

Program Description I

Program Title 67-FORTY-FOUR: A GAME OF DEDUCTION

Contributor's Name LARRY DON COLTON

Address 618 EAST 500 NORTH

City PROVO

State UTAH

Zip Code 84601

Program Description, Equations, Variables Program generates and stores in secondary memory a pseudo-random number. Each guess entered by a player (by pressing "E") is compared with the target number, and a count is produced and placed as an exponent to "1.guess" which is then re-displayed to the player (as a clue).

The number chosen by the calculator is generated one digit at a time, and stored in secondary registers S1 through S4 (one for each digit). Digits are slightly disguised.

RANDOM DIGITS are generated by using the modulo 997 generator. Each time a digit is generated, a new seed is stored in memory where the old seed had been. This will be called "incrementing the seed." ($seed_1$ is replaced by $seed_{i+1}$.)

Each time "C" is pressed, the seed is incremented once. Each time "B" is pressed, a whole new number is stored in S1 through S4, and the seed is incremented four times. Further, each time a guess is entered, the seed is incremented, ensuring that after the first number (6678) subsequent numbers are fairly unpredictable.

Each time "A" (start) is pressed, the seed .7058031 is stored in register C. To start with a different number, simply input any other fairly complex seed between zero and one. ("Fairly complex" means more than three digits after the decimal.) For example, the seed .5284136 input by pressing "D" (input seed) yields the number 8520 rather than the number 6678 which is obtained by pressing "A" (start).

Operating Limits and Warnings WARNING: Repeating digits cause problems in making deductions. Example: Suppose the number is really 0005. You guess 0123. Count is 11. Since you only offered one zero, you are not told of the other two. You guess 0011. Count is 22, since two zeros were offered and at least two were found. You guess 0000. Count is 33, since of four zeros offered, three matched up. **DUPLICATED DIGITS** as in 0005 and 6678 occur frequently (49.6% of the time), and should therefore NOT be assumed away (as is often tempting). This offers additional challenge to FORTY-FOUR players.

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.

Sketch(es) Sample Calculator Display:

1.5678

44

Used for
exponent
stabilityGuess as entered by
player.

Number of Correct Digits

Number of Correctly Located Digits

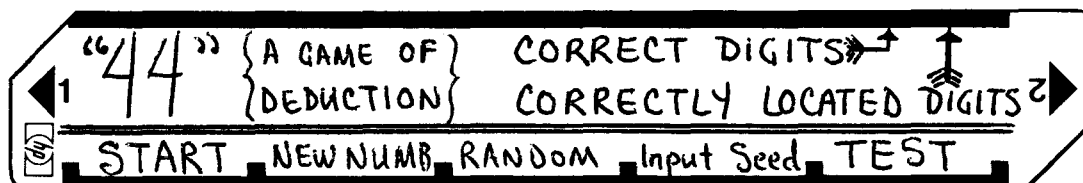
Sample Problem(s) Lead the program (both sides) and press "A" (start). Calculator responds with "0.0000 00" display. A number is safely tucked away, and we must guess it.

Enter 1234 and press "E" (test). 1234 is our first guess. After 15 seconds or so, the calculator responds with "1.1234 00" in the display. 00 is the count. It tells us that no 1, 2, 3, or 4 appears in the number we are trying to deduce. (I.e. zero correct digits and zero in the right places.) For our second guess, enter 5678 and press "E" (test). After another 15 seconds (we also try to teach patience) the display is "1.5678 33" (We may as well tell you now: The "1" before the decimal point serves no other purpose than to keep the exponent where we want it. You can ignore it. This tells us that of the digits 5, 6, 7, and 8, three occur in our target number, and they are correctly positioned in our guess. One number is still wrong. For our third guess enter 1678 "E" The display again reads "1.1678 33" We know that 1 is not in our target number, so therefore the correct digits must be 6, 7, and 8, located right where they are! For the fourth try (let's do the right one) enter 6678 "E" We

Solution(s) should get the following displays: first "1.6678 44" for five seconds, then second "4." for another five seconds (it means we got it on the fourth try), and last, after some slight delay, "0.0000 00" indicating a new number awaits us. Before making a guess, press RCL 1, RCL 2, RCL 3, and RCL 4. You should see a record of your guessing. This is always available to help you figure out the target number. You may prefer to keep track on paper also. Now, the next number is 7230 BUT DON'T ENTER IT YET! Make a few wrong guesses first. With the knowledge you have of the number, you should be able to become quite familiar with the interpretation of the "COUNT" displayed after each guess. When you think you have it down well, enter 7230. And Happy Deducing! LDC

Reference(s) This game has appeared in several forms, with colors instead of digits (MASTERMIND game by INVICTA), and with letters of the alphabet (JOTTO by SELCHOW and RIGHTER, makers of SCRABBLE). My first acquaintance was with "OX" a game using O's and X's to represent the count. "OX" was on an Altair 680 mini-computer (with CRT) owned by a friend. At that time I became interested in putting it on my HP-67. LDC

User Instructions



STEP	INSTRUCTIONS	INPUT DATA/UNITS	KEYS	OUTPUT DATA/UNITS
1	Load side 1 and side 2.		<input type="text"/> <input type="text"/>	0.0000 00
2	Input seed by either of the following methods:		<input type="text"/> <input type="text"/>	
	(A) Use the seed provided at "A" (start)		A <input type="text"/>	0.0000 00
	(B) Provide a seed between zero and one	.nnnnnnnn	D <input type="text"/>	0.0000 00
	Your number is now ready to be guessed.		<input type="text"/> <input type="text"/>	
3	Enter your four-digit guess	nnnn	E <input type="text"/>	1.nnnn cc
	The exponent ("cc") part of the calculator response is the COUNT for your guess.		<input type="text"/> <input type="text"/>	
4	Continue guessing (step 3) until you finally get the right number. Say it was wwwwww	wwwwww	E <input type="text"/>	1.wwww 44
			<input type="text"/> <input type="text"/>	n.
			<input type="text"/> <input type="text"/>	0.0000 00
	Right while you watch, n (the number of guesses you entered) is displayed 5 sec. Then a new number is generated for you, and you are tempted to go back to step 3.		<input type="text"/> <input type="text"/>	
			<input type="text"/> <input type="text"/>	
	NOTE: There is never any need to use steps 5 and 6. They are used by the program, and simply made available to users who might find them useful (as has the author).		<input type="text"/> <input type="text"/>	
			<input type="text"/> <input type="text"/>	
5	In order to get a completely new number to work on (say for example that you know the first one is 6678 anyway) just press "B"		B <input type="text"/>	0.0000 00
			<input type="text"/> <input type="text"/>	
6	If you have any use for a random digit (between zero and nine) press "C"		C <input type="text"/>	r.0000 00
	"r" is your random digit.		<input type="text"/> <input type="text"/>	
			<input type="text"/> <input type="text"/>	
7	To review a past guess and its count, determine what number guess it was, and use the last digit of that number as follows:		<input type="text"/> <input type="text"/>	
	EXAMPLE: say the number is 9		RCL 9	1.nnnn cc
	EXAMPLE: say the number is 4		RCL 4	1.nnnn cc
	EXAMPLE: say it was the 20th guess		RCL 0	1.nnnn cc
	EXAMPLE: say it was the 11th guess		RCL 1	1.nnnn cc
	Note that only the most recent ten guesses are available in the manner described.		<input type="text"/> <input type="text"/>	
			<input type="text"/> <input type="text"/>	

STEP KEY ENTRY KEY CODE

001 *LBL5 21 14
 002 ST00 35 13
 003 ST05 35 12
 004 *LBLA 21 11
 005 . -62
 006 7 07
 007 0 00
 008 5 05
 009 3 06
 010 0 00
 011 3 03
 012 1 01
 013 ST00 35 13
 014 *LBL5 21 12
 015 F25 16-51
 016 CLX -51
 017 ST00 35 00
 018 G8B0 23 13
 019 ST01 35 01
 020 G8B0 23 13
 021 ST02 35 02
 022 G8B0 23 13
 023 ST03 35 03
 024 G8B0 23 13
 025 ST04 35 04
 026 1 01
 027 ST01 35 46
 028 G8B0 23 00
 029 F25 16-51
 030 RTN 24
 031 *LBL0 21 13
 032 R0L0 35 13
 033 9 05
 034 3 03
 035 7 07
 036 A -35
 037 FRC 16 44
 038 ST00 35 13
 039 1 01
 040 0 00
 041 X -35
 042 INT 16 34
 043 RTN 24
 044 *LBL5 21 15
 045 F25 16-51
 046 ST05 35 05
 047 G8B0 23 13
 048 R0L0 35 00
 049 X=0? 16-43
 050 G8B4 23 04
 051 R0L5 35 05
 052 1 01
 053 ST+0 35-55 00
 054 ST05 35 05
 055 XZY -41
 056 SEV -23

COMMENTS

Input User's
Random Seed --

 Input Author's
 Random Seed
 (which is:
 .7058031)

 Generate a four-
 -digit "target"
 number to be guessed
 Zero out guess count

 store first rand dgt
 store 2nd rand digit
 store 3rd rand digit
 store 4th rand digit
 increment each digit

 return
 RANDOM DIGIT ROUTINE
 recall seed

 "increment the seed"

 produce random digit

 increment the seed
 on the first
 guess, we zero
 out guess rcrd.

 # of guesses:
 increment count

 Bring back the

STEP KEY ENTRY KEY CODE

057 4 04
 058 + -24
 059 ST+5 35-55 05
 060 1 01
 061 0 00
 062 X -35
 063 INT 16 34
 064 ST06 35 06
 065 LSTX 16-63
 066 XZY -41
 067 - -45
 068 1 01
 069 0 00
 070 X -35
 071 INT 16 34
 072 ST07 35 07
 073 LSTX 16-63
 074 XZY -41
 075 - -45
 076 1 01
 077 0 00
 078 X -35
 079 INT 16 34
 080 ST08 35 08
 081 LSTX 16-63
 082 XZY -41
 083 - -45
 084 1 01
 085 0 00
 086 X -35
 087 INT 16 34
 088 ST09 35 09
 089 0 00
 090 ST03 35 46
 091 G8B0 23 00
 092 R0L1 35 01
 093 G8B1 23 01
 094 R0L2 35 02
 095 G8B1 23 01
 096 R0L3 35 03
 097 G8B1 23 01
 098 R0L4 35 04
 099 G8B1 23 01
 100 R0L1 35 01
 101 R0L6 35 06
 102 ABB 16 31
 103 X=V0 16-33
 104 G8B3 23 03
 105 R0L2 35 02
 106 R0L7 35 07
 107 ABB 16 31
 108 X=V7 16-33
 109 G8B3 23 03
 110 R0L3 35 03
 111 R0L8 35 08
 112 ABB 16 31

COMMENTS

guess

 pull off the 1st
 digit and store
 it in reg. 6

 pull off 2nd digit
 put in reg. 7

 pull off 3rd digit
 put in reg. 8

 pull off 4th digit
 put in reg. 9

 increment each digit

 was dgt 1 guessed?
 was dgt 2 guessed?
 was dgt 3 guessed?
 was dgt 4 guessed?

 is dgt 1 exactly
 right?
 is dgt 2 exactly
 right?
 is dgt 3 exactly
 right?

REGISTERS

0 User's 10th gue	1 User's 1st guess	2 User's 2nd guess	3 User's 3rd guess	4 User's 4th guess	5 User's 5th guess	6 User's 6th guess	7 User's 7th guess	8 User's 8th guess	9 User's 9th guess
S0 Current guess #	S1 Target digit1 +1	S2 Target digit2 +1	S3 Target digit3 +1	S4 Target digit4 +1	S5 Count preparath	S6 Guess digit1 +1	S7 Guess digit2 +1	S8 Guess digit3 +1	S9 Guess digit4 +1
A unused cleared often	B unused cleared often	C random seed	D unused cleared often	E unused cleared often	F unused cleared often	G unused cleared often	H unused cleared often	I indexing for guess storage	

STEP	KEY ENTRY	KEY CODE	COMMENTS	STEP	KEY ENTRY	KEY CODE	COMMENTS
113	N=Y?	16-35		169	GSE3	23 02	
114	GSE3	23 03		170	F00	16 23 00	does it match
115	RCL4	36 04		171	RTN	24	guess dgt 3?
116	RCL9	36 09		172	GSE2	23 02	does it match
117	HE9	16 31	is dgt 4 exact-	173	RTN	24	guess dgt 4?
118	N=Y?	16-35	ly right?	174	*LBLE2	21 02	
119	GSE3	23 03		175	ISZ1	16 26 45	given a digit
120	RCL9	36 08		176	RCL1	35 45	from the target
121	.	-62	fix up the	177	XZY	-41	number, does it
122	1	01	index (guess #	178	N=Y?	16-32	NOT match this
123	4	-35	modulo 10) for	179	RTN	24	digit in guess?
124	FRC	16 44	storing this	180	1	01	
125	1	01	guess & count	181	DR3	-22	If it did match,
126	0	00		182	STW1	35-35 45	we flag guess
127	X	-35		183	SFO	16 21 00	digit
128	STOI	35 46		184	EEN	-23	
129	EE1	-23	create a stan-	185	1	01	and we set flag
130	4	04	dard for test-	186	0	00	to quit looking
131	4	04	ing completion	187	STW5	35-35 05	for another mch
132	RCL5	36 05		188	RTN	24	with target dgt.
133	PZ8	16-51		189	*LBLE3	21 03	& increm. count!
134	STOI	35 45	store the guess	190	1	01	
135	N=Y?	16-35	test completion	191	3	00	Increment count
136	RTN	24		192	STW5	35-35 05	by 1 in the
137	PZ5	16-51	5 sec. display	193	RTN	24	"number correct
138	PRTX	-14	5 sec. display	194	*LBLE4	21 04	ly located" dgt
139	FIX	-11	of # of guesses	195	PZ8	16-51	
140	GSP0	-63 00	entered up to	196	RCLC	36 13	saving the seed,
141	RCL0	36 00	now.	197	CLRG	16-53	zero out all the
142	PRTX	-14		198	STOC	35 13	rest of primary
143	SCI	-12		199	PZ6	16-51	memory.
144	DSP4	-63 04	get a new	200	RTN	24	Restore seed.
145	PZ5	16-51	random number				
146	STOE	22 12					
147	*LBLE0	21 00	increment four				
148	CLX	-51	memories in a				
149	ISZ1	16 26 45	row, starting				
150	ISZ1	16 26 46	with memory "1"				
151	ISZ1	16 26 45	and ending at				
152	ISZ1	16 26 46	memory "1+3"				
153	ISZ1	16 26 45	(allows flagging	210			
154	ISZ1	16 26 46	of zeros)				
155	ISZ1	16 26 45					
156	RTN	24	digit comes from				
157	*LBLE1	21 01	target number...				
158	CF0	16 22 00	starting at mem-				
159	5	05	ory 5...				
160	STOI	35 46					
161	CLX	-51					
162	+	-55					
163	GSE2	23 02	does it match	220			
164	F00	16 23 00	guess dgt 1?				
165	RTN	24					
166	GSE2	23 02	does it match				
167	F00	16 23 00	guess dgt 2?				
168	RTN	24					

LABELS					FLAGS	SET STATUS			
A Start	B New Number	C Random Digit	D Input Seed	E TEST	0 Do digits match?	FLAGS		TRIG	DISP
a unused	b unused	c unused	d unused	e unused	1 unused	0 ON OFF		DEG <input checked="" type="checkbox"/>	FIX <input type="checkbox"/>
0 add 1 to each digit	1 check for given digit	2 compare given digits	3 if correct located...	4 clear 0-9 A, B, D, & E	2 unused	1 <input type="checkbox"/> <input checked="" type="checkbox"/>		GRAD <input type="checkbox"/>	SCI <input checked="" type="checkbox"/>
5 unused	6 unused	7 unused	8 unused	9 unused	3 unused	2 <input type="checkbox"/> <input checked="" type="checkbox"/>		RAD <input type="checkbox"/>	ENG <input type="checkbox"/>
						3 <input type="checkbox"/> <input checked="" type="checkbox"/>			n 4