00325D

Program Description 1

Program Title CYBERNETIC NIM GAME

Contributor's Name JOHN RAUSCH

Address 402 VIRGINIA AVE.

City FRANKLIN **State** OHIO **Zip Code** 45005

Program Description, Equations, Variables CYBERNETIC NIM IS A SUBTRACTION GAME THAT YOU PLAY AGAINST THE CALCULATOR. WHEN YOU BEGIN, THE CALCULATOR KNOWS ONLY THE RULES OF THE GAME. NO PLAYING STRATEGY IS PROGRAMMED. HOWEVER, EACH TIME YOU WIN, THE CALCULATOR REMEMBERS THE MOVE THAT CAUSED IT TO LOSE AND WILL NOT MAKE THAT SAME MOVE AGAIN. EVENTUALLY THE CALCULATOR WILL PLAY A PERFECT GAME. THE GAME IS PLAYED AS FOLLOWS: YOU BEGIN WITH A NUMBER FROM 5 TO 21. ON EACH TURN YOU OR THE CALCULATOR CAN SUBTRACT 1, 2, OR 3 FROM THE NUMBER. THIS CONTINUES UNTIL SOMEONE MUST TAKE THE LAST 1 AND MAKE THE NUMBER 0. WHOEVER HAS TO DO THIS LOSES. GRANTED, THIS IS A VERY SIMPLE GAME. HOWEVER, THE PURPOSE HERE IS TO DEMONSTRATE THE WAY A COMPUTER CAN BE PROGRAMMED TO "LEARN." THE FIRST TIME YOU PLAY CHOOSE A SMALL NUMBER SO THE LEARNING PROCESS IS SHORT AND IS IMMEDIATELY APPARENT. IT IS INTERESTING TO PLOT THE WON/LOST RECORD WHILE PLAYING DUMB AND THEN PLAY YOUR BEST. THE BETTER YOU PLAY, THE FASTER THE LEARNING PROCESS.

Operating Limits and Warnings STEPS 9 & 10 LIMIT THE STARTING VALUE TO A NUMBER OF 21 OR LESS. THE PROGRAM WILL FUNCTION WITH A VALUE NOT GREATER THAN 33 BUT THE LEANING PROCESS IS VERY LONG (60+ GAMES MAYBE). IF YOU WOULD WANT TO CHANGE THESE STEPS, THE PROGRAM WILL RUN FINE.

00325D

Program Description 11

Sketch(es)

Sample Problem(s) IT IS IMPOSSIBLE TO DEMONSTRATE THE LEARNING PROCESS IN THIS MUCH SPACE. HOWEVER, A SAMPLE GAME IS SHOWN TO ILLUSTRTE THE OUTPUTS. A RANDOM NUMBER SEED IS NOT GENERATED SO THE SAMPLE CAN BE DUPLICATED. THE GAME WILL BE PLAYED WITH A STARTING VALUE OF 15. MAKE SURE REGISTER A IS ZERO (RANDOM NUMBER SEED).

Solution(s) 1) 15 [B] → 15

2) YOU TAKE 2: 2 [C] → 13, -1, 12 (LEAVING 13, CALC TAKES 1 LEAVING 12)

3) YOU TAKE 1: 1 [C] → 11, -2, 9 (LEAVING 11, CALC TAKES 2 LEAVING 9)

4) YOU TAKE 3: 3 [C] → 6, -3, 3 (LEAVING 6, CALC TAKES 3 LEAVING 3)

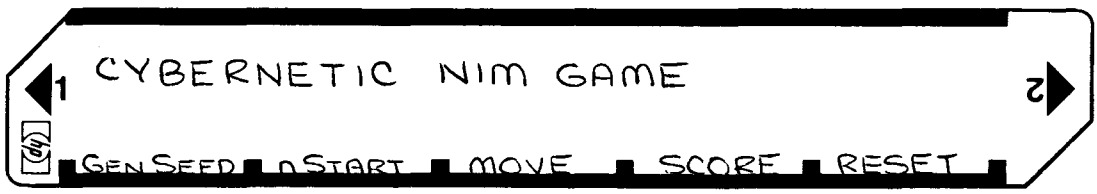
5) YOU TAKE 2: 2 [C] → 1, -1, 0, 15 (LEAVING 1, CALC TAKES 1 LEAVING 0 AND LOSES. START AGAIN WITH 15)

6) PRINT SCORE: [D] → 1, 0, 15 (YOU LEAD 1 TO 0. DISPLAY RETURNS TO 15)

Reference(s) 65 NOTES, VOL 2 NO 4 PUBLISHED BY HP-65 USERS CLUB, 2541 W. CAMDEN PLACE, SANTA ANA, CALIFORNIA 92704. CYBERNIM PROGRAM BY T. WIRTZ BASED ON HEXAPAWN PROGRAM BY J. RAUSCH.

User Instructions

00325D



STEP	INSTRUCTIONS	INPUT DATA/UNITS	KEYS	OUTPUT DATA/UNITS
1	LOAD SIDE 1 AND SIDE 2.			
2	GENERATE RANDOM NUMBER SEED. AFTER PRESSING [A] ALLOW THE PROGRAM TO RUN A FEW SECONDS, THEN PRESS [R/S]. YOU CAN OPTIONALLY STORE ANY NUMBER IN REGISTER A.	n	A R/S -OR- STO A	? n
3	SELECT A STARTING VALUE FROM 5 TO 21	5 TO 21	B	VALUE
4	MOVE. IF YOU WANT THE CALCULATOR TO MOVE FIRST JUST PRESS [C]. THIS IS PERMITTED ONLY ON THE FIRST MOVE OF A GAME. IF YOU WANT TO MOVE, ENTER 1, 2, OR 3 AND PRESS [C]. THE FOLLOWING WILL BE DISPLAYED: 1) THE RESULT OF YOUR MOVE. 2) THE CALCULATORS MOVE WITH A MINUS SIGN TO IDENTIFY IT. 3) THE RESULT OF THE CALCULATORS MOVE. IF EITHER YOU OR THE CALCULATOR TAKE THE LAST ONE, THE STARTING VALUE WILL BE DISPLAYED FOR A NEW GAME RIGHT AFTER THE ZERO RESULT. REPEAT STEP 4 ALL YOU WANT.	FIRST MOVE OF GAME ↓ NOTHING -OR- 1, 2, 3	C OR C	SEE TEXT
5	DISPLAY SCORE		D	YOUR WINS/LOSSES
6	RESET -- THIS ALLOWS YOU TO RE-ENTER A STARTING VALUE. NORMALLY [B] IS PROTECTED SO YOU WON'T WIPE OUT MEMORY.		E	
	NOTE: AFTER COMPLETING A GAME YOU CAN WRITE THE CONTENTS OF MEMORY TO A CARD AND START UP LATER WHERE YOU LEFT OFF. DO THE FOLLOWING: ISSUE A WRITE/DATA COMMAND FROM THE KEYBOARD, THEN WRITE A DATA CARD. TO RESTART: COMPLETE STEPS 1 THRU 3 USING THE SAME VALUE. READ IN THE DATA CARD.			

Program Listing I

00325D

STEP	KEY ENTRY	KEY CODE	COMMENTS
001	LBL A	2111	LOOP TO GENERATE RANDOM NUMBER SEED.
	f π	16-24	
	+	-55	
	STO A	3511	
	GTO A	2211	
	LBL B	2112	START. VALIDATES STARTING VALUE. ESTABLISHES MOVE CUES BY RAISING 2 TO THE POWER OF THE STARTING VALUE AND REMOVING THE MOVES THAT ARE NOT PERMITTED FOR A GIVEN VALUE. FOR EXAMPLE, YOU CAN'T TAKE 2 WHEN THERE IS ONLY ONE LEFT.
	f F? 0	1623 00	
	GTO fe	22 16 15	
	2	02	
010	2	02	
	f X≤Y	16-35	
	GTO e	22 16 15	
	CLX	-51	
	5	05	
	f X>Y	16-34	
	GTO e	22 16 15	
	f STF 0	16 21 00	
	f CLREG	16-53	
	CLX	-51	
020	2	02	
	X≷Y	-41	
	STO B	3512	
	Y^x	31	
	2	02	
	-	-45	
	STO 1	3501	
	2	02	
	-	-45	
	STO 2	3502	
030	4	04	
	-	-45	
	STO 3	3503	
	LBL 0	2100	
	f CLF 1	16 22 01	
	f CLF 3	16 22 03	
	RCL B	3612	
	STO 0	3500	
	f SPACE	16-11	
	PRINT X	-14	
040	RTN	24	
	LBL C	2113	MOVE. IF DATA FLAG IS NOT ON, GO TO CALCULATORS MOVE. VALIDATE MOVE. THEN SUBTRACT FROM VALUE AND PRINT RESULT. IF 0, GO TO LOSE.
	f F? 3	1623 03	
	f F? 3	1623 03	
	GTO 1	2201	
	f X>0	16-44	
	f X=0	16-43	
	GTO fe	22 16 15	
	3	03	
	X≷Y	-41	
050	f X>Y	16-35	
	GTO fe	22 16 15	
	RCL 0	3600	
	X≷Y	-41	
	f X>Y	16-35	
	GTO fe	22 16 15	
	-	-45	

STEP	KEY ENTRY	KEY CODE	COMMENTS
	PRINT X	-14	
	f X=0	16-43	
	GTO 5	2205	
060	STO 0	3500	
	GTO 2	2202	
	LBL 1	2101	
	f F? 1	1623 01	CALCULATORS MOVE. IF NOT FIRST MOVE IN GAME, ERROR. INDICATE NOT FIRST MOVE. GENERATE RANDOM MOVE FROM 1 TO 3. SET LOOP FOR 3 ITERATIONS
	GTO fe	22 16 15	
	LBL 2	2102	
	f STF 1	16 21 01	
	RCL A	3611	
	f π	16-24	
	+	-55	
070	X^2	53	
	f FRAC	16 44	
	STO A	3511	
	3	03	
	X	-35	
	f INT	16 34	
	1	01	
	+	-55	
	STO 4	3504	
	2	02	
080	RCL 0	3600	
	Y^x	31	
	STO 5	3505	
	3	03	
	STO I	3546	
	LBL 3	2103	
	3	03	LOOP. WILL TRY ALL THREE POSSIBILITIES. 2 IS RAISED TO THE POWER OF THE CURRENT VALUE AND DIVIDED INTO THE CUES FOR THE MOVE. IF THE RESULT HAS A FRACTIONAL PART ≥ .5 THE MOVE HAS NOT PREVIOUSLY CAUSED A LOSS AND IT IS MADE. IF NO MOVES ARE FOUND THAT WON'T CAUSE A LOSS, THE PRIOR MOVE THAT PUT THINGS HERE IS REMOVED FROM ITS CUE, AND THE MOVE IS MADE 1 FOR NOW.
	RCL 4	3604	
	1	01	
	+	-55	
090	f X>Y	16-34	
	1	01	
	STO 4	3504	
	f X≷I	16-41	
	RCL (i)	3545	
	X≷Y	-41	
	f X≷I	16-41	
	X≷Y	-41	
	RCL 5	3605	
	÷	-24	
100	f FRAC	1644	
	.	-62	
	5	05	
	f X≤Y	16-35	
	GTO 4	2204	
	f DSZ I	16 25 46	
	GTO 3	2203	
	RCL 6	3606	
	STO I	3546	
	RCL 7	3607	
110	2	02	
	÷	-24	
	STO-(i)	35-4545	

REGISTERS

0 CURRENT VALUE	1 CUES FOR MOVE 1	2 CUES FOR MOVE 2	3 CUES FOR MOVE 3	4 CURRENT MOVE	5 CURRENT 2^VALUE	6 PRIOR MOVE	7 PRIOR 2^VALUE	8 YOUR WINS	9 YOUR LOSSES
S0	S1	S2	S3	S4	S5	S6	S7	S8	S9

A RANDOM NUMBER SEED	B VALUE	C	D	E	I USED

Program Listing II

00325D

STEP	KEY ENTRY	KEY CODE	COMMENTS
	CLX	-51	
	STO 5	3505	
	1	01	
	STO 4	3504	
*	LBL 4	2104	MAKE MOVE. SAVE
	RCL 5	3605	IN CASE LATER RE-
	STO 7	3507	MOVAL IS REQUIRED.
120	RCL 4	3604	
	STO 6	3506	
	STO-0	35-4500	
	CHS	-22	
	PRINT X	-14	
	RCL 0	3600	
	PRINT X	-14	
	f X≠0	16-42	
	RTN	24	
	1	01	
130	STO+8	35-5508	
	GTO 0	2200	
*	LBL 5	2105	
	1	01	
	STO+9	35-5501	
	GTO 0	2200	
*	LBL D	2114	DISPLAY SCORE
	RCL 8	3608	
	PRINT X	-14	
	RCL 9	3609	
140	PRINT X	-14	
	RCL 0	3600	
	RTN	24	
*	LBL E	2115	RESET
	f CLFO	162200	
	RTN	24	
150			
160			
170			
180			
190			
200			
210			
220			

LABELS

A GEN SEED	B n START	C MOVE	D SCORE	E RESET	0 ON-NO START
a	b	c	d	e "error"	1 ON-NOT 1ST MOVE
0 START GAME	1 CALC FIRST	2 CALC MOVE	3 MOVE LOOP	4 MAKE MOVE	2
5 LOSE	6	7	8	9	3 DATA ENTRY

SET STATUS

FLAGS			TRIG	DISP	
	ON	OFF			
0	☐	☒	DEG ☒	FIX ☒	
1	☐	☒	GRAD ☐	SCI ☐	
2	☐	☒	RAD ☐	ENG ☐	
3	☐	☒		n 0	