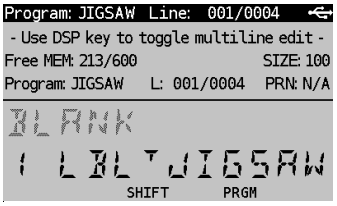
JIGSAW

Display



(HP-41CX, Hewlett Packard 1983 and DM41X, SwissMicros 2020)

Overview[1]

The JIGSAW program is the famous 2-dimensional jigsaw game wrapped into a 1-dimensional display. The user can move the pieces Up, Down, Right or Left depending on where the empty space is. The game can be played with a numerical series of "pieces" varying from 1-8 or a more difficult alphanumerical mode in which the "pieces" can either vary from A..S-H-Z, e.g. N-U or I-K as shown in the example below. The aim is to make the least number of moves when ordering a random pattern like:

IDK:EF_:JGH into DEF:GHI:JK_

The program will ask the user to choose the Numerical (N) or the Alphanumerical (A) mode at the beginning. Between the moves the calculator will make a time-out which is defaulted to 4 periods of 10 seconds. Each interval is indicated by displaying flags 1-4. Without a response to the time-out question at data entry, e.g. 60 seconds, the calculator will take 40 seconds as default. The reason for this is that the user can simply press the [U] (ENTER), [D] (LOG), [R] (7) or [L] (STO) key to make a move. The GETKEYX instruction will intercept this until the time-out has passed. If so, the user will then get the option to continue or not. The display will prompt for R/S in that case. When a valid move is requested, the calculator will check if the move is allowed. If not, it will show this with "??" and give beep tone such that the user can try again.

Example

KEYSTROKES	DISPLAY	COMMENTS
		Run JIGSAW
[XEQ][ALPHA]JIGSAW[ALPHA]	N/A?	Press N for Numerical and A for Alpha
A[R/S]	TIME-OUT=?	Default 40 seconds, or enter number, e.g. 60
60[R/S]	IDK:EFK:JGH	I D _ E F K J G H
[D]	DOWN ??	Press D for down
	R/S?	Time-out expired. Press R/S to continue
60[R/S]	IDK:EFK:JGH	I D _ E F K J G H
[U]	UP ..	
	IDK:EF_:JGH	I D K E F _ J G H
[R]	RIGHT ..	

[1] *This program is copyright and is supplied without representation or warranty of any kind. The author assumes no responsibility and shall have no liability, consequential or otherwise, of any kind arising from the use of this program material or any part thereof*

	IDK: E_F: JGH	I D K E _ F J G H
[R]	RIGHT . .	
	ID: _EF: JGH	I D K _ E F J G H
[D]	DOWN . .	
	_DK: IEF: JGH	_ D K I E F J G H
[L]	LEFT . .	
	D_K: IEF: JGH	D _ K I E F J G H
[U]	UP . .	
	DEK: I_F: JGH	D E K I _ F J G H
[R]	RIGHT . .	
	DEK: _IF: JGH	D E K _ I F J G H
[D]	DOWN . .	
	_EK: DIF: JGH	_ E K D I F J G H
[L]	LEFT . .	
	E_K: DIF: JGH	E _ K D I F J G H
[L]	LEFT . .	
	EK_: DIF: JGH	E K _ D I F J G H
[U]	UP . .	
	EKF: DI_: JGH	E K F D I _ J G H
[R]	RIGHT . .	
	EKF: D_I: JGH	E K F D _ I J G H
[U]	UP . .	
	EKF: DGI: J_H	E K F D G I J _ H
[L]	LEFT . .	
	EKF: DGI: JH_	E K F D G I J H _
[D]	DOWN . .	
	EKF: DG_: JHI	E K F D G _ J H I
[R]	RIGHT . .	
	EKF: D_G: JHI	E K F D _ G J H I
[D]	DOWN . .	
	E_F: DKG: JHI	E _ F D K G J H I
[R]	RIGHT . .	
	_EF: DKG: JHI	_ E F D K G J H I
[U]	UP . .	
	DEF: _KG: JHI	D E F _ K G J H I
[L]	LEFT . .	
	DEF: K_G: JHI	D E F K _ G J H I
[L]	LEFT . .	
	DEF: KG_: JHI	D E F K G _ J H I
[U]	UP . .	
	DEF: KGI: JH_	D E F K G I J H _
[R]	RIGHT . .	

	DEF :KGI :J_H	D E F K G I J _ H
[R]	RIGHT . .	
	DEF :KGI :_JH	D E F K G I _ J H
[D]	DOWN . .	
	DEF :_GI :KJH	D E F _ G I K J H
[L]	LEFT . .	
	DEF :G_I :KJH	D E F G _ I K J H
[U]	UP . .	
	DEF :GJI :K_H	D E F G J I K _ H
[L]	LEFT . .	
	DEF :GJI :KH_	D E F G J I K H _
[D]	DOWN . .	
	DEF :GJ_ :KHI	D E F G J _ K H I
[R]	RIGHT . .	
	DEF :G_J :KHI	D E F G _ J K H I
[U]	UP . .	
	DEF :GHJ :K_I	D E F G H J K _ I
[R]	RIGHT . .	
	DEF :GHJ :_KI	D E F G H J _ K I
[D]	DOWN . .	
	DEF :_HJ :GKI	D E F _ H J G K I
[L]	LEFT . .	
	DEF :H_J :GKI	D E F H _ J G K I
[U]	UP . .	
	DEF :HKJ :G_I	D E F H K J G _ I
[L]	LEFT . .	
	DEF :HKJ :GI_	D E F H K J G I _
[D]	DOWN . .	
	DEF :HK_ :GIJ	D E F H K _ G I J
[R]	RIGHT . .	
	DEF :H_K :GIJ	D E F H _ K G I J
[U]	UP . .	
	DEF :HIK :G_J	D E F H I K G _ J
[L]	LEFT . .	
	DEF :HIK :GJ_	D E F H I K G J _
[D]	DOWN . .	
	DEF :HI_ :GJK	D E F H I _ G J K
[R]	RIGHT . .	
	DEF :H_I :GJK	D E F H _ I G J K
[R]	RIGHT . .	
	DEF :_HI :GJK	D E F _ H I G J K
[U]	UP . .	

	DEF: GHI: _JK	D E F G H I _ J K
[L]	LEFT . .	
	DEF: GHI: J_K	D E F G H I J _ K
[L]	LEFT . .	
	DEF: GHI: JK_	D E F G H I J K _
	OK 46x	
[R/S]	N/A?	Choose Numerical or Alpha and run again

Program Listing

The listing of JIGSAW is given below with 2 XROM functions SEED and RNDM on lines 21, 22 and 35. These can be taken as explicit calls to other programs in memory or to XROM functions in for example the CCD module. If replaced by these XROM functions (as in the RAW and TXT file) the total number of bytes will be 335 instead of 347.

```
01▪LBL "JIGSAW"
02▪LBL 02
03 CLRG
04 CF 00
05 "N/A?"
06 AON
07 PROMPT
08 AOFF
09 ATOX
10 78
11 X=Y?
12 SF 00
13 "TIME-OUT=?"
14 40
15 PROMPT
16 4
17 /
18 STO 14
19 CF 29
20 FIX 00
21 XEQ "SEED"
22 XEQ "RNDM"
23 19
24 *
25 73
26 +
27 INT
28 FS?C 00
29 57
30 STO 11
31 9
32 STO 10
33 SF 06
34▪LBL 06
35 XEQ "RNDM"
36 9
37 *
38 1
39 +
40 INT
41 FS? 06
42 STO 00
43 .
44 X#NN?
45 GTO 06
46 RCL 11
47 RCL 10
48 -
49 FS?C 06
50 95
51 STO IND T
52 DSE 10
53 GTO 06
54▪LBL 11
55 CLA
56 "::"
57 ASTO 10
58 CLA
59 9
60▪LBL 01
61 STO Z
62 3
63 MOD
64 X=0?
65 ARCL 10
66 +
67 RCL IND T
68 XTOA
69 R^
70 DSE X
71 GTO 01
72 ATOX
73 ATOX
74 AVIEW
75 8
76 ABS
77 RCL 09
78▪LBL 09
79 RCL IND L
80 X<=Y?
81 GTO 07
82 DSE L
83 GTO 09
84 PSE
85 "OK "
86 ARCL 13
87 >"X"
88 CLX
89 X<>F
90 SF 29
91 FIX 05
92 PROMPT
93 GTO 02
94▪LBL 07
95 CLX
96 STO 12
97▪LBL 10
98 AVIEW
99 CLX
100 X<>F
101 1
102 ST+ 12
103 SF IND 12
104 FS?C 05
105 GTO 03
106 RCL 14
107 GETKEYX
108 RDN
109 X=0?
110 GTO 10
111 SF 25
112 GTO IND X
113 CF 25
114 GTO 10
115▪LBL 52
116 3
117 ENTER
118 6
119 ENTER
120 9
121 XEQ 04
122 1
123 "RIGHT"
124 GTO 08
125▪LBL 33
126 1
127 ENTER
128 4
129 ENTER
130 7
131 XEQ 04
132 -1
133 "LEFT"
134 GTO 08
135▪LBL 14
136 7
137 ENTER
138 8
139 ENTER
140 9
141 XEQ 04
142 3
143 "DOWN"
144 GTO 08
```

```
145▪LBL 61
146 1
147 ENTER
148 2
149 ENTER
150 3
151 XEQ 04
152 -3
153 "UP"
154 GTO 08
155▪LBL 04
156 RCL 00
157 XEQ 05
158 XEQ 05
159 XEQ 05
160 RTN
161▪LBL 08
162 FS?C 06
163 GTO 00
164 >" .."
165 AVIEW
166 RCL 00
167 +
168 RCL IND X
169 X<> IND 00
170 STO IND Y
171 X<>Y
172 STO 00
173 1
174 ST+ 13
175 GTO 11
176▪LBL 00
177 >" ??"
178 AVIEW
179 TONE 00
180 GTO 11
181▪LBL 03
182 "R/S?"
183 PROMPT
184 GTO 11
185▪LBL 05
186 X=Y?
187 SF 06
188 X<>Y
189 RDN
190 END

(347 bytes)
```

Registers, Labels and Flags

REGISTERS	COMMENTS
R00	(Alpha)Numerical start char
R01-R09	Char values of jigsaw
R10	Temp storage in LBL06/LBL11
R11	Temp storage base character
R12	Timer period 1-4
R13	Number of moves

LABELS	COMMENTS
LBL00	False move; show and beep
LBL01	Build jigsaw
LBL02	Start of the program
LBL03	Time-out, continue or not?
LBL04	Check all moves feasible
LBL05	Check one move feasible
LBL06	Randomise jigsaw field
LBL07	Show jigsaw and start timer
LBL08	Apply the moves
LBL09	Check jigsaw is completed
LBL10	Show jigsaw
LBL11	Initiate build of jigsaw
LBL12	Not used
LBL13	Not used
LBL14	Handle (key) move “Down”
LBL33	Handle (key) move “Left”
LBL52	Handle (key) move “Right”
LBL61	Handle (key) move “Up”

FLAGS	COMMENTS
00	Generate numerical or alphanumerical jigsaw
01-04	Activation of each of the four time-period intervals (LBL10)
05	Time-out activation
06	Check for duplicates (LBL06). Check for feasible move (LBL05)
25	Ignore “GTO IND X” error if other key than 14, 33, 52, 61 is pressed
29	Set or reset thousand separator (goes with default to FIX 5 in line 91)

Downloads

The RAW/TXT format of the program is available via the website: JIGSAW (in zip file).