Designation:
T = 10 J = Jack (11) Q = Queen (12) K = King (13) A = Ace (1)

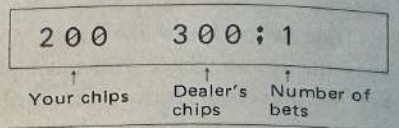
The keys used are the numerical keys 0 – 9 and EXE .

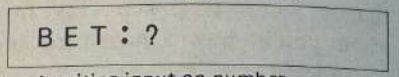
The numerical keys are used to input the number of chips staked. After this input, EXE is pressed.

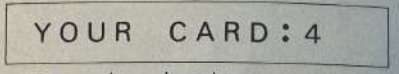


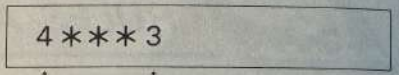
DISPLAY

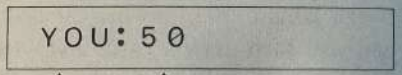
● Play display
Play starts as soon as you start the program.



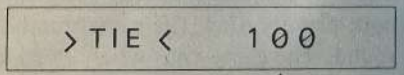






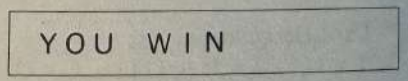


When it's a tie

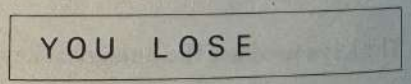


● End of game

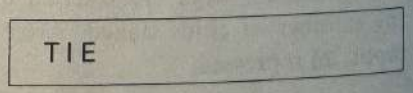
If you win



If you lose

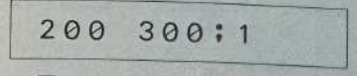


If it's a tie

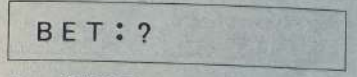


HOW TO PLAY (EXAMPLE)

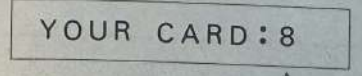
Start by pressing [S] [P0]

1 

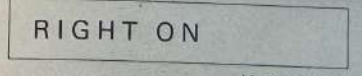
[EXE]

2 

Press [5] [0] [EXE] Stake 50 chips

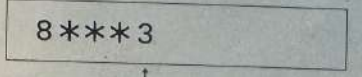
3 

[EXE]

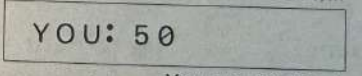
4 

Here we go!

[EXE]

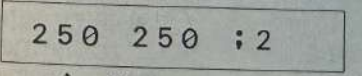
5 

[EXE] You win

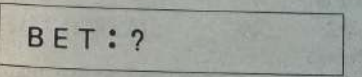
6 

You won 50 chips!

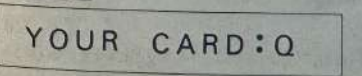
[EXE]

7 

[EXE]

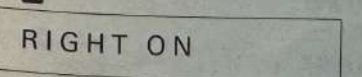
8 

[3] [0] [EXE]

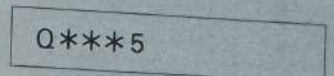
9 

Pretty good

[EXE]

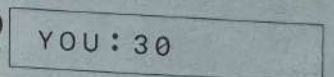
10 

[EXE]

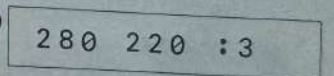
11 

Good!

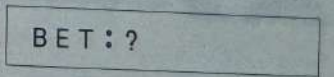
[EXE]

12 

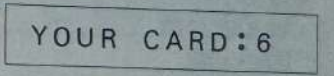
[EXE]

13 

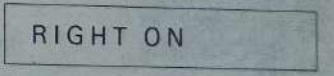
[EXE]

14 

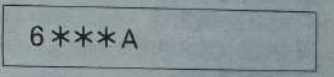
[5] [0] [EXE]

15 

[EXE]

16 

[EXE]

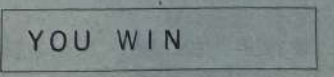
17 

Oh no!

[EXE]

18 

Twice the number of chips staked go to the dealer

52 

- **Variables**

Name of variable	Description	Name of variable	Description
N	Number of bets	V	Number of chips to go to next bet when a tie
Y	Your chips	W	Number of chips to go to next bet when a tie
Z	Dealer's chips	$	Card
S	Loop counter	P	Your card
W	Loop counter	Q	Dealer's card
X	Number of chips staked	O	Random integer number
TS	Your card	$	Types of cards (2,3,4 ..... 9, T, J, Q, K, A)
US	Dealer's card		

PROGRAM

MODE 1

CLEAR A EXE

S DEFM ⊘ EXE

```
10 'AC
15 N=1:Y=200:Z=300
   :$="23456789TJQ
   KA"
20 FOR S=0 TO 12:A
   (S)=S+1:NEXT S
30 GOSUB 220:P=0:T
   $=MID(0,1)
40 GOSUB 220:Q=0:U
   $=MID(0,1)
50 PRINT CSR 9;N;C
   SR 9;":";CSR 5;
   Z;CSR 0;Y
60 PRINT :INPUT "B
   ET:",X:IF X>Y T
   HEN 60
70 IF X<10:X=10
80 IF X>50:X=50
90 PRINT "YOUR CAR
   D:";T$:PRINT "R
   IGHT ON"
100 PRINT T$;"***";
    U$
110 IF P*13 THEN 14
    0
120 IF Q=13:V=X*2:G
    OTO 170
130 X=X*2:GOTO 150
140 IF Q=13:X=X*2
150 IF P>Q:Y=Y+W+X:
    Z=Z-X:PRINT "YO
    U:";W+X:W=0:GOT
    O 180
160 IF P<Q:Z=Z+W+X:
    Y=Y-X:PRINT "DE
    ALER:";W+X:W=0:
    GOTO 180
170 W=W+V+X*2:V=0:Y
    =Y-X:Z=Z-X:PRIN
    T ")TIE(";W
180 IF Z>0:IF Y>0:I
    F N<25:N=N+1:GO
    TO 30
190 IF Y>Z:PRINT "Y
    OU WIN":END
200 IF Y<Z:PRINT "Y
    OU LOSE":END
210 PRINT "TIE":END

220 R=INT (RAN#*13)
    :IF FRAC A(R)=.
    4 THEN 220
230 A(R)=A(R)+.1:O=
    INT A(R):RETURN
```

VAC EXE

MODE ⊘

Total 536 steps

ROYAL SEVENS

You and another player play this game. Taking turns, you each bet 5 times. Your highest objective in this game is to hit 777. When you do, you win 500 times the number of chips you bet. Also, though less exciting, if you hit a number with two sevens – 771 or 277, for example – you win 10 or 20 times more than you bet. When you and your opponent complete 5 turns, compare your chips to determine the winner of the game.

RULES AND KEY OPERATION

Both players start with a credit of 50 chips. First, input the number of chips you want to stake in the bet. You cannot stake more than what you have, so at the start of the game, the maximum you can stake is 50 chips. Next, press key EXE. The computer then flashes 3-digit numbers one after another on the display. Stop a number by pressing key ⊘. To win your bet, you must stop a number with the same pattern as one of those shown on the next page (table). Depending on the pattern, you can win many times more than the number of chips you staked. For example, when you stop 777, you win 500 times the number of chips you staked. If the number you stop does not correspond to any of the patterns given, you lose the chips staked in the bet.

After you and your opponent repeat the above play 5 times, compare your chips and determine the winner. If one player loses all his chips before the other completes 5 turns, input "0" after each "BET?" display and continue to take your turn.

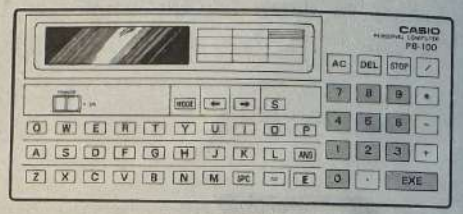
PATTERNS

7 7 7	500 times	○ ○ ○	50 times
7 7 ∗	20	○ ○ ∗	5
∗ 7 7	10	∗ ○ ○	3
7 ∗ 7	5	○ ∗ ○	2
7 ∗ ∗	2		
∗ 7 ∗	1		
∗ ∗ 7	1		

(∗ ..... Any number
○ ..... Same number)

The keys used are the numerical keys [0] – [9] and [EXE].

The numerical keys are used to input your bets, and [EXE] is pressed after each such input. Press [0] to stop a number.



DISPLAY

- **First player's start**

```
#-1.1
```
1st player ↑ Bet number

```
CREDIT 50
```
No. of chips held by 1st player

```
BET?
```
Input No. of chips you want to stake in the bet. Must be within your credit.

- **Second player's start**

```
#-2.1
```
2nd player ↑ Bet number

```
CREDIT 50
```
No. of chips held by 2nd player

```
BET?
```
Input No. of chips 2nd player wants to stake in the bet. Must be within his credit.

- **End of game**

```
#-1 WINS
```
If 1st player wins

```
100/50
```
1st player's credit after 5 bets ↑ 2nd player's credit after 5 bets

```
TIE 50/50
```
When both players end with the same credit

HOW TO PLAY (EXAMPLE)

Start by pressing [S] [P0]

1
```
#-1.1
```
↑1st player ↑1st bet

[EXE]

2
```
CREDIT 50
```
Initial credit 50 chips

[EXE]

3
```
BET ?
```
Awaiting input (Must be within your credit)

Press [1] [0] [EXE] (1st player bets 10 chips)

4
```
368
```

3-digit numbers flash one after another on the display

12
```
741
```

Press ⊡
Try to stop a number with a 7 in it

13 741

↑ A number is stopped

EXE

14 ◆CHIPS◆ / 20

↑ 2 times the number of chips you bet

EXE

15 #−2.1

↑ 2nd player ↑ 1st bet

EXE

16 CREDIT 50

EXE

17 BET?

Press 2 ⊡ EXE (2nd player bets 20 chips)

18 523

3-digit numbers flash one after another on the display

29 234

⊡

30 234

EXE Too bad, it doesn't match any of the patterns

31 ◆CHIPS◆ −20

↑

EXE 20 chips are subtracted from 2nd player's credit

32 #−1.2

↑ 2nd bet

Both players bet 5 times

EXE

43 #−1 WINS

1st player wins

EXE

44 140 / 100

- Variables

Name of variable	Description	Name of variable	Description
A(0)	1st player's credit	P	Number in hundreds place
A(1)	2nd player's credit	Q	Number in tens place
D	Credit working area	R	Number in ones place
I	Loop counter	S	3-digit number
W	Loop counter	X	Number of chips bet
N	Number of operations	Z$	Key input data

PROGRAM

MODE 1

```
CLEAR A EXE
S DEFM ⊡ EXE

10 A(0)=50:A(1)=50
   :FOR N=1 TO 5:F
   OR L=0 TO 1
20 PRINT "#";(L+1+
   (N/10))*(-1)
30 PRINT "CREDIT";
   A(L)
40 INPUT "BET",X:I
   F X>A(L) THEN 4
   0
50 GOSUB 200:X=X*T
   :A(L)=A(L)+X:PR
   INT CSR 0;"◆CHI
   PS◆";X
60 NEXT L:NEXT N
70 IF A(0)>A(1);PR
   INT "#-1 WINS":
   GOTO 100
80 IF A(0)<A(1);PR
   INT "#-2 WINS":
   GOTO 100
90 PRINT "TIE";
100 PRINT A(0);" /"
   ;A(1):END
200 S=INT (RAN#*100
   0):IF S<111 THE
   N 200
210 PRINT CSR 0;S;
220 Z$=KEY:IF Z$="0
   " THEN 240
230 GOTO 200
240 STOP :P=INT (S/
   100):Q=INT ((S-
   100*P)/10):R=S-
   P*100-Q*10
250 IF P=Q;IF P=R;I
   F Q=R THEN 270
260 GOTO 290
270 IF P=7;T=500:RE
   TURN
280 T=50:RETURN
290 IF P*Q THEN 320
300 IF P=7;T=20:RET
   URN
310 T=5:RETURN
320 IF Q*R THEN 350
330 IF Q=7;T=10:RET
   URN
340 T=3:RETURN
350 IF P*R THEN 380
360 IF P=7;T=5:RETU
   RN
370 T=2:RETURN
380 IF P=7;T=2:RETU
   RN
390 IF Q=7;T=1:RETU
   RN
400 IF R=7;T=1:RETU
   RN
410 T=-1:RETURN
   VAC EXE
   MODE ⊡
```

Total 540 steps

270,000

WAR OF NUMBERS

Pit your strength against the computer to win this war of numbers. To win each encounter, you must input a number larger than the one held by the computer by guessing correctly from the clue given. But even if you keep winning, don't slacken your guard – there's a decisive battle to be fought at the very end, and if you lose this battle, you have to start all over again.

RULES AND KEY OPERATION

Your initial score is 5,000. You and the computer have several encounters until your score reaches either 0 or 10,000. To win each encounter, you must input a number greater than the 3-digit number – which is hidden from you – held by the computer, by guessing the computer's number correctly from the clue given. The clue is the sum of the three digits in the ones, tens and hundreds positions of the number held by the computer. To input your number, you can either choose from among 1,000, 750, 500 and 250, or you can choose any 3-digit number you like.

The number you input to challenge the computer is automatically subtracted from your score. If you lose the encounter (in other words, if your number is smaller than the computer's), you receive no additional points. If you win, you receive twice the number held by the computer. Now, this is trickier than it seems.
You must be careful what number you input – if your number is too large, you'll end up with a smaller score than at the start of the encounter, even if you win the encounter. For example, say you start with a score of 4,000 and you input 1,000. Your score automatically becomes 3,000. Then, if the computer's number is 150, you win 300, making your score 3,300, which is less than what you started with. Therefore, choose your number carefully and try to input a number only slightly greater than the computer's.

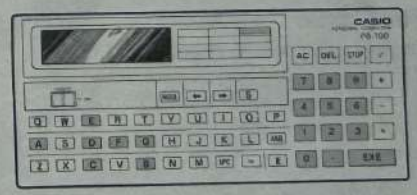
If your score falls to 0, you lose the war and the game ends. If, on the other hand, your score reaches 10,000, you and the computer fight a final decisive battle to determine the winner of this war. This time, the computer displays a number, and you must choose a key from among keys A, B, C, D, E, F and G that holds a number smaller than the one displayed. The numbers held by each key are hidden from you. If you win this battle, the game ends. If you lose, you start all over again.

The keys used are [0] – [9], [.], and [A], [B], [C], [D], [E], [F] and [G].

Key [0] inputs 250
Key [1] inputs 500
Key [2] inputs 750
Key [3] inputs 1,000
Key [.] is pressed to input numbers other than the above. After pressing [.], press the numerical keys that correspond to the number chosen, then press [EXE].

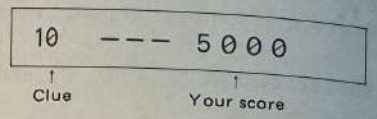
(Example)
Press [.], [1], [2], [5], [EXE], to input 125.

Keys [A] – [G] are used in the final battle. Each key hides a different number. Choose the one which you think hides a number smaller than the one displayed.

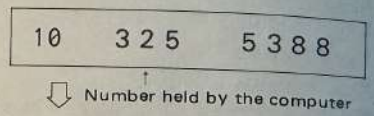


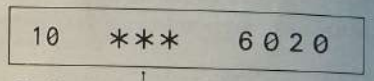
DISPLAY

- **Starting display**
  The game starts as soon as the program is started.

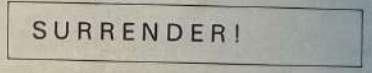


- **Display of the number held by the computer**

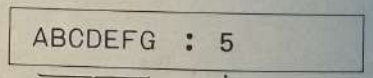


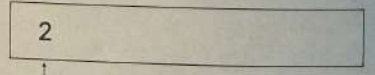


- **If your score reaches 0, the game ends**

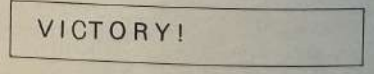


- **Final battle (when your score reaches 10,000)**

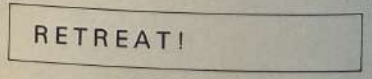




- **When you win the final battle**



- **When you lose the final battle**



HOW TO PLAY (EXAMPLE)

Start by pressing S P0

1 18 --- 5000

Press 1 (input 500)

2 18 954 4500

Number held by the computer (You lose)

3 10 --- 4500

Press 2 (input 750)

4 10 802 3750

Number held by the computer (You lose)

5 16 --- 3750

Press 3 (input 1,000)

6 16 493 3736

Your score = 3750 - 1000 + (493x2)

7 16 *** 3736

Don't press 3 too many times

8 13 --- 3736

Press .

9 13 --- 3736?

3 6 7 ENT

10 13 733 3379

19 15 --- 9865

Press 2

20 15 555 10225

21 15 *** 10225

22 ABCDEFG: 6

Press A

23 9

24 RETREAT!

32 19---808

Press .

33 19---808?

8 0 9 ENT

34 SURRENDER!

- Variables

Name of variable	Description	Name of variable	Description
C	Number in tens position of D	W	Number held by the computer
D	Numbers designated by A-G	X	Number in hundreds position
M	Number in ones position of D	Y	Number in tens position
N	Number input	Z	Number in ones position
P	Number of encounters (source of random number)	AS	Key input data (0, 1~3, ·)
T	Sum of numbers in ones, tens and hundreds positions	Q	Loop counter
U	Score	BS	Key input data (A-G)
V	Number displayed when score reaches 10,000		

PROGRAM

MODE 1

CLEAR A EXE

S DEFM ( ) EXE

```
10 VAC
15 U=5000
20 X=INT (RAN#*10)
   :Y=INT (RAN#*10
   ):Z=INT (RAN#*1
   0):T=X+Y+Z
30 IF X≠Y:IF X≠Z:I
   F Y≠Z THEN 50
40 GOTO 20
50 W=X*100+Y*10+Z:
   PRINT CSR 2;" "
   ;CSR 0;T;CSR 4;
   "---";U;
60 A$=KEY:IF A$≠"0
   ";IF A$≠"1";IF
   A$≠"2":IF A$≠"3
   ";IF A$≠".." THE
   N 60
70 IF A$=".";INPUT
   N:GOTO 90
80 N=250*(VAL(A$)+
   1)
81 U=U-N:PRINT CSR
   7;U;
82 IF U<1E3:PRINT
   " ";
90 PRINT CSR 5;Z;C
   SR 4;Y;CSR 3;X;
   CSR 0;T::GOSUB
   900
100 IF W>N THEN 120
110 PRINT CSR 4;"**
   *";:GOSUB 900:U
   =U+W*2:IF U≥100
   00 THEN 200
120 IF U≤0:PRINT :P
   RINT CSR 0;"SUR
   RENDER!":END
130 GOTO 20
200 V=INT (RAN#*10)
   :$="ABCDEFG :":
   PRINT CSR 0;" "
   ;$;V;
210 B$=KEY:IF B$=""
    THEN 210
220 FOR I=1 TO 7:IF
    B$=MID$(I,1) TH
   EN 240
230 NEXT I
240 D=INT (RAN#*10)
   +I:IF D>9:D=D-9
250 PRINT CSR I-1;D
   ;:GOSUB 900
260 IF D>V;PRINT CS
   R 0;"RETREAT! "
   ;:GOSUB 900:GOT
   O 10
270 PRINT :PRINT CS
   R 0;"VICTORY!":
   END
900 FOR Q=1 TO 50:N
   EXT Q:RETURN
```

VAC EXE

MODE ( )

Total 543 steps

DRAGON-SLAYING

In this game, a battalion of knights takes on a force of fierce dragons. No matter how many dragons your knights slay, the dragons just keep coming and coming. You begin with three knights, and the game is over when all your knights are wounded. To achieve a high point score, you have to slay as many dragons in as few moves as possible.

RULES AND KEY OPERATION

The battles between dragons and knights are conducted on a one-to-one basis. Move your knight left or right either to chase or escape from the dragon. The dragon moves automatically to chase after the knight. The two take turns moving. When your knight catches up with the dragon in a move, the dragon is slain and you score a point. On the other hand, if the dragon catches up with your knight in a move, the knight is wounded and taken off the play. Both dragons and knights move 0 – 5 spaces at random. When you score 2,000 points, a new knight will be added to the play. The fewer the number of moves it takes your knight to slay a dragon, the higher your point score for that battle. When all your knights are wounded, or when 7 moves fail to conclude a match, the game is over. If you score the highest number of points won so far, you may enter your name in the computer, using up to 7 letters.

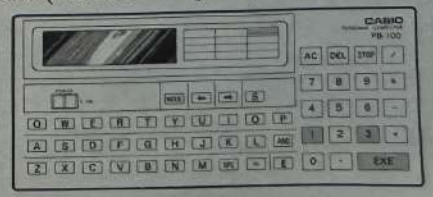
The keys used are: 1 and 3

These are used to move a knight.

Key 1 is for moving the knight leftward (within 0 – 5 spaces).

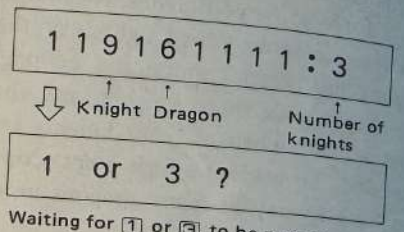
Key 3 is for moving the knight rightward (within 0 – 5 spaces).

When it's the knight's turn to move, the display will indicate, "1 or 3? ". Press either 1 or 3, and then press EXE key.

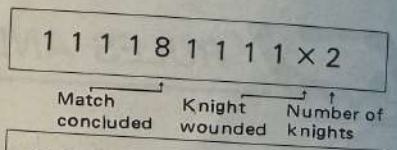


DISPLAY

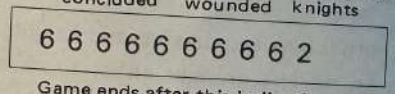
- **Starting display**
  (The game begins immediately when the [S] and [P0] keys are pressed)



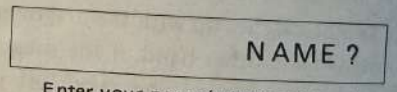
- **When a match is concluded**
  (score or one less knight)

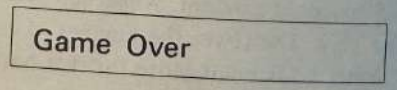


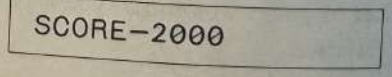
- **When a match is not concluded after 7 moves**

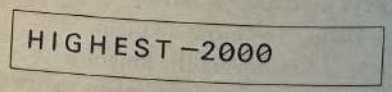


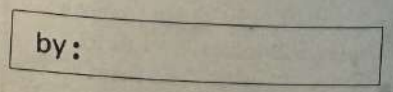
- **If you score the highest number of points**





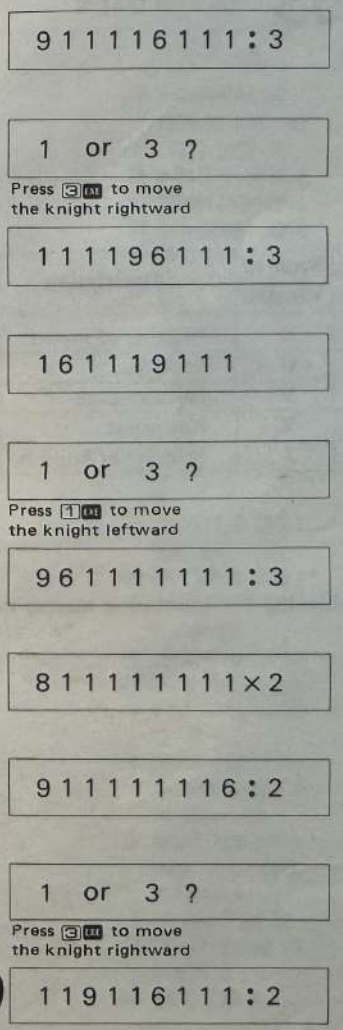






HOW TO PLAY (EXAMPLE)

Start by pressing [S] [P0]



```
49 NAME?
   J O H N EXE

50 Game Over

51 SCORE-2500

52 HIGHEST-2500

53 by: JOHN
```

- Variables

Name of Variable	Description	Name of Variable	Description
G	Position of dragon	N	Number of inputs
H	Position of knight	U	Score
I	Game display	V	Highest score
J	New position after a move	X	Key input
K	1111111111	Z	Number of knights

NOTE

In the program, allow 12 spaces within the " " (quotation marks) in lines 50 and 140.

PROGRAM

```
MODE 1
CLEAR A EXE
S DEFM ( ) EXE
10 U=0:P=0:K=INT (
   10↑9/9):FOR Z=3
    TO 1 STEP -1
20 N=0:IF P=0:IF U
   >2000:Z=Z+1:P=1
30 S=9:GOSUB 180:G
   =T-1
40 Q$=":":GOSUB 18
   0:H=T-1:IF G=H
   THEN 40
50 N=N+1:GOSUB 200
   :PRINT CSR 0;"
                ";CS
   R 0;
60 INPUT "1 or 3";
   X
70 IF X*1;IF X*3 T
   HEN 50
80 S=6:GOSUB 180:J
   =H-(X-2)*T:GOSU
   B 220:H=J
90 IF G=H;Q$="O":G
   OSUB 190:U=U+10
   *INT (100/N):GO
   TO 20
100 GOSUB 200:IF N>
   6;I=6*K:GOSUB 2
   10:GOTO 150
110 GOSUB 180:J=G+T
   *SGN (H-G):GOSU
   B 220:G=J
120 IF G*H THEN 50
130 Q$="X":GOSUB 19
   0:NEXT Z
140 PRINT CSR 0;"
                ";
150 IF U>V;V=U:INPU
   T "NAME",B$
160 PRINT "Game Ove
   r":PRINT "SCORE
   ";-U:PRINT "HIG
   HEST";-V
170 PRINT " by:";B$
   :GOTO 10
180 T=INT (RAN#*S)+
   1:RETURN
190 I=K+10↑H*7:GOTO
    210
200 I=K+10↑G*5+10↑H
   *8
210 PRINT CSR 10;Z;
   CSR 10;Q$;CSR 0
   ;I;:FOR W=1 TO
   300:NEXT W:RETU
   RN
220 IF J<0;J=0
230 IF J>8;J=8
240 RETURN

   VAC EXE
   MODE
   Total 543 steps
```

POISON BERRIES

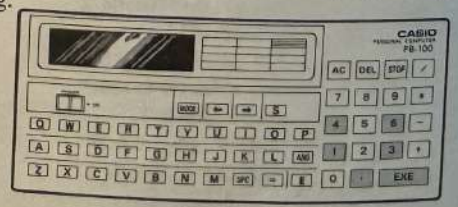
Bozo, the bear, goes around eating as many berries as possible, trying to avoid those that are poison. If Bozo fails to eat a berry before he opens his mouth 8 times, the berry changes positions. It's up to you to move Bozo quickly enough so he won't go hungry. The game ends either when Bozo fails to eat a berry within the established limit or when he eats 3 poison berries.

RULES AND KEY OPERATION

When Bozo succeeds in eating a berry, you score a point. If Bozo fails to eat a berry before he opens his mouth 7 times, the position of the berry changes. A berry can change positions 8 times at the start of the game, but one less after each time Bozo eats a berry. The game ends if Bozo fails to eat a berry within the time that the berry changes positions the allowed number of times. In addition, a berry may suddenly turn to poison when Bozo opens his mouth for the 5th time. Each time a berry turns to poison, the number of times that a berry may change positions increases by one, but this allowance is limited to 8 times at the most. The game also ends if Bozo eats 3 poison berries.

The keys used are [1], [4], [.], [3] and [6]. These are used to move Bozo.

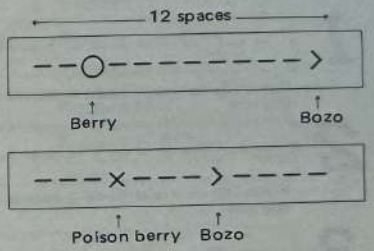
Key [4] is for moving Bozo 2 spaces to the left.
Key [1] is for moving Bozo 1 space to the left.
Key [.] is for moving Bozo all the way to the right or left in one bound, depending on which direction Bozo is facing.
Key [3] is for moving Bozo 1 space to the right.
Key [6] is for moving Bozo 2 spaces to the right.
Using the above keys, you move Bozo to score as many points as possible.



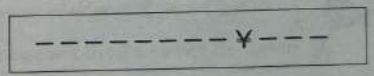
DISPLAY

- **Play display**

The game begins immediately when the [S] and [Pθ] keys are pressed.

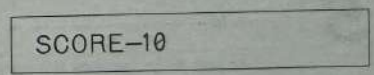


- **When Bozo eats a berry or a poison berry**



- **End of game**

The game ends when Bozo eats 3 poison berries or when he fails to eat a berry within the limit given.

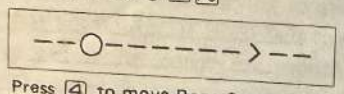


Note
The berries appear on the display at least 8 spaces away from Bozo. Try to guess when the next berry will appear and move Bozo accordingly, so he can eat the berry faster.

HOW TO PLAY (EXAMPLE)

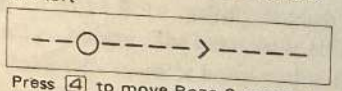
Start by pressing [S] [P0]

1



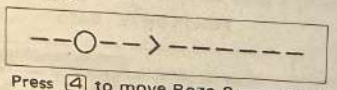
Press [4] to move Bozo 2 spaces to the left

2



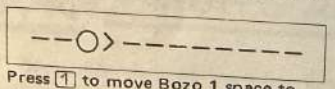
Press [4] to move Bozo 2 spaces to the left

3



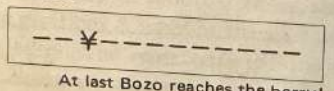
Press [4] to move Bozo 2 spaces to the left

4



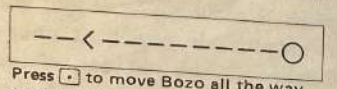
Press [1] to move Bozo 1 space to the left

5



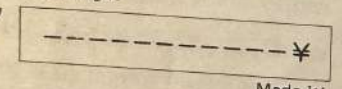
At last Bozo reaches the berry!

6



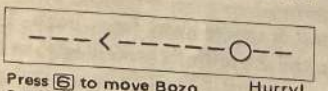
Press [.] to move Bozo all the way to the right

7



Made it!

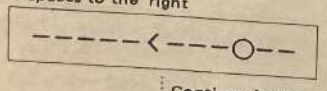
8



Press [5] to move Bozo 2 spaces to the right

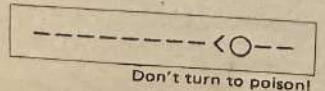
Hurry!

9



Continue in the same way

11

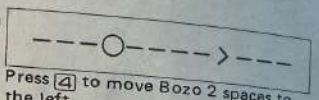


Don't turn to poison!

Press [3] to move Bozo 1 space to the right

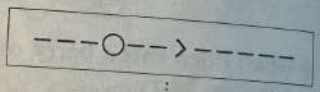
12

13



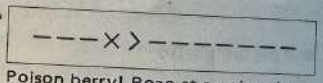
Press [4] to move Bozo 2 spaces to the left

14



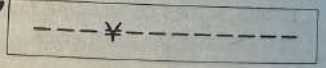
Continue in the same way

16

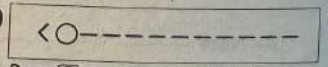


Poison berry! Bozo at a poison berry
Press [1] to move Bozo to the left

17

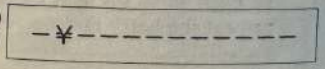


32



Press [3] to move Bozo 1 space to the right

33



34

Game Over

[EXE]

35

SCORE−14

Score display (Game over)

- Variables

Name of variable	Description	Name of variable	Description
A	Number of tries allowed	K	RAN#
C$	Key input	M	X or not X
D$	>	N	When to indicate X
E$	< Bozo	O	Number of Xs
F$	> or <	P	Position of ○
I	Number of tries	Q	Position of Bozo
J	Loop	S	Score

PROGRAM

[MODE] [1]

CLEAR A [EXE]

[S] [OPTN] [2] [EXE]

```
10 D$=">":E$="<":F
   $=D$:Q=11:S=0:A
   =9:O=0
20 A=A-1:M=0:IF A<
   1:A=1
30 PRINT CSR 0;"--
   ----------";CSR
    Q;F$;:I=0
40 I=I+1:K=INT (RA
   N#*100)/(A+2):I
   F K=INT K=0:M=1
   0:I=I-1
50 P=INT (RAN#*12)
   :IF ABS (P-Q)≤5
    THEN 50
60 FOR J=1 TO 7:N=
   M+J:IF N≥15:PRI
   NT CSR P;"x";:A
   =A+1:GOTO 80
70 PRINT CSR P;"o"
   ;:GOTO 90
80 IF A>8:A=8
90 C$=KEY:PRINT CS
   R Q;" ";:IF C$=
   "." THEN 120
100 IF F$=D$:Q=-1:G
   OTO 120
110 Q=12
120 IF C$="1":Q=Q-1
   :F$=D$
130 IF C$="4":Q=Q-2
   :F$=D$
140 IF C$="3":Q=Q+1
   :F$=E$
150 IF C$="6":Q=Q+2
   :F$=E$
160 IF Q<0:Q=0:F$=E
   $
170 IF Q>11:Q=11:F$
   =D$
180 PRINT CSR Q;F$;
   :IF Q=P THEN 22
   0
190 PRINT CSR Q;"*"
   ;:FOR W=1 TO 10
   0:NEXT W
200 IF N≥15:O=O+1:S
   =S-1:IF O=3 THE
   N 230
210 S=S+1:GOTO 20
220 NEXT J:PRINT CS
   R P;"-";:IF I=A
    THEN 40
230 PRINT :PRINT "G
   ame Over":PRINT
    "SCORE";-S:GOT
   O 10
```

VAC [EXE]

[MODE] [2]

Total 543 steps

BASEBALL GAME

(Memory Pack Required with PB-100 only)

It's your team's turn at bat. The computer's the pitcher. This pitcher throws only straight balls, so all you have to do is get the timing right and wham that ball. See how many points you can run up in this inning.

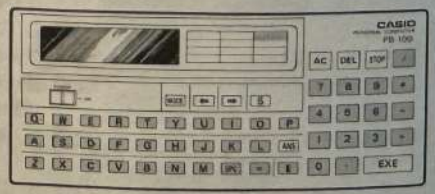
RULES AND KEY OPERATION

Each game constitutes an inning. When the program is started, a ball (*) is pitched to the 1st batter at home base (H). Press a key (any key you like) just when the ball reaches home base. If your timing is right, the batter hits the ball. The computer tells you whether it's a one-, two- or three-base hit, homerun, foul, or out. If your timing is wrong, the batter misses completely, and it's 1 strike. After 3 strikes, the next batter comes to bat. Whenever there's a hit, the runner (↑) is displayed moving from home base to the appropriate base. If there's another runner before him, both are shown. When a runner reaches home base, your team scores a point. The game ends with 3 outs, and the team's score is displayed.

You can use any key you like to hit the ball, except the following:

S DEL AC STOP

ANS ⇦ ⇨ EXE MODE



DISPLAY

• **Starting display**

```
        *   H
        ↑   ↑
      Ball  Home base
```

• **Play display**

```
1 – S
↑
Number of strikes
⇩
1 – 0
↑
Number of outs
⇩
FOUL
⇩
1 – BASE HIT
⇩
OUT
⇩
2 – BASE HIT
⇩
3 – BASE HIT
⇩
HOMERUN
```

• **End of game**

```
GAME OVER

SCORE 3
↑
Final score
```

HOW TO PLAY (EXAMPLE)

Start by pressing [S] [P0]

1
```
*    H
```
Press [=] (any key) as soon as the ball reaches home base

2
```
1- BASE HIT
```
[EXE]

3
```
↑·····↑
```
Runner moves from home base to 1st base (right to left)

[EXE]

4
```
SCORE 0
```
[EXE]

5
```
*    H
```
[SPC]

6
```
1-S
```
[EXE]

7
```
*    H
```

9
```
1-O
```

[EXE]

10
```
*    H
```
[=]

11
```
HOMERUN
```

12
```
↑·····↑
```
Runner runs around the bases and returns home base (right to left)

13
```
SCORE 2
```

14
```
*    H
```

23
```
OUT
```

24
```
3-O
```

25
```
GAME OVER
```

26
```
SCORE 2
```

- **Variables**

Name of variable	Description	Name of variable	Description
E	Number of outs	U	Ball position (loop counter)
F	Number of strikes	V	Random number for deciding ball position
I	Ball position (loop counter)	Z	Type of ball
J	Loop counter	A(1)	1st base 1=runner at 1st base 0=no runner
N	Runner position	A(2)	2nd base 1=runner at 2nd base 0=no runner
P	Score	A(3)	3rd base 1=runner at 3rd base 0=no runner
T	Number of base hits	A(0)	Home base 1=runner at home base 0=no runner

PROGRAM

[MODE] [1]

CLEAR A[EXE]

[S] [DEFM] [↵] [EXE]

```
10 VAC
15 $=" ↑"
20 FOR I=0 TO 11 S
   TEP RAN#+.3
30 PRINT CSR I;"*"
   ;CSR 10;"H";:FO
   R G=1 TO 5:NEXT
    G
35 PRINT CSR X;" "
   ;:X=I:K$=KEY:IF
    K$="":NEXT I
40 IF INT I=10 THE
   N 90
50 F=F+1:PRINT CSR
    0;F;"-S";
55 FOR G=1 TO 50:N
   EXT G:PRINT :ST
   OP
60 IF F=3:E=E+1:PR
   INT E;"-O":F=0
70 IF E=3:PRINT "G
   AME OVER":PRINT
    "SCORE";P:END
80 GOTO 20
90 FOR I=10 TO 0 S
   TEP -1:PRINT CS
   R I;"*";
95 FOR G=1 TO 5:NE
   XT G:PRINT CSR
   X;" ":X=I:NEXT
    I
100 Z=INT (RAN#*100
   ):PRINT CSR 0;
110 IF Z≤19:IF F=2:
   PRINT "FOUL":F=
   1:GOTO 50
115 IF Z≤19:PRINT "
   FOUL":GOTO 50
120 IF Z≤24:PRINT "
   HOMERUN":T=4:GO
   TO 170
130 IF Z≤54:PRINT "
   1-BASE HIT":T=1
   :GOTO 170
140 IF Z≤62:PRINT "
   2-BASE HIT":T=2
   :GOTO 170
150 IF Z≤68:PRINT "
   3-BASE HIT":T=3
   :GOTO 170
160 PRINT "OUT":F=3
   :GOTO 60
170 A(0)=1:FOR I=0
   TO 3
175 PRINT CSR (3-I)
   *3;MID(A(I)+1,1
   );:NEXT I
180 FOR I=1 TO 1:FO
   R J=3 TO 0 STEP
    -1
190 IF J=3:P=P+A(3)
   :GOTO 210
200 A(J+1)=A(J)
210 NEXT J:A(0)=0
220 FOR J=0 TO 3:PR
   INT CSR (3-J)*3
   ;MID(A(J)+1,1);
   :NEXT J
230 NEXT I
240 PRINT CSR 0;"SC
   ORE";P:F=0:GOTO
    20
```

VAC[EXE]

[MODE] [↵]

Total 627 steps

GOLF GAME

(Memory Pack Required with PB-100 only)

Here's a game of desk-top golf which you can enjoy on a rainy day. Taking into account the three factors of distance to the hole, par and wind velocity, determine the strength of your stroke and try to get the ball on the green in as few strokes as possible. If you get it just right, you can even shoot a hole-in-one!

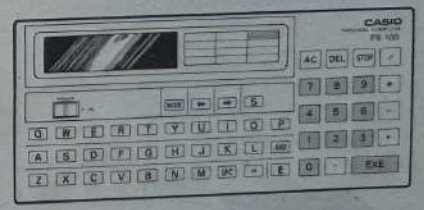
RULES AND KEY OPERATION

First, input the number of holes you want to play. Then, the hole number, distance to the hole, par and wind velocity are displayed. If the wind velocity is a negative number, your ball travels against the wind, and if it's a positive number, with the wind. In addition, the larger the number, the stronger the wind. After confirming these data, input the strength of your stroke, choosing from among 0–60. The computer then displays the remaining distance to the hole. Input your stroke strength again, but this time input a smaller number as you've less distance to go. Continue in the same way. When you come to within 5 meters in front of or behind the pin, the pin and your ball are displayed. Judging the distance between pin and ball, press a key from among keys 1–5 and putt the ball into the hole. Repeat the above steps as many times as the number of holes you input at the start of the game. At the end, total par, your total number of strokes, and your final score are displayed.

Note: The stroke strength (0–60) determines the ball's flight as follows. If the strength (unit m/S at initial velocity) is 60, the ball starts traveling at an initial speed of $60/\sqrt{2}$ (m/S) in both the vertical and horizontal directions. (The acceleration of gravity is 9.8 m/S downwards.) The wind affects the initial speed in the horizontal direction in proportion to its velocity.

The keys used are the numerical keys [0] – [9] (10 keys) and [EXE].

The numerical keys are used to input your stroke strength (0–60). Key [EXE] is pressed after inputting the numbers. Only keys [1] – [5] are pressed when putting.



DISPLAY

- **Starting display**

  Before starting the game, input the number of holes you want to play.

  ```
  Holes ?
  ```
  Awaiting input

- **Play display**

  ```
  Hole No. 1
  ```

  ```
  416 m  p4 – 1 m
  ```
  Distance · Par · Wind velocity (Against the wind)

  ```
  Stroke (0→60) ?
  ```
  Input the strength of your stroke

After inputting your stroke strength:

84 m

Remaining distance to the hole

When your ball comes to within 5m in front of or behind the pin:

P .

↑ Pin ↑ Ball

72 68 −4

↑ Total par ↑ Total strokes ↑ Final score (4 under)

Game Over

- End of game

HOW TO PLAY (EXAMPLE)

Start by pressing [S] [P0]

1 Holes ?

[3] [EXE] (To play 3 holes)

2 Hole No. 1

Start with hole number 1

3 [EXE] 385m p4−0m

↑ Distance

4 [EXE] Stroke (0→60)?

[4] [0] [EXE] (Try a strength of 40)

5 260m

↑ Remaining distance

6 [EXE] Stroke (0→60)?

[4] [2] [EXE]

7 123m

↑ Remaining distance

8 [EXE] Stroke (0→60)?

12 −18m

↑ Green over

13 [1] [3] [EXE] P .

Nice on

14 [3] P

1 putt

15 4 Over

Too bad

16 [EXE] Hole No. 2

17 [EXE] 331m p3−5m

18 [EXE] Stroke (0→60)?

[6] [0] [EXE] (I'll try the highest)

19 P .

One-on!

20 [5] P

1 putt

21 Birdie

22 [EXE] Hole No. 3

23 [EXE] 303m p3−9m

24 [EXE] Stroke (0→60)?

[4] [0] [EXE] (Lightly with a driver)

25 88m

26 [EXE] Stroke(0→60)?

[1] [5] [EXE] (Lightly with a 9 iron)

27 46m

28 [EXE] Stroke (0→60)?

[1] [6] [EXE]

29 Chip in !

30 [EXE] Par

31 [EXE] 10 13 3

↑ 3 over

32 [EXE] Game over

• Variables

Name of variable	Description	Name of variable	Description
A	Number of holes	I	Decision on direction
B	Distance	J	Final score
C	Par	K	Wind velocity
D	Green resistance	L	For differentiating hole-in-one and chip-in
E	Stroke strength	M	Total par
F	Key input	N	Hole no.
G	Loop counter	S	MID function
H	Putting distance	W	Loop counter

PROGRAM

MODE 1

```
CLEAR A
S DEFM
```

```
10 'AC
15 INPUT "Holes",A
20 B=RAN#*3:C=INT
   B+3:M=M+C
30 K=INT (RAN#*19)
   -9
40 B=INT (B*150)+2
   00
50 D=RAN#/5+.72
60 N=N+1:L=1
70 PRINT "Hole No.
   ";N
80 PRINT CSR 8;K;"
   m";CSR 6;C;CSR
   6;"P";CSR 0;B;"
   m"
90 INPUT "Stroke(0
   >60)",E
100 IF E>60 THEN 90
110 C=C-1:I=SGN B
120 B=INT B-INT (I*
   SQR 2*E/9.8*(E/
   SQR 2+K*I))
130 IF B=0;IF L=1;P
   RINT "Hole in o
   ne!":GOTO 300
140 IF B=0;PRINT "
    Chip in ! ":GO
   TO 300
150 L=0
160 IF ABS B>5;PRIN
   T B;"m":GOTO 90
170 PRINT CSR 5;"P"
   ;CSR B+5;".";
180 F$=KEY:IF F$=""
    THEN 180
190 C=C-1
200 $="12345"
210 FOR H=1 TO 5:IF
    F$=MID(H,1) TH
   EN 230
220 NEXT H:GOTO 180
230 I=SGN B
240 H=INT (H*D)+1
250 FOR G=1 TO H
260 PRINT CSR B+5;"
    ";:B=B-I
270 PRINT CSR B+5;"
   .";CSR 5;"P";
280 NEXT G
290 IF B<>0 THEN 160
295 PRINT
300 IF C<-4 THEN 44
   0
350 GOTO 390-C*10
360 PRINT "Albatoro
   ss":GOTO 450
370 PRINT "Eagle":G
   OTO 450
380 PRINT "Birdie":
   GOTO 450
390 PRINT "Par":GOT
   O 450
400 PRINT "Bogey":G
   OTO 450
410 PRINT "2 Bogey"
   :GOTO 450
420 PRINT "3 Bogey"
   :GOTO 450
430 PRINT "4 Over":
   GOTO 450
440 PRINT -C;" Over
   "
450 J=J-C:IF A<>N TH
   EN 20
460 PRINT M;M+J;J
470 PRINT "Game ove
   r":END
```

```
VAC
MODE
```

Total 702 steps

FLASHY FIVE

(Memory Pack Required with PB-100 only)

Here's a game of luck, played by you and the computer. The computer and you each call out numbers for the other to cancel. The first one to cancel out a row of five numbers going vertically, horizontally or diagonally wins the game.

RULES AND KEY OPERATION

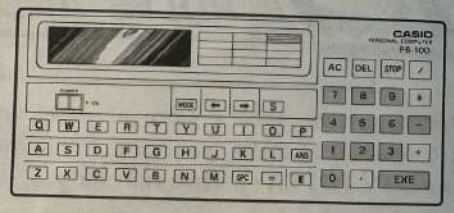
Before starting, take a piece of paper and divide it into 25 squares (5 squares × 5 squares) to make your card. Fill in the squares with numbers 1–25 at random (see HOW TO PLAY). Then start the program.

1. When the program is started, the computer takes about 4 seconds to standby.
2. As soon as the computer is ready, "ORDER?" is displayed. Decide whether you want to go first or last. If first, input a negative number and if last, a positive number.
3. If you start first, the computer displays a number after about 26 seconds. Cross out that number on your card. (Symbol "X" for your side)
4. When you have corssed out the number, press EXE key. "NUMBER?" is displayed. Circle a number on your card and input that number. (Symbol "O" for computer's side) Note that you cannot input numbers already crossed out or circled. Reverse steps 3 and 4 if you go last.
5. Repeat steps 3 and 4. You win when you cross out a vertical, horizontal or diagonal row of 5 numbers on your card.
6. When 4 numbers ("X"s) of a row are crossed out, warn the computer that you have one more to go before you cross out 5 numbers by pressing key ▣ after the "NUMBER?" display, instead of inputting a number. The computer then displays an unused number, so repeat from step 3. However, if the computer crosses out 4 numbers before you do, it displays "1 MORE TO GO". When this happens, press EXE and then input the number that you think is not the computer's 5th number.

7. If the computer finishes a row first, it displays "DIT IT!". If, on the other hand, you cross out a row of 5 numbers first, input [−] [1] and end the game.
8. Following the "LIST" display at the end, the computer displays each row of numbers in its card and then displays which numbers were crossed out (numbers input by you are shown as "X", those displayed by the computer for you to cross out are shown as "O", and those not called out, as "?"). This will let you know how the computer did in the game.

The keys used are the numerical keys [0] – [9] and [EXE].

Key [−] is pressed when you choose to go first. The numerical keys are used to input numbers 1 – 25. Press key [0] when declaring one-more-to-go. Press [EXE] to go to the next step after the computer displays a number, and also after each number input.



DISPLAY

- **Starting display**
  When the program is started, the following are displayed:

```
FLASHY FIVE
⇩
ORDER?
```

It will take about 4 seconds for "ORDER?" to be displayed. When it is, input a negative number if you want to go first and a positive number if you want to go last.

- **Play display**
  Say you start first.

```
19
↑
Number called out by the computer
(for you to cross out on your card)
```

What number will you call out?

```
NUMBER=?
```

```
22
↑
Number called out by you (for the
computer to cross out on its card)
```

It will take about 27 seconds for the computer to display a number.

- **When declaring one-more-to-go**

```
0
```
↑ You've got one-more-to-go

```
1 MORE TO GO
```
↑ The computer has one-more-to-go

- **End of game**

```
YOU WIN
```
You've won

```
DID IT!
```
The computer's won

- **Display of the computer's card**

```
LIST
```
Each row is displayed

```
3  19 15 17  5
```

```
?  ○  ○  ×  ○
```
Numbers cancelled, called out, and uncalled

- **Score**

```
SCORE:  1-0
```
↑ Computer's score ↑ Your score

- **When you input a number already called out**

```
PICK AGAIN
```


HOW TO PLAY (EXAMPLE)

Start by pressing [S] [D]

1
```
FLASHY FIVE
```
[EXE]

21	22	5	23	11
6	1	2	15	17
18	8	19	10	4
9	7	3	14	24
25	16	20	12	13

2
```
ORDER?
```

Press [−] [2] [EXE] (you go first)

3
```
19
```
↑ Number to be crossed out on your card

21	22	5	23	11
6	1	2	15	17
18	8	✕	10	4
9	7	3	14	24
25	16	20	12	13

[EXE]

4
```
NUMBER=?
```

[2] [2] [EXE]

5
```
22
```
↑ Number to be crossed out on the computer's card

21	○22	5	23	11
6	1	2	✕	17
18	8	✕	10	4
9	7	3	14	24
25	16	20	12	13

6
```
15
```
↑ Number to be crossed out by you

○21	○22	✕5	○23	✕11
○6	○1	○2	✕15	○17
18	8	✕	✕	4
✕9	✕7	✕3	✕14	✕24
25	16	✕20	○12	○13

22
```
11
```
One more to go!

Press EXE
(You declare one-more-to-go)

23 0

24 25

EXP − 1 EXE

25 YOU WIN

EXE

26 LIST

EXE

27 3 19 15 17 5

1st square, 2nd square, 3rd square, 4th square, 5th square

EXE

28 OOO×O

32 SCORE : 0−1

- Variables

Name of Variable	Description	Name of Variable	Description
B	Value input in indicating first or last to play	N	Row of card
C	Computer's score	P	Number of circles in each diagonal or vertical row
D	Your score	Q	Number of circles in each diagonal or vertical row
F	Number of circles in each horizontal row	R	Working area for calculations
G	Number of circles in each vertical row	S	Starting position on card
I	Loop counter	T(0)~ T(24)	Computer's card
J	Loop counter	T(25)~ T(49)	Table for recording marks entered 1=initial value 2=specified
K	Key input numerical value	T(50)~ T(70)	Working area
M	Row of card		

PROGRAM

MODE 1

CLEAR A EXE

S DEFM 6 8 EXE

```
1 PRINT "FLASHY F
  IVE":C=0:D=0
2 FOR I=0 TO 24:T
  (I)=I+1:NEXT I
3 FOR J=0 TO 24
4 I=INT (RAN#*25)
  :R=T(J):T(J)=T(
  I):T(I)=R:NEXT
  J
5 FOR I=25 TO 49:
  T(I)=1:NEXT I
6 INPUT "ORDER",B
  :IF B<0 THEN 19
7 INPUT "NUMBER="
  ,K:IF K<0:PRINT
   "YOU WIN":D=D+
  1:GOTO 42
8 IF K=0 THEN 19
9 FOR I=0 TO 24
10 IF T(I)=K THEN
   13
11 NEXT I
12 PRINT "PICK AGA
  IN":GOTO 7
13 IF T(I+25)=1 TH
  EN 12
14 T(I+25)=2:GOSUB
    33
15 IF (F-16)*(G-16
  )*(P-16)*(Q-16)
  =0 THEN 17
16 PRINT "1 MORE T
  O GO":GOTO 7
17 IF (F-32)*(G-32
  )*(P-32)*(Q-32)
  =0 THEN 19
18 PRINT "DID IT!"
  :C=C+1:GOTO 42
19 FOR I=0 TO 24
20 IF T(I+25)=1:T(
  I+50)=1025:GOTO
    24
21 GOSUB 33
22 IF F+G+P+Q=1 TH
  EN 28
23 T(I+50)=F*F+G*G
  +P*P+Q*Q
24 NEXT I:L=50
25 FOR I=51 TO 74
26 IF T(I)<T(L):L=
  I
27 NEXT I:I=L-50
28 PRINT T(I):T(I+
  25)=0:GOTO 7
29 P=1:S=(6-R)*2
30 FOR J=S+25 TO 4
  9-S STEP R
31 P=P*T(J)
32 NEXT J:RETURN
33 IF I=12:R=4:GOS
  UB 29:Q=P:GOTO
  35
34 Q=0
35 IF FRAC (I/6)=0
  :R=6:GOSUB 29:G
  OTO 38
36 IF FRAC (I/4)=0
  :R=4:GOSUB 29:G
  OTO 38
37 P=0
38 F=1:G=1:M=INT (
  I/5)*5:N=I-M
39 FOR J=0 TO 4
40 F=F*T(M+J+25):G
  =G*T(N+J*5+25)
41 NEXT J:RETURN
42 A=0:PRINT "LIST
  "
43 FOR I=0 TO 24 S
  TEP 5
44 PRINT CSR 8;T(I
  +4);CSR 6;T(I+3
  );CSR 4;T(I+2);
45 PRINT CSR 2;T(I
  +1);CSR 0;T(I)
46 IF T(A+25)=0:PR
  INT " o";:GOTO
  49
47 IF T(A+25)=2:PR
  INT " x";:GOTO
  49
48 PRINT " ?";
49 IF FRAC ((A+1)/
  5)=0:A=A+1:GOTO
   46
50 PRINT "":A=A+1:
  NEXT I:PRINT "S
  CORE:";C;"-";D:
  GOTO 2
```

VAC EXE

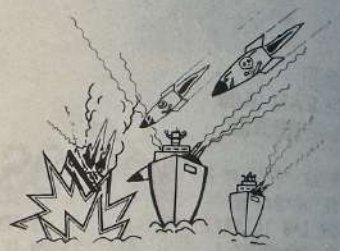
MODE 2

Total 953 steps

BOMBARDMENT GAME

(Memory Pack Required with PB-100 only)

Your fleet is under attack. Various enemy bombardments are launched against your 5 heavy cruisers. You have to save them by shooting long or short beams at the enemy before they attack the cruisers. Study the various forms of enemy attack carefully and destroy as many enemy ships as possible.



RULES AND KEY OPERATION

First, study carefully the 6 types of enemy attack described below. The key to winning this game lies in grasping the different forms of attack your enemy launches against you. Following each description is the number of points you score when you hit that particular enemy ship with a long or short beam.

- ＊..... Makes no attack, just passes by (50 points)
- ＜.... Enters the position you occupy and blasts itself at a probability of 3/4 to destroy your heavy cruiser (100 points)
- ≦.... Attacks with laser beams from a point 3 spaces outside the position you occupy (200 points)
- X..... Attacks with laser beams from a random position (450 points)
- ○.... Crashes into your heavy cruiser and sinks it (400 points)
- ?..... Unknown (one of the 5 above but you don't know which)

The attacks are launched one by one. When an enemy ship appears on the display and starts approaching, attack before you're attacked. You have 2 ways of attacking, as follows:

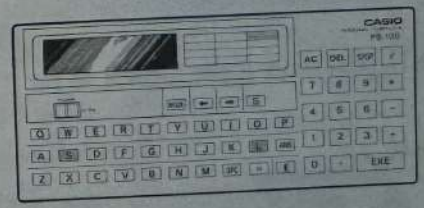
With short beams (press [S]) – effective only against ＊ and ＜ when they've entered the position you occupy (thus, wait until they've entered the position you occupy before shooting)

With long beams (press [L]) – effective only against ≦, X and ○ outside your position

Note that you have only one chance to discharge your laser gun at each enemy ship that appears. If you make a mistake on which beam (long or short) to use, you'll be destroyed, so be careful. You have 5 heavy cruisers. The game ends when all 5 are destroyed.

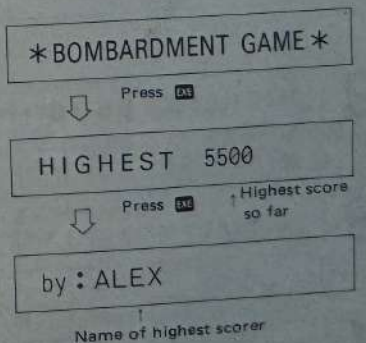
The keys used are [S] and [L].

Key [S] is for attacking with short beams.
Key [L] is for attacking with long beams.



DISPLAY

- **Starting display**



• Play display

```
≧ : < 5
```
Heavy cruiser · Enemy ship · Number of heavy cruisers

```
≧ : * 5
```
Position you occupy · Outside your position

• End of game

```
SCORE−1600
```

```
GAME OVER
```

• If your score is the highest so far

```
Name ?
```

HOW TO PLAY (EXAMPLE)

Start by pressing S P0

1
```
*BOMBARDMENT GAME*
```
EXL

2
```
HIGHEST−5500
```
EXE

3
```
by : ALEX
```
Press EXE to start game

4
```
≧ : ≦ 5
```
Your ship · Enemy ship that can be destroyed only by long beams · Number of your ships

5
```
≧ : ≦ 5
```
When the enemy ship approaches this far, discharge your laser gun before they attack

Press L

6
```
≧..... ! 5
```
Zap! (200 points)

If you miss...

7
```
! ( ( ( ( ( ≦ 4
```
Your ship is destroyed · The enemy attacks with laser beams

Here comes the next enemy ship

8
```
≧ : < 4
```
Position you occupy · Destroy this enemy ship with a short beam when it enters your position

Press S (short beam)

9
```
≧··· ! 4
```
Made it!

10
```
≧ : ? 4
```
You don't know how it will attack. Take a guess and try long beams...

Press L

11
```
! ? 3
```
It's a < ! You're destroyed

12
```
≧ : < 3
```
It'll enter the position you occupy and blast itself so don't let it get you

Press L

13
```
! ( + ) 2
```
Blast! Should've been short beams

18
```
≧ : X 2
```
If you hit it, you score 450 points

Press L

19
```
! ( ( ( ( X 1
```
It attacked first! · Last cruiser

20
```
≧ : * 1
```
It doesn't attack so get it with short beams when it enters your position

31
```
SCORE−4500
```

32
```
GAME OVER
```

• Variables

Name of variable	Description	Name of variable	Description
B	Selection of enemy attack	I	Position of laser beams discharged by X
C	Attack probability of "<"	J	X's score
D	Types of enemy attack	K	Position of your laser beams
E	Enemy position	L	Number of "≧"s
F	Score	M	Flag for judging key input
G$	Key input	W	Highest score so far
H	Flag when attacked	Z	Position of start of X's attack

PROGRAM

MODE 1

CLEAR A EXE

S DEFM ∅ EXE

```
10 L=5:F=0:$="*(ΣX
   0"
20 PRINT "*BOMBARD
   MENT GAME*"
110 PRINT "HIGHEST"
   ;-W:PRINT "by:"
   ;N$;"  "
120 A=INT (RAN#*10)
   :B=INT (RAN#*30
   ):IF B<8;D=1:GO
   TO 160
130 IF B≥8;D=2:IF B
   ≥12;D=3:IF B≥23
   ;D=4:IF B≥26;D=
   5
140 C=INT (RAN#*4):
   J=INT (RAN#*3)*
   100+450
150 Z=INT (RAN#*4)+
   3:Y=INT (RAN#*3
   )+6
160 H=0:G$=" ":M=0:
   FOR E=Y TO 1 ST
   EP -1
170 PRINT CSR 0;"
            ";CSR 1
   0;L;CSR 0;"≥";
175 PRINT CSR 3;":"
   ;CSR E+1;" ";
180 IF A=0:PRINT CS
   R E;"?";:GOTO 2
   30
190 IF D=4;IF E≥6 T
   HEN 220
200 IF D=5;IF E≥5 T
   HEN 220
210 PRINT CSR E;MID
   (D,1);
220 IF E=1;PRINT CS
   R E;" ";
230 GOTO 100*D+200
300 IF M=1 THEN 100
   0
310 G$=KEY:IF E>2 T
   HEN 1000
320 IF G$="S";PRINT
    CSR E;"!";:F=F
   +50:GOTO 2000
330 GOTO 1000
400 IF M=1 THEN 430
410 G$=KEY:IF E>2 T
   HEN 1000
420 IF G$="S";PRINT
    CSR E;"!";:F=F
   +100:GOTO 2000
430 IF C=0 THEN 100
   0
440 IF E=2;PRINT CS
   R 2;"+";CSR 1;"
   (+)";CSR 0;"!(+
   )";:H=1:GOTO 20
   00
450 GOTO 1000
500 Z=6
600 IF M=1 THEN 660
610 G$=KEY:IF G$≠"L
   " THEN 660
620 GOSUB 3000
630 IF D=4 THEN 650
640 F=F+200:GOTO 20
   00
650 PRINT CSR 4;J;:
   F=F+J:GOTO 2000
660 IF E≠Z THEN 100
   0
670 FOR I=E-1 TO 1
   STEP -1:PRINT C
   SR I;"(";:NEXT
   I
680 PRINT CSR 0;"!"
   ;:H=1:GOTO 2000
700 IF M=1 THEN 740
710 G$=KEY:IF G$≠"L
   " THEN 740
720 GOSUB 3000
730 F=F+400:GOTO 20
   00
740 IF E=1;PRINT CS
   R 0;"!";:H=1:GO
   TO 2000
1000 IF G$="L";M=1
1010 IF G$="S";M=1
1020 NEXT E
2000 IF H=1;L=L-1:IF
    L≠0 THEN 2020
2010 GOTO 120
2020 PRINT CSR 0;" S
   CORE";-F;"
    ";
2030 IF F>W:W=F:INPU
   T "Name",N$
2040 PRINT "GAME OVE
   R":END
3000 FOR K=1 TO E-1:
   PRINT CSR K;"-"
   ;:NEXT K:PRINT
   CSR E;"!";
3010 RETURN
```

VAC EXE

MODE ∅

Total 971 steps

SLOT-MACHINE GAME

(Memory Pack Required with PB-100 only)

This game is a computer version of a slot machine.

The slot machine is simulated in this game by having symbols (numbers, letters and signs) flash on the display in quick succession. To hit the jackpot — 250 points — you must get three sevens (777). In addition, there's a hidden factor involved which can double the number of points you win. The game ends when your credit reaches 0.

RULES AND KEY OPERATION

As soon as the program is started, "LIMIT? " is displayed. Set an arbitrary maximum limit on the number of points you can stake in a single bet and input that number. Next, "Credit—10→bet? " is displayed. Input the number of points you want to bet. This cannot exceed the limit you set at the start of the game and must not exceed your credit. (At the start of the game, your credit is 10 points.) If you mistakenly input too many points, the computer will ask you to repeat this step.

As soon as you input your bet, the computer starts flashing various symbols one after another on the display. Stop a symbol by pressing any key except those shown on page 109. (Choose a key that's easy for you to press.) Then go on to stop two more flashing symbols which appear to the right of the first in the same way. It will take about 0—1.5 sec for a symbol to stop. If you don't press any key, the flashing symbols will stop automatically after about 7.5—11 sec. In this case, the display stops automatically at "$$$."

As soon as the 3rd symbol is stopped, the hidden factor (×1 or ×2) is displayed at the far right. If the 3-symbol combination you end up with matches any of the combinations shown below, you win your bet. If not, you lose. How much you win is calculated as follows:

number of points given for the combination × number of points bet × hidden factor (1 or 2).

< COMBINATIONS >

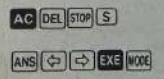
①	②	③	Points
7	7	7	250
—	—	—	200
7	7	—	62
—	7	7	62
$	$	$	50
G	G	G	25
7	7	•	25
•	7	7	25
L	L	L	20
—	—	•	20
•	—	—	20
$	$	—	12
—	$	$	12
A	A	A	8
G	G	—	6
—	G	G	6
L	L	—	5
—	L	L	5
$	$	•	5
•	$	$	5
A	A	—	2
—	—	A	2
$	•	•	2
•	7	•	2
•	•	7	2

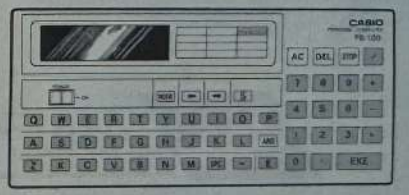
* "•" = any symbol

If you lose your bet, "YOU LOSE! " is displayed, and the number of points you staked in the bet is subtracted from your credit. The game ends when your credit reaches 0.

Since you can use any key to stop a symbol, use the one that's easiest for you to press.

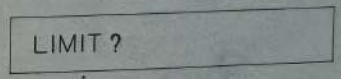
You cannot, however, use the following keys to stop a symbol:



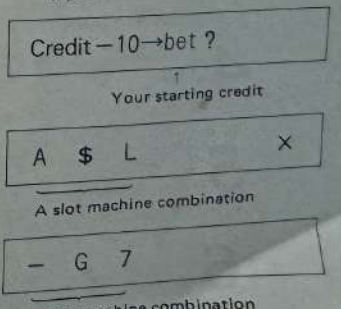


DISPLAY

- **Starting display**



- **Play display**



YOU LOSE!

↑ When you lose a bet

2 5 1

No. of points ↑ Hidden factor (1 or 2)

• End of game

＊Game Over＊

HOW TO PLAY (EXAMPLE)

Start by pressing S P0

1 LIMIT?

100 EXE Input maximum limit for bets

2 Credit −10→bet ?

10 EXE Input your bet

3 A

Press P (or any key you like)

4 A G

P

5 A A

P

6 A A −

7 A A − ×2

↑ Hidden factor

8 2 10 2

↑ No. of points ↑ No. of points bet ↑ Hidden factor

9 Credit −40→bet ?

20 EXE Input your bet

10 A

P

11 A G

P

12 A G −

13 A G − ×1

14 YOU LOSE!

Display moves leftward

15 Credit −20→bet ?

20 EXE Input your bet

16 L

17 L A

18 L A G

19 L A G ×1

20 YOU LOSE!

21 ＊Game Over＊

• Variables

Name of variable	Description	Name of variable	Description
P	Credit	M	Symbol display position
S	Symbol data	E	Hidden factor (1 or 2)
L	Limit	G	Increase rate
I	Loop counter	D$	Symbol
J	Loop counter	Q	No. of points bet
N	Loop counter	B	Working area
A$	Key input	W	Loop counter

PROGRAM

```
MODE 1
CLEAR A
S DEFM
```

```
10 P=10
20 INPUT "LIMIT",L
   :IF L<0 THEN 20
30 $="A$LAG-AL$AG7
   A.."
40 PRINT "Credit";
   -P;:INPUT "->bet
   ",Q:Q=INT Q
50 IF Q>P THEN 40
60 IF Q>L THEN 40
70 D$="":FOR N=0 T
   O 2:M=2*N:I=0
80 I=I+1:IF I>1 TH
   EN 140
90 J=0
100 J=J+1:IF J>15 T
   HEN 80
110 PRINT CSR M;MID
   (J,1);
120 A$=KEY:IF A$=""
   ;B=INT (RAN#*4)
   :GOTO 150
130 GOTO 100
140 B=INT (RAN#*14)
150 FOR I=J TO M+B:
   K=I:IF K>15:K=I
   -14
160 A$=MID(K,1):PRI
   NT CSR M;A$;:FO
   R W=1 TO 50:NEX
   T W
170 NEXT I:D$=D$+A$
   :NEXT N:$=D$
180 IF RAN#>.8:E=2:
   GOTO 200
190 E=1
200 PRINT CSR 7;" x
   ";E;:FOR W=1 TO
    95:NEXT W
210 GOSUB 300:IF G>
   0 THEN 230
220 PRINT :PRINT "Y
   OU LOSE!";:FOR
   W=1 TO 100:NEXT
    W:P=P-Q
222 IF P<0:PRINT :P
   RINT "*Game Ove
   r*":END
225 GOTO 30
230 B=Q*G*E:PRINT :
   PRINT Q;G;E;-B;
   ":";:FOR W=1 TO
    30:NEXT W
240 P=P+B:GOTO 30
300 IF D$="777";G=2
   50:RETURN
310 IF D$="---";G=2
   00:RETURN
320 IF D$="77-" THE
   N 560
330 IF D$="-77" THE
   N 560
340 IF D$="$$$";G=5
   0:RETURN
350 IF D$="GGG" THE
   N 570
360 IF MID(1,2)="77
   " THEN 570
370 IF MID(2,2)="77
   " THEN 570
380 IF D$="LLL" THE
   N 580
390 IF MID(1,2)="--
   " THEN 580
400 IF MID(2,2)="--
   " THEN 580
410 IF D$="$$-" THE
   N 590
420 IF D$="-$$" THE
   N 590
430 IF D$="AAA";G=8
   :RETURN
440 IF D$="GG-" THE
   N 600
450 IF D$="-GG" THE
   N 600
460 IF D$="LL-" THE
   N 610
470 IF D$="-LL" THE
   N 610
480 IF MID(1,2)="$$
   " THEN 610
490 IF MID(2,2)="$$
   " THEN 610
500 IF D$="AA-" THE
   N 620
510 IF D$="-A" THE
   N 620
520 IF MID(1,1)="$"
    THEN 620
530 IF MID(2,1)="7"
    THEN 620
540 IF MID(3,1)="7"
    THEN 620
550 G=0:RETURN
560 G=62:RETURN
570 G=25:RETURN
580 G=20:RETURN
590 G=12:RETURN
600 G=6:RETURN
610 G=5:RETURN
620 G=2:RETURN
```

```
VAC
MODE
```

Total 984 steps

BLACKJACK GAME

(Memory Pack Required with PB-100 only)

This is a computer version of blackjack. The computer is the dealer, and you are the challenger. The rules are about the same as in regular blackjack.

RULES AND KEY OPERATION

The computer is the dealer, and you are the challenger. The 13 cards are indicated as follows:

A, 2, 3, 4, 5, 6, 7, 8, 9, T, J, Q, K

(T, J, Q and K each have values of 10, and A has a value of 1 or 11.)

First, the dealer deals out 2 cards to both players. Your 1st card is dealt face up, your 2nd card face down. The dealer's 2 cards are kept face down. Input the number of chips you want to bet. (If you don't want to place a bet, input 0. In this case, "PASS" is displayed, and you lose 10% of your chips.) Next, if the dealer's 2 cards total 21 (in value), only those cards are displayed, and you lose twice the number of chips you bet. Otherwise, the number of chips you bet is displayed at the left, and your 2nd card is opened. If you want another card, press [Y] key. If not, press [N] key. As long as your cards total less than 21, you may continue to ask for more cards until you have 6 cards. Any time your cards total more than 21, you lose, so be careful. You must press [N] key and challenge the dealer while your cards remain 21 or less. The dealer continues to take additional cards until its card value totals 17 or more. Then, whoever has the closest to 21 but not more than 21 wins the bet. If your and the dealer's cards total the same, you lose. If you lose, you lose the chips staked in the bet. If you win, you receive the number of chips you bet.

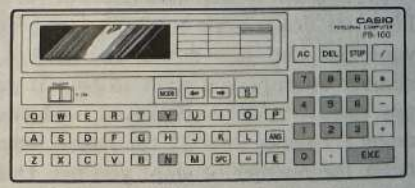
* When your cards total 21 in the following ways, you win, even if the dealer also has 21:

AJ (or JA)	3 times (with a probability of 1/16)	
3A25T	2 times (win with 5 cards)	Any number combinations of 5 or 6 cards in any order.
AA2764	3 times (win with 6 cards)	
777	3 times	
678	2 times (in any order)	

In this case, you win 2 or 3 times the number of chips you staked in the bet, as given above.
Continue to play until you have 10,000 chips or more, or until you lose all your chips. The game then ends.

The keys used are the numerical keys and key [EXE] for inputting your bet.

When you want another card, press [Y] key, and when you don't want any more, press [N] key.



DISPLAY

• **Starting display**

♥ Blackjack Game ♥
↑ Game Title

: Y / yes :N / no
↑ When you want another card
↑ When you don't want any more cards

HIGHEST: 20
↑ Highest score so far

• **Play display**

CHIPS : 10
↑ Your chips

3■■■■ : _ ■■■■?
↑ Your cards
↑ Dealer's cards
↑ Awaiting input on number of chips staked

349■■ : _ ■■■■
↑ Your cards
↑ Dealer's cards

When you win

YOU WIN!

When the dealer's cards or your cards total more than 21

♠♥♦♣XXXXXXXX

When you lose

YOU LOSE!

- **End of game**

If your score is the highest so far — NEW HIGH

If you win the game (10,000 chips or more) — CONGRATS

If you lose the game (no chips) — TOO BAD

End of round — ∗GAME OVER∗

HOW TO PLAY (EXAMPLE)

Start by pressing [S][PQ]

1 ♥ Blackjack Game ♥

[EXE]

2 :Y/yes :N/no

[EXE]

3 HIGHEST:12

(Highest score so far)

[EXE]

4 CHIPS:10

Your chips

[EXE]

5 7■■■■:_■■■■?

Press [5][EXE] (Bet 5 chips)

6 7T ■■:_■■■■

(You have 17)

Press [Y] (To ask for another card. If you don't want any more cards, press [N].)

7 7T2■■:_■■■■

Press [N] (You challenge with 19)

8 7T2■■:AAJ48

The dealer has 24, so you win.

9 YOU WIN!

[EXE]

10 CHIPS:15

[EXE]

11 A■■■■:_■■■■?

Repeat from step 6.

- **Variables**

Name of variable	Description	Name of variable	Description
B	Integers 1~13	O	Loop counter
C	Loop counter	P	Flag for 777
D	Chips	Q	Flag for 678
E	Number of chips bet	R	Loop counter
F	Number of cards dealt additionally to you	S	Loop counter
G	Number of cards dealt additionally to the dealer	T	Loop counter
H	Loop counter	U	Flag for AJ
I	Total value of cards held by you	W	PASS
J	Total value of cards held by the dealer	X	Highest score so far
K	Loop counter	Z(0)~ Z(5)	Your cards
L$	Key input	Z(6)~ Z(11)	Dealer's cards
M	Chips increase rate	Z(12)~ Z(24)	Counter to prevent dealing more than 4 cards with the same value
N	Loop counter		

PROGRAM

MODE 1

CLEAR A EXE

S DEFM 24 EXE

```
10 $="A23456789TJQ
   K":D=10:PRINT "
   *Blackjack Game
   *"
20 PRINT ":Y/yes:N
   /no":PRINT "HIG
   HEST:";X
30 E=0:F=0:G=0:L$=
   " ":U=0
40 FOR R=12 TO 24:
   Z(R)=0:NEXT R
50 FOR C=0 TO 11
60 B=INT (RAN#*13)
   +1
70 IF Z(B+11)>4 TH
   EN 60
80 Z(B+11)=Z(B+11)
   +1:Z(C)=B:NEXT
   C
90 PRINT "CHIPS:";
   D
100 PRINT CSR 0;MID
   (Z(0),1);"■■■■:
   ";CSR 6;"_■■■■"
   ;CSR 11;:INPUT
   E
110 IF E>D THEN 100
120 IF E=0:W=D:D=D-
   INT (D/10):PRIN
   T "PASS";D-W:GO
   TO 30
130 IF E<0 THEN 90
140 PRINT CSR 0;MID
   (Z(0),1);"■■■■:
   ";CSR 6;"_■■■■"
   ;
150 G=2:GOSUB 700
160 IF J=21:PRINT C
   SR 7;MID(Z(6),1
   );MID(Z(7),1):D
   =D-E*2:GOTO 490
170 PRINT CSR 1;MID
   (Z(1),1);
180 L$=KEY:IF L$="N
   " THEN 260
190 IF L$="Y" THEN
   210
200 GOTO 180
210 IF F=0:F=2
220 F=F+1
230 PRINT CSR F-1;M
   ID(Z(F-1),1);:G
   OSUB 600:IF I>2
   1:D=D-E:GOTO 50
   0
240 IF F=6 THEN 270
250 GOTO 180
260 IF F=0:F=2:GOSU
   B 600
270 PRINT CSR 7;MID
   (Z(6),1);MID(Z(
   7),1);
280 IF J>16 THEN 31
   0
290 G=G+1:IF G=6 TH
   EN 320
300 PRINT CSR G+6;M
   ID(Z(G+5),1);:G
   OSUB 700:GOTO 2
   80
310 IF J>21:D=D+E:G
   OTO 490
320 IF I*21 THEN 47
   0
330 P=0:Q=0:M=0
340 IF F=5:M=1:GOTO
    480
350 IF F=6:M=2:GOTO
    480
360 FOR N=0 TO 2:IF
    Z(N)=7:P=P+1
370 NEXT N
380 IF P=3:M=2:GOTO
    480
390 FOR O=0 TO 2
400 IF Z(O)=6:Q=Q+1
410 IF Z(O)=7:Q=Q+.
   1
420 IF Z(O)=8:Q=Q+.
   01
430 NEXT O
440 IF Q=1.11:M=1:G
   OTO 480
450 IF Z(0)=1:IF Z(
   1)=11:U=1
460 IF U=1:IF INT (
   RAN#*15)=7:PRIN
   T "Blackjack !"
   :M=2:GOTO 480
470 IF I≤J:PRINT "Y
   OU LOSE !  ":D=
   D-E:GOTO 500
480 PRINT "YOU WIN
   !   ":D=D+E*(1+
   M)
490 IF X≤D:X=D:PRIN
   T "NEW HIGH"
500 IF D≤0:PRINT "T
   OO BAD":GOTO 53
   0
510 IF D≥10000:PRIN
   T "CONGRATS":GO
   TO 530
520 GOTO 30
530 PRINT "*GAME OV
   ER*":END
600 I=0:FOR H=0 TO
   F-1:IF Z(H)=1:I
   =I+11:GOTO 630
610 IF Z(H)>9:I=I+1
   0
620 IF Z(H)≤9:I=I+Z
   (H)
630 NEXT H
640 IF I<22:RETURN
650 FOR S=0 TO F-1:
   IF Z(S)=1:I=I-1
   0:IF I<22:RETUR
   N
660 NEXT S:GOTO 770
700 J=0:FOR K=6 TO
   G+5:IF Z(K)=1:J
   =J+1:GOTO 730
710 IF Z(K)>9:J=J+1
   0
720 IF Z(K)≤9:J=J+Z
   (K)
730 NEXT K
740 IF J<22:RETURN
750 FOR T=6 TO G+5:
   IF Z(T)=1:J=J-1
   0:IF J<22:RETUR
   N
760 NEXT T:GOTO 770
770 PRINT "♠♥♦♣xxxx
   xxxx":RETURN
```

VAC EXE

MODE 2

Total 1,344 steps