

01798D Program Description I

Program Title BRIDGE-IT

Contributor's Name Mondecai Schwartz, M.D.

Address 119 Hewlett Neck Road

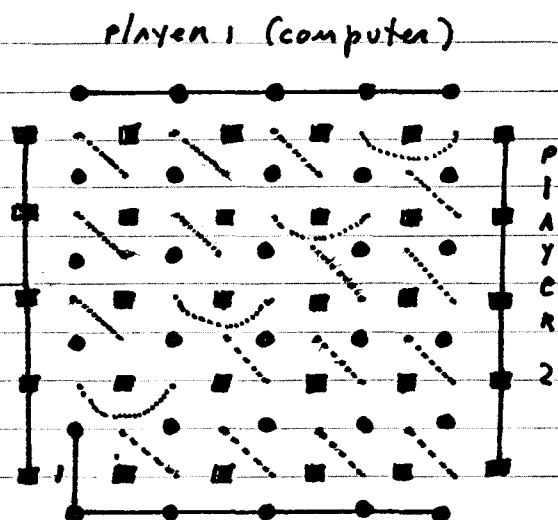
City Woodmere, L.I.

State N.Y.

Zip Code 11598

Program Description, Equations, Variables

Even without computer simulation, the game of BRIDGE-IT (see references) is fun to play, and a real game of skill. It is available commercially. Try playing it first against a human opponent to get a feel for its complexity and depth. Two players compete (use different colors) to build a rectilinear interconnected network between their respective posts (see example). Player 1 connects $\bullet + \bullet$. Player 2, $\blacksquare + \blacksquare$. No bridge may cross an opponent's.



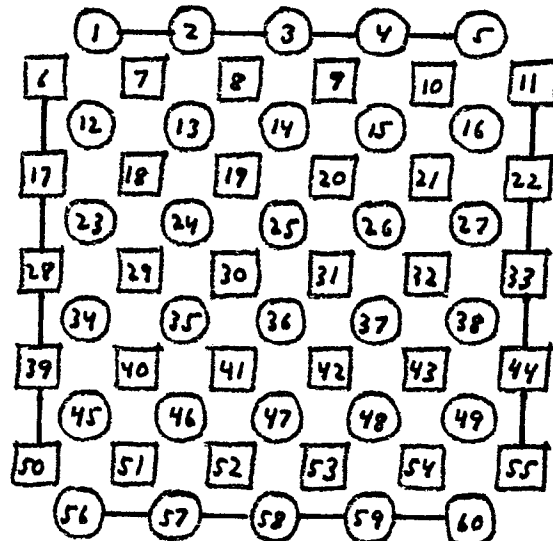
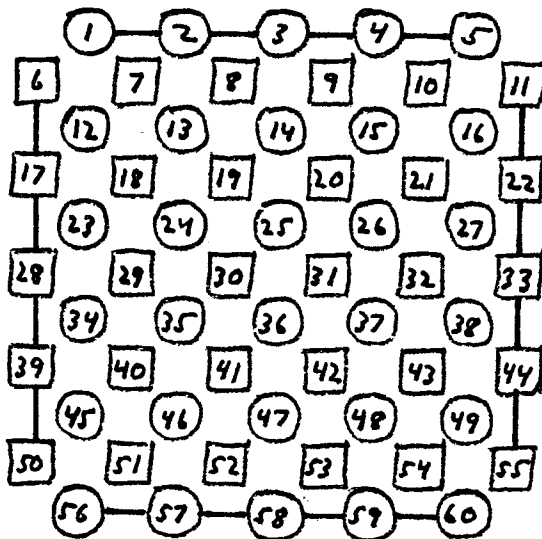
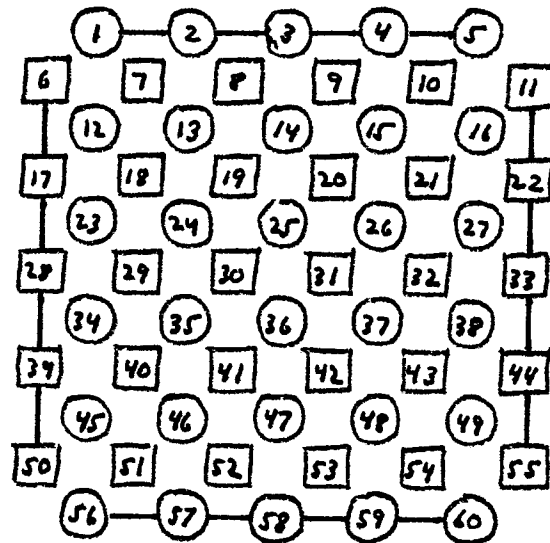
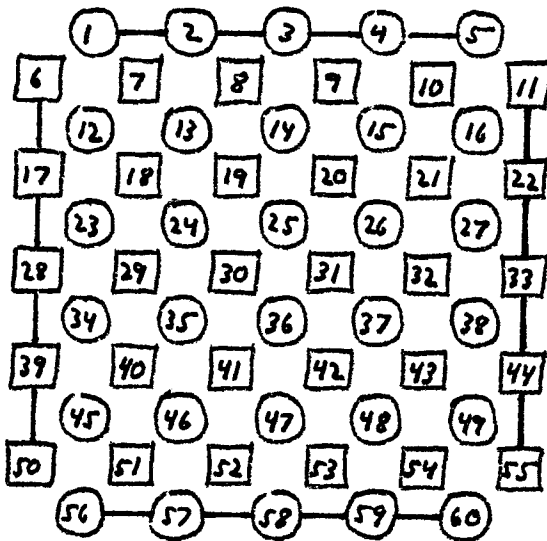
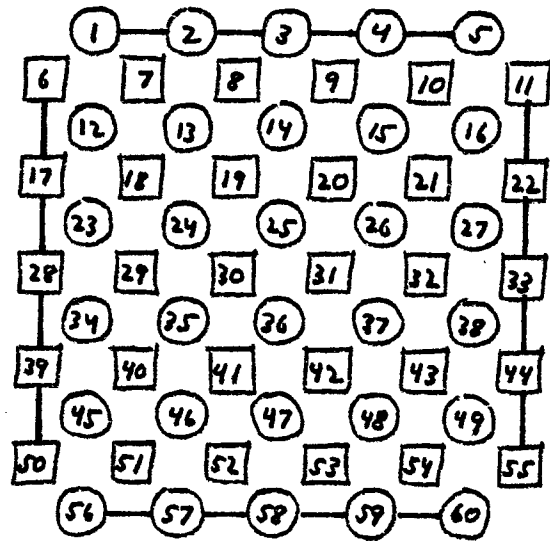
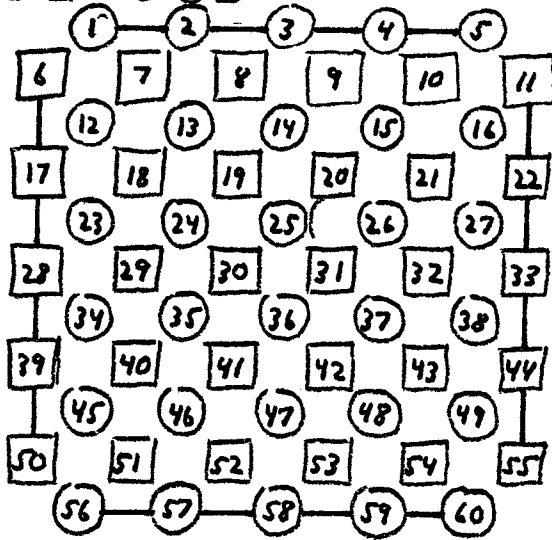
Surprisingly, this apparently complex game was found, by Oliver Gross, a year after its introduction, to have a completely defined and clearly stated strategy forcing a win for player 1. After his initial move (shown), each opponent move is answered by connecting two circles such that the opposite end of the appropriate dotted line bisects the bridge.

Operating Limits and Warnings The program implements this far-from-obvious winning strategy using no arithmetic numbering of the squares and circles. It further encompasses all symmetric variations so that the 6797 has a choice of opening moves and a corresponding variation in response to specific player moves. Play your best and observe the computer strategy - it is most instructive and entertaining. When desired, up to 19 moves may be reviewed.

This program has been verified only with respect to the numerical example given in *Program Description II*. User accepts and uses this program material AT HIS OWN RISK, in reliance solely upon his own inspection of the program material and without reliance upon any representation or description concerning the program material.

NEITHER HP NOR THE CONTRIBUTOR MAKES ANY EXPRESS OR IMPLIED WARRANTY OF ANY KIND WITH REGARD TO THIS PROGRAM MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. NEITHER HP NOR THE CONTRIBUTOR SHALL BE LIABLE FOR INCIDENTAL OR CONSEQUENTIAL DAMAGES IN CONNECTION WITH OR ARISING OUT OF THE FURNISHING, USE OR PERFORMANCE OF THIS PROGRAM MATERIAL.

01798D



01798D Program Description II BRIDG-IT

Page 3 of 6

Sketch(es)	<u>SAMPLE GAME OF BRIDG-IT</u> (NO RNG INITIATING)					
16.05						Moves are listed vertically starting with computer's first move 16.05
21.22	41.42	31.20	51.40	19.30		
26.15	46.35	25.14	45.34	24.13		
21.10	29.30	8.09	40.29	7.08		
15.04	26.35	15.14	34.23	14.13		
				END		Each player's move and computer response is then shown.
31.32	21.32	51.52	17.18			
36.25	38.27	56.45	24.23			

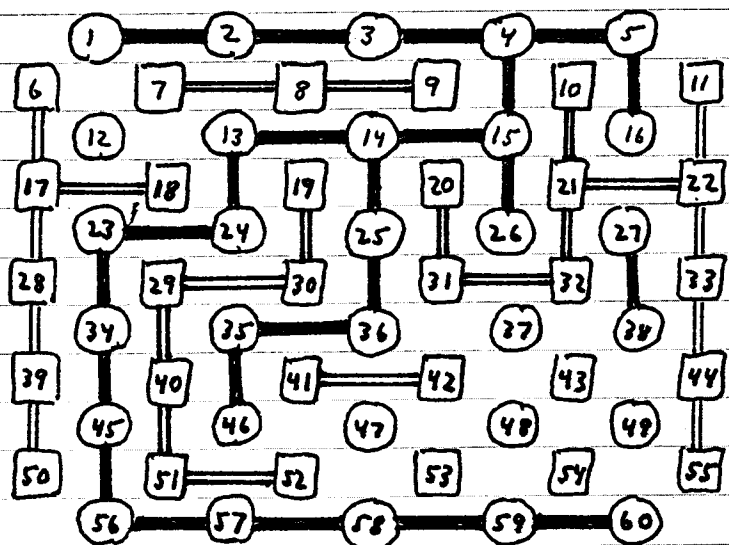
Sample Problem(s)

- (1) Press A to start.
→ 16.05

Computer constructs a bridge between
(5) and (16)

- (2) Player then connects
[21] and [22] and HP-67/97
responds with (26) + (15)
[21.22 C → 26.15]

- (3) Game continues as shown
above + in diagram.



Final position. HP-67/97 (—) wins.

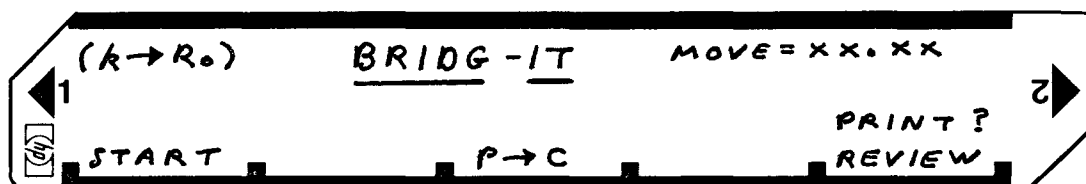
Solution(s)

Move 13 (38.27) is not the best computer response to player's preceding move. The 67/97 plays only as good as it has to to win, generally playing stronger against stronger opposition. Note how the computer eventually forces a winning connection despite strong opposition.

At move 19 you may press E to review the entire game. After this move, only the last 19 moves will be displayed.

Reference(s) Bridg-It was introduced in Scientific American, Oct. 1958 by Martin Gardner as "the game of GALE", and subsequently developed further in his New Mathematical Diversions from Scientific American (Simon & Schuster, 1966). The game was created by DAVID GALE, a mathematician at Brown University.

User Instructions



STEP	INSTRUCTIONS	INPUT DATA/UNITS	KEYS	OUTPUT DATA/UNITS
1	Enter both sides of program card + (optionally) initial RAND	$0 < k < 1$	<input type="text"/> <input type="text"/> STO <input type="text"/> <input type="text"/>	
2	Choose display mode (alternate presses → 1 + 0)		<input type="text"/> <input type="text"/> f <input type="text"/> e	1 = PRINT 0 = NO PRINT
3	START → computer's first move		<input type="text"/> <input type="text"/> A	C_0
4	Player moves + computer replies Moves are entered by indicating — in either order — the two ends of the bridge (DSP 2 format). Eg., either 7.18 or 18.07 (NOT 18.7) designates a bridge between squares 7 and 18. Computer responses are in the same format. The HP-97 will print/format all moves if print mode was chosen.	p_i	<input type="text"/> <input type="text"/> C	p_i, c_i c_i only is displayed if print mode was not chosen
5	Repeat 4 until a winner is determined. The 67/97 does not reject illegal moves or announce a win, but adjusts its level of play to its opponent's strength.		<input type="text"/> <input type="text"/>	
6	At any time, review the last up-to-19 moves if desired. Output is in pause/print mode, but is most useful on the HP-67. The first move displayed is always c_i , and subsequent moves appear in the order played.		<input type="text"/> <input type="text"/> E	c, p, c, \dots
7	For a new game, go to 2 or 3.		<input type="text"/> <input type="text"/>	

Program Listing I BRIDGE-1T

Page 5 of 6

STEP	KEY ENTRY	KEY CODE	COMMENTS	STEP	KEY ENTRY	KEY CODE	COMMENTS
001	F LBLA	31 25 11	<u>START</u>		S	05	
	RCL 0	34 00	zero all registers		h STI	35 33	
	f C/Rg	31 43	EXCEPT RAND.	060	f LBL6	31 25 06	Retain p move
	f P=5	31 42			f DSZ	31 33	+ perform save
	f C/Rg	31 43			f DSZ	31 33	cycle to preserve
	STO 0	33 00	clear symmetry		f DSZ	31 33	a maximum of 19
	h C/F0	35 61 00	flags Fo + Fi		S	05	game moves.
	h C/F1	35 61 01			h RCI	35 34	
	f GSBO	31 22 00			g X=Y	32 51	
010	f X=0	31 51	Then reposition		h GTO 7	22 07	Decompose +
	h SFO	35 51 00	the symmetry flags		RCL 4	34 24	input p = [a _p] [b _p]
	f GSBO	31 22 00	randomly.		f ISZ	31 34	into R ₁ + R ₂ +
	f X=0	31 51			f ISZ	31 34	evolve a response.
	h SFI	35 51 01		070	STO 4	33 24	
	4	04			GTO 6	22 06	
	S	05	obtain trans-		f LBL7	31 25 07	
	STO 1	33 01	formed basnl		RCL 4	34 04	
	S	05	move = computer's		STO 7	33 07	
	6	06	first move.		ENTER ↑	41	
020	STO 2	33 02			f INT	31 83	
	f b	32 22 12			STO 1	33 01	
	g LBLd	32 25 14			-	51	
	RCL 1	34 01	<u>OUTPUT</u>		EEX	43	
	RCL 2	34 02	Form (a _c) (b _c) =	080	2	02	
	EEX	43	computer move.		X	71	
	2	02			STO 2	33 02	
	÷	81	pause/print		f b	32 22 12	
	+	61			f c	32 22 13	
	STO 6	33 06	p = [a _p] [b _p]		f b	32 22 12	
030	h F?2	35 71 02	c = (a _c) (b _c)		GTO fd	22 31 14	
	GTO 5	22 05	if print mode		g LBLb	32 25 12	
	h RTN	35 22	was selected,		h F?0	35 71 00	
	f LBL5	31 25 05	otherwise just		GTO 1	22 01	
	h SFI	35 51 02	output c's move.	090	6	06	
	RCL 7	34 07			1	01	
	h SPACE	35 84			RCL 1	34 01	
	f X=0	31 61			-	51	
	f -X-	31 84			STO 1	33 01	
	h =	35 52			6	06	
040	f -X-	31 84			1	01	
	h RTN	35 22			RCL 2	34 02	
	f LBL0	31 25 00			-	51	
	RCL 0	34 00	<u>RAND</u>		STO 2	33 02	
	h π	35 73	R _i = fract.pmt	100	f LBL1	31 25 01	
	+	61	(R _{i-1} + π) ²		h F?1	35 71 01	
	g X ²	32 54			h RTN	35 22	
	g fract	32 83			RCL 1	34 01	
	STO 0	33 00	Output = 0 on 1		f a	32 22 11	
	2	02	to determine		STO 1	33 01	
050	X	71	Fo + Fi, settings.		RCL 2	34 02	
	f INT	31 83			f a	32 22 11	
	h RTN	35 22			STO 2	33 02	
	f LBLC	31 25 13			h RTN	35 22	
	h CF3	35 61 03		110	g LBLa	32 25 11	
	STO 4	33 04			STO 4	33 04	
	2	02			1	01	

Retain p move
+ perform save
cycle to preserve
a maximum of 19
game moves.

Decompose +
input p = [a_p] [b_p]
into R₁ + R₂ +
evolve a response.

TRANSFORM
{ [a_p] } TRUE
{ [b_p] } MOVE
{ a_p } MOVE
{ b_p } IN
{ a_c } RESPONSE
{ b_c } IN
{ a_c } TRUE
{ b_c } RESPONSE

Then output

TRANSFORM
Symmetry trans-
formation as per
Fo + Fi coding.
61-a → a
61-b → b
for center-point
symmetry.

Horizontal line
symmetry

f-a → a
f-b → b
f = transposition
factor

FACTOR f

REGISTERS									
0	RAND	1 EVOLVING a _i	2 EVOLVING b _i	3 m _f	4 hold applied to	5 -	6 (a _c) (b _c)	7 [a _p] [b _p]	8 c
9									p
S0	c	S1 p	S2 c	S3 p	S4 c	S5 p	S6 c	S7 p	S8 c
									S9 p
A	c	B p	C c	D p	E c	I	index		

STEP	KEY ENTRY	KEY CODE	COMMENTS	STEP	KEY ENTRY	KEY CODE	COMMENTS
	1	01	First obtain + store m_f $m_f = a \text{ mod. } 11$ but $m_f + 11 \rightarrow m_f$ if orig. $m_f = 0$ make sure m_f is integral.		9 $x \leq y$	32 71	if $(b_p - a_p) = 1$ do horizontal move analysis (LBL 2): $R_4 = 0 \rightarrow \text{LBL 3}$ $R_4 = 9 \rightarrow \text{LBL 4}$ but if $a_p \text{ mod. } 10$ $\neq 0 + \neq 9$ $a_p \pm 6 \rightarrow a_c$ $b_p \pm 6 \rightarrow b_c$ as per actual mod. value.
	\div	81			CHS	42	
	g FRACT	32 83		170	STO + 1	33 61 01	
	1	01			STO + 2	33 61 02	
	1	01			h RTN	35 22	
	x	71			f LBL 2	31 25 02	
	f RND	31 24			9	09	
120	1	01			RCL 4	34 04	
	1	01			f $x = 0$	31 51	
	h \leq	35 52			GTO 3	22 03	
	f $x \neq 0$	31 61	----- $(m_f \leq 5) \rightarrow$ $f = 55 + 2m_f$ $(m_f > 5) \rightarrow$ $f = 44 + 2m_f$ Then f is used for $f - a \rightarrow a$ $f - b \rightarrow b$ in horiz. symmetry ($f = 61$ for point symm.)	180	9 $x = y$	32 51	ap mod. 10 = 0 \rightarrow $a_p - 6 \rightarrow a_c$ $b_p + 4 \rightarrow b_c$ ap mod. 10 = 9 \rightarrow $a_p - 4 \rightarrow a_c$ $b_p + 6 \rightarrow b_c$
	0	00			GTO 4	22 04	
	+	61			6	06	
	STO 3	33 03			9 $x > y$	32 81	
	5	05			CHS	42	
	h \leq	35 52			STO + 1	33 61 01	
	9 $x \leq y$	32 71			STO + 2	33 61 02	
130	h SF 3	35 51 03			h RTN	35 22	
	4	04			f LBL 3	31 25 03	
	h F? 3	35 71 03			6	06	
	5	05	RESPONSE First order $a_p + b_p$ so that $a_p < b_p$ ----- Then obtain + store $a_p \text{ mod. } 10$ + use it for ...	190	STO - 1	33 51 01	REVIEW Pause/print the game — up to 19 moves if $n > 19$, the last 19 moves are outputted. First move shown is always c/a.
	ENTER P	41			4	04	
	1	01			STO + 2	33 61 02	
	1	01			h RTN	35 22	
	x	71			f LBL 4	31 25 04	
	RCL 3	34 03			4	04	
140	2	02			STO - 1	33 51 01	
	x	71			6	06	
	+	61			STO + 2	33 61 02	
	RCL 4	34 04			h RTN	35 22	
	-	51	VERTICAL MOVE ANALYSIS: $a_p \pm 6 \rightarrow a_c$ $b_p \pm 6 \rightarrow b_c$	200	f LBL 5	31 25 15	PRINT? $F_{\text{on}} \rightarrow 1 = \text{print}$ $F_{\text{off}} \rightarrow 0 = \text{no}$ 1 + 0 alternate.
	h RTN	35 22			h SPACE	35 84	
	g LBL C	32 25 13			2	02	
	ORCL 1	34 01			4	04	
	RCL 2	34 02			h STI	35 33	
	9 $x \leq y$	32 71			f LBL 8	31 25 08	
150	h \leq	35 52			RCL 4	34 04	
	STO 2	33 02			f $x \neq 0$	31 61	
	h \div	35 53			f $x =$	31 84	
	STO 1	33 01			f DSZ	31 33	
	1	01	Labels	210	6	06	Flags
	0	00			h RCI	35 34	
	\div	81			9 $x \neq y$	32 61	
	g FRACT	32 83			GTO 8	22 08	
	1	01			RCL 6	34 06	
	0	00			f $x =$	31 84	
	x	71			h RTN	35 22	
160	STO 4	33 04			g LBL 6	32 25 15	
	RCL 2	34 02			h F? 2	35 71 02	
	RCL 1	34 01			GTO 5	22 05	
	-	51	SET STATUS		h SF 2	35 51 02	Trig
	1	01			1	01	
	9 $x = y$	32 51		220	h RTN	35 22	
	GTO 2	22 02			f LBL 5	31 25 05	
	RCL 4	34 04			0	00	
	6	06			h RTN	35 22	

LABELS					FLAGS		SET STATUS		
A START	B -	C P \rightarrow C	D -	E REVIEW	0 SYMMETRY	ON OFF 0 <input type="checkbox"/> <input checked="" type="checkbox"/> 1 <input type="checkbox"/> <input checked="" type="checkbox"/> 2 <input type="checkbox"/> <input checked="" type="checkbox"/> 3 <input type="checkbox"/> <input checked="" type="checkbox"/>	TRIG		DISP
a FACTOR	b TRANSFORM	c RESPOND	d OUTPUT	e PRINT ?	1 VARIANTS		DEG <input checked="" type="checkbox"/>	FIX <input checked="" type="checkbox"/>	
0 RAND	1 LINE SYMMETRY	2 HORIZ. MOVE	3 $m = 0$	4 $m = 9$	2 PRINT		GRAD <input type="checkbox"/>	SCI <input type="checkbox"/>	
5 PRINT (MULTIPLE)	6 SAVE CYCLE	7 INPUT	8 REVIEW CYCLE	9 -	3 m_f value		RAD <input type="checkbox"/>	ENG <input type="checkbox"/>	n 2