03398D Program Description I

Program Title Simple 2D Star Trek

Contributor's Name John Nelson
Address 1226 E. University Ave.
City Des Moines **State** Iowa **Zip Code** 50616

Program Description, Equations, Variables

This program is a simple one-card version of Star Trek. It was designed for people who do not want to read a card for performing another function of the program. The game is a two dimensional game so that the board is easier to diagram and the position is easier to visualize. The basic pattern of play is as follows:

The game is started by inputing a seed and pressing the Initialize function key. The seed must be a decimal number between zero and one. The initialization will locate three klingons and the Enterprise on a 100 x 100 grid (location coordinates are 00 thru 99 for both x & y). It also initializes some energy into the main energy banks of the Enterprise for use in Phasers, Shields and warp drive.

The Captain of the Enterprise then decides to run the sensors, move the ship, divert energy to shields to prepare for battle, or attack enemy vessels. The sensors are necessary to locate enemy ships and determine damage of Enterprise and enemy. As soon as an enemy is found, (on the sensors) it opens fire. This is the only time that the Enterprise will automatically put up its shields. Power is not actually diverted from main banks at this time, but the shot is deflected.

Operating Limits and Warnings

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.

Program Description I

Program Title Simple 2D Star Trek

Contributor's Name John C. Nelson
Address 1226 E, University Ave.
City Des Moines **State** Iowa **Zip Code** 50316

Program Description, Equations, Variables The damage caused by the enemy fire is added to a central damage accumulator and when the scan is completed, all damage is displayed. If more than one enemy vessel is detected, all that are detected will fire. This makes the enemy the aggressor and the Enterprise must fight to save itself. As soon as the sensor scan has completed, the Enterprise coordinates are returned to the display. You may then prepare for battle. That is, if you have detected an enemy within range. The range of the phasers is $\frac{1}{2}$ the range of the sensors. To prepare for battle, input the amount of energy to be diverted to the shields. This energy will be taken out of the main energy banks and may never be replaced. The main banks are used for moving the Enterprise, which is the highest use of energy, so be careful not to run yourself too low in the main banks. The maximum you sould ever need at one time in shield banks is 100 units. This will be depleted from one shot to another by enemy fire, but it can be replentished between shots. The shield power may be viewed at any time by pressing RCL 6. It is also displayed after inputting the ship # to be battled and pressing 'E'. Do not attempt to replentish them at this point, however because the program must be restarted at the point of the R/S for phaser power.

Operating Limits and Warnings

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.

03398D Program Description I

Program Title Simple 2D Star Trek

Contributor's Name John C. Nelson
Address 1226 E. University Ave.
City Des Moines **State** Iowa **Zip Code** 50316

Program Description, Equations, Variables After the ship has been properly shielded, you may attack the enemy (or enemies) one at a time using the 'E' function. This is the first half of the attack routine. The ship # that you want to attack is input and the function key pressed. This will compute the power of the enemy attack, its effect on the shields and damage to the Enterprise (although this is not displayed yet.)

The program then stops with the remaining shield power in the display. DO NOT press any function key at this time because the program must be restarted at this point with an R/S. You may recall and display any information you want to, but the program pointer must be unaffected. The program has stopped for phaser power to be input. The power for the phaser is a digit from 1 to 9 inclusive with the maximum = 9 and minimum = 1. The more power you use, the more energy it will burn up. The energy used is = power x distance^2. Phasers lock on target automatically. Damage to enemy is output with a print-(flash display) then the damage to the Enterprise. (Unless a ship is destroyed.) If a ship is destroyed, it's I.D. (ship #) is displayed as a negative number. If the Enterprise is destroyed, for example, the program will stop with a -4.00 in the display. The final output

Operating Limits and Warnings

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.

Program Description I

Program Title Simple 2D Star Trek

Contributor's Name John Nelson

Address 1226 E. University Ave.

City Des Moines **State** Iowa **Zip Code** 50316

Program Description, Equations, Variables is the Enterprise coordinates (assuming that no ships have been destroyed.).

Formulas Used:

Phaser Energy Used = Power setting x distance^2 or $E_{Ph} = p \times d^2$

Speed = Warp# x 10

Maximum Warp# = 8D/100 where D = damage to Enterprise

Power of enemy attack = $(100 - D_e)r$ where D_e is damage to enemy
r is decimal random #.

Damage to Enterprise = $\frac{1}{4}$(Power of enemy attack) if shields hold.
or 4(Power of enemy attack - shields) if shields do not hold.

Shields reduction = Power of enemy attack if Shields are greater than Power of attack, otherwise the shields are reduced by (power of attack - shields) and The 'power of attack' is also deducted from R_8.

Operating Limits and Warnings

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.

Program Description I

Program Title Simple 2D Star Trek

Contributor's Name John C. Nelson

Address 1226 E. University

City Des Moines **State** Iowa **Zip Code** 50316

Program Description, Equations, Variables

Each ship's information is contained in a single register in the format: XX.YYDDDVDDD

where XX is the ship's x coordinates.

YY is the ship's y coordinates.

DDDVDDD is the damage to the ship. (The V is an assumed decimal place.)

Whenever the damage factor reaches 100 units, the ship is considered to be destroyed. Ships are located by their ship #. Ship #s 1, 2, and 3 are Klingons, ship # 4 is the Enterprise. As soon as damage is added to a ships register the damage is checked for 100 to see if the ship has been destroyed.

Operating Limits and Warnings

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.

Program Description I

Program Title Simple 2D Star Trek

Contributor's Name John C. Nelson
Address 1226 E. University Ave.
City Des Moines **State** Iowa **Zip Code** 50316

Program Description, Equations, Variables This portion of the program description will contain more detailed information about the programs internal functions. In case you want to modify the program this may prove a valuable feature.

All labels are used and they are arranged in such a manner that it is difficult if not impossible to re-use any of them. This was not done on purpose, it just came out that way. If you want another label you could remove LBL A from the program and initialize with RTN R/S. I have shortened the program time after time and I now think it is about as short as it can be, so you won't find too many steps that can be cut from it.

The labelled functions are detailed below:

LBL A - Initializes the program. Everything that the program needs is initialized EXCEPT R_5. Register 5 is used for storing Enterprise damages temporarily until the Enterprise register is updated with the new damages. The R_5 register is zeroed when the damage is put into the Enterprise register. This is not done when an enemy ship has been destroyed, because the program stops before

Operating Limits and Warnings

Damage to the Enterprise is not updated until after the enemy damage is updated. Therefore, if a ship is destroyed and you are not going to repair the Enterprise before going to battle the next ship, the damage has not been applied to the Enterprise yet. This means that register 5 contains the damage that still has not been applied. If you want to nullify this, simply press Ø STO 5. If you want to apply it press f a.

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.

03398D

Program Description I

Program Title Simple 2D Star Trek

Contributor's Name John C. Nelson
Address 1226 E. University Ave.
City Des Moines **State** Iowa **Zip Code** 50316

Program Description, Equations, Variables

the Enterprise damage is updated. This means that if you do not take action as given on prvious page under Operating Limits and Warnings, then the next time you do battle with an enemy this damage will be dropped if you do battle before using sensors. If you use the sensors before doing battle again, then the remaining damage in register 5 will be applied to the Enterprise along with any new damage from finding an enemy ship.

LBL B - Sensors. This routine loops from register 3 to register 1 looking for enemy vessels whose distance from the Enterprise is less than the range of the sensors. The range of the sensors is set up in program steps 7 thru 12 along with two other fields at the same time. This is done by calculation. Since a constant of '100' must be stored in RA, and a 4 must be stored in register I to begin locating ships, the range was calculated from the two using ÷ (division). If you want to change the range of the sensors, the easiest way is to store the new range in reg. 0 after the initialize routine has been run. To compute the damage suffered by the Enterprise when an enemy vessel is

Operating Limits and Warnings

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.

Program Description I

Program Title Simple 2D Star Trek

Contributor's Name John C. Nelson
Address 1226 E. University Ave.
City Des Moines **State** Iowa **Zip Code** 50316

Program Description, Equations, Variables

detected, the random number generator is run and when it returns to the calling routine Lastx is recalled and used as the damage factor. This makes the damage factor always less than 45 units. When the damage is applied from register 5, it is divided by 4 before being applied. This is always true with the damage factor in register 5 except when the shields do not hold in the battle routine.

LBL C - Move Routine. There is nothing really tricky in the move routine, except the way that the coordinates are updated. They are updated by computing the x & y coordinate changes then adding to the current x & y coordinates. This simply means that if you go off the board, you may get unpredictible results.

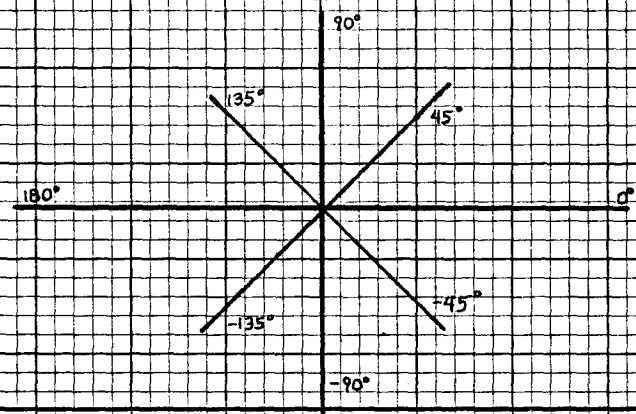
LBL D - Diverts energy from main energy banks to shields. If there is not enough energy in the main banks to satisfy the request, you will get a continuous pausing '0.00'. This same routine is performed within the program to take energy from the main banks for phasers and warp drive. Whenever there is not enough, you will get a

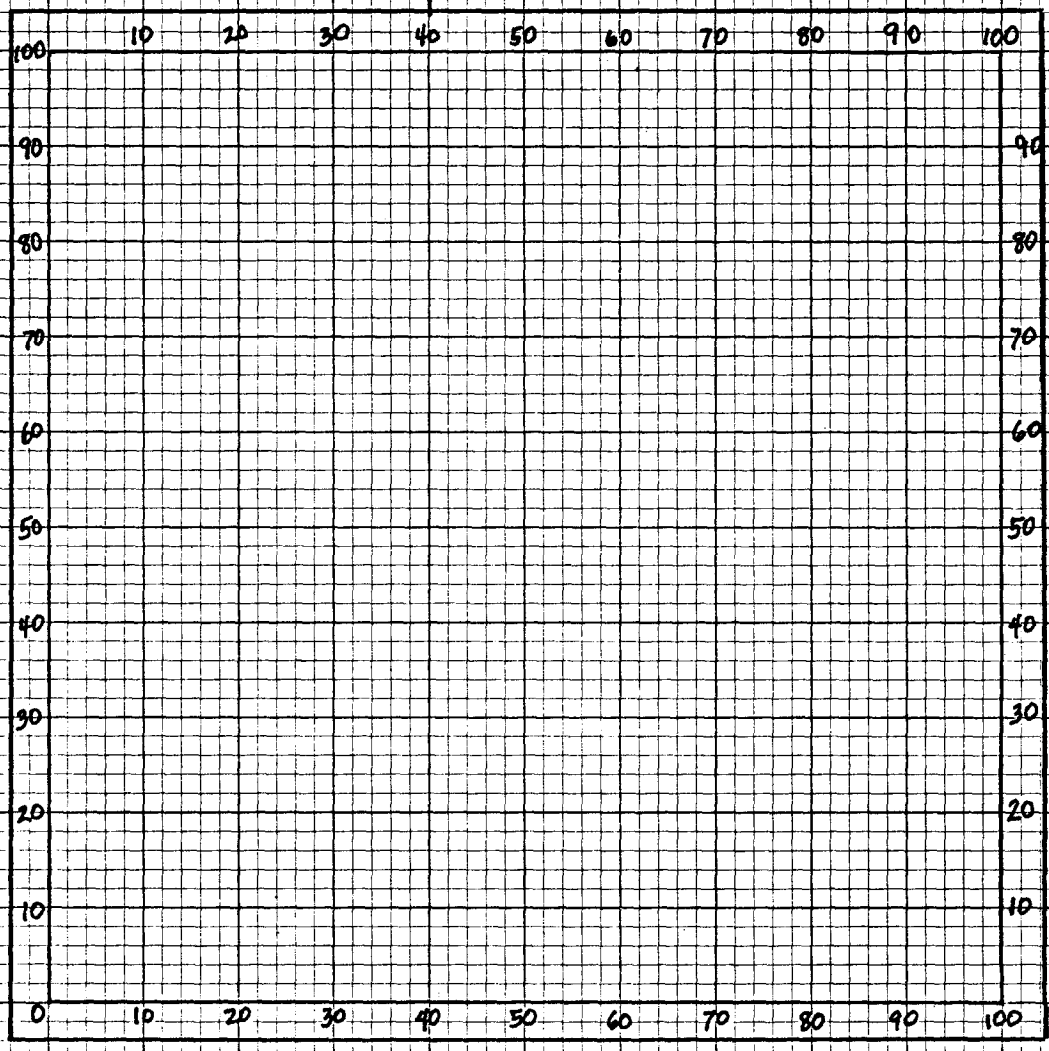
Operating Limits and Warnings

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.

Playing Board





Each square = 2 Units

Program Description I

Program Title Simple 2D Star Trek

Contributor's Name John C. Nelson
Address 1226 E. University Ave.
City Des Moines **State** Iowa **Zip Code** 50316

Program Description, Equations, Variables

blinking zero. If you want to continue the game, stop the program during one of the pauses using the R/S, then store some energy in R_8. Continue game where you left off. This might not work in some cases, depending on where you got the continuous '0.00' display. When I designed the game, however, this was supposed to be a signal that you lost the war, so take your own chances.

LBL E - This is the battle routine. The distance from the Enterprise to the enemy being attacked is computed and the power of the alien attack is computed. The more the alien is damaged, the less severe (generally) is his attack, although there is also a random factor involved. The power of the alien attack is taken directly from the shield power and if there is not enough, woe on you!

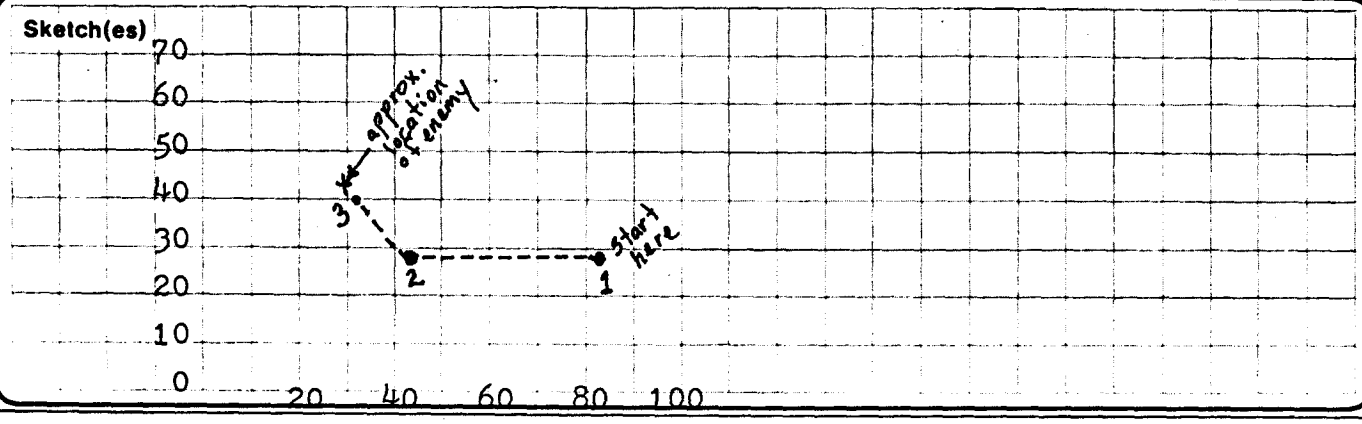
Operating Limits and Warnings

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.

Program Description 11

Sketch(es)

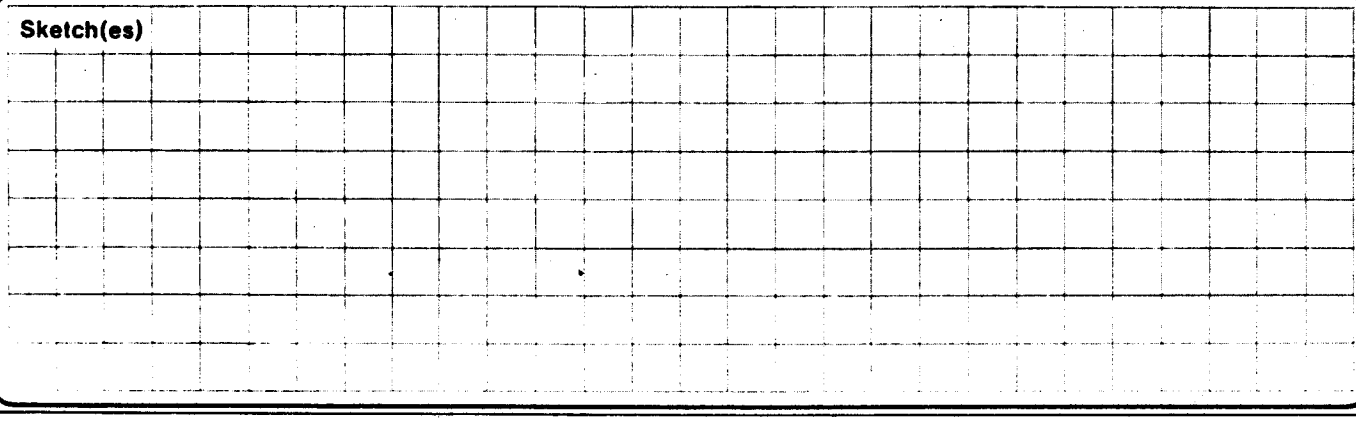


Sample Problem(s) Use .5281953751 as a seed: .5281953751 A ----→ 84.28*** Enterprise coordinates. (84,28). The coordinates are given in standard ordered pair notation of x first then y. Now we are ready to scan for enemy vessels. B ----→ 84.28***. Enterprise coordinates returned. No ships found. Enterprise is now at position 1 in above sketch. Let's move to the left (180°) about 40 units. This will mean an angle of 180 and a speed of warp 4. Input move data as follows: 180 ↑ 4 C ----→ 44.28***. New coordinates. Scan again: B ----→ 1.*** 22.***... 137*** 4.*** 44.28***. We are now at position 2 and have detected a ship at a bearing of 137°. The scan tells us the ship# (1st output) which is # 1, the distance (2nd output) which is 22 units away, the bearing (3rd output) which is 137° and the damage to the Enterprise (4th output) from enemy fire. Then the coordinates are returned to the display. To attack the enemy, we must be ½ the range of the sen-

Solution(s) sors from it. This means we must be within 12½ units of the enemy, since the range of the sensors is 25 by default. (see user instructions for default values.) Let's move closer. Sensors indicate the enemy is at bearing of 137° from us. Therefore let's move at 137° for a speed of 1.8. Input move data: 137 ↑ 1.8 C ----→ 31.40*** we are now at position 3 in diagram. We should be close. If we scan again, we will be fired on again, so let's assume we have done this correctly and that we are close enough for battle. First we should put up our shields. Let's set them to 110 units. 110 D ----→ 110.00

Reference(s)

Sketch(es)

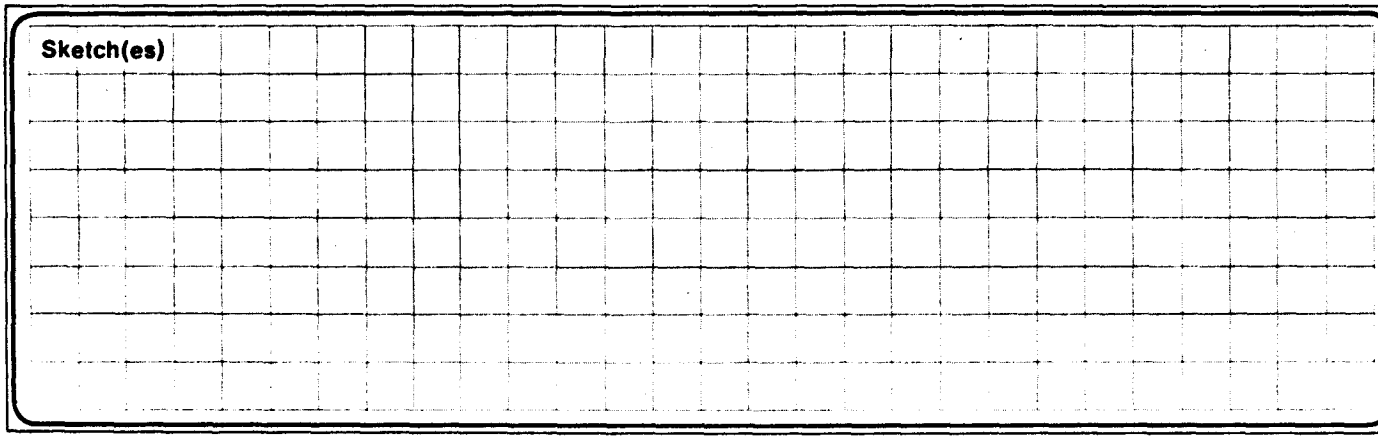


Sample Problem(s) Now we are ready to attack. Input the ship number to be attacked.(#1) and attack: 1 E ----→ 102.58 Shields are reduced to 102.58 from 110.00. This is not very much damage when compared to what damage is possible. Enemy power can conceivably reduce the shield by as much as 100 units. Now fire Phasers: 9 (maximum power) R/S----→ 11.92*** 1.86*** 31.40***. Damage to enemy was 11.92 units. The damage to the Enterprise was 1.86 units and the Enterprise coordinates are returned to the display. Since the shields are still above 100 units (they are 102.58) we will not have to replentish them yet, so we can attack again: 1 E ----→ 32.27 Shields left. This time the damage was more extensive and the power of the enemy attack reduced the shields to 32.27 units. We will have to re-vitalize them before our next attack, but right now let's fire: 9 R/S ----→ 29.10*** ... 17.58*** 31.40***. Damage to enemy vessel was 29.10 units. Damage to

Solution(s) Enterprise was 17.58. Then the coordinates are returned to the display. Let's check the total damage to the enemy and to us. First check damage to the Enterprise: 4 f b ----→ 23.40. We are almost 25% destroyed. Let's try some repair before we scan for damage on the enemy ship. Let's say we want to repair 6 units of damage. To repair any damage, you must have the ship # to be repaired stored in R_I. Then input the amount of damage to be repaired as a negative number and press f c. So now press: 6 CHS f c ----→ 31.40. Note that we did not have to store the ship # (a 4) in R_I because the prior

Reference(s)

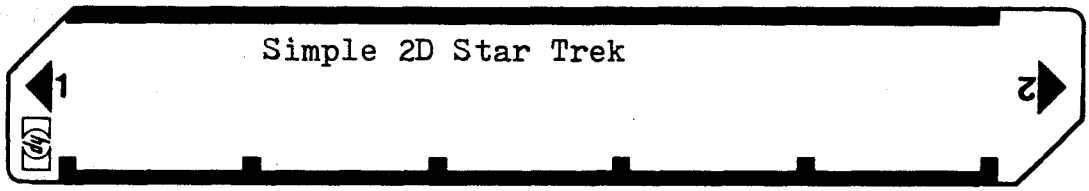
Sketch(es)



Sample Problem(s) step was a recall of the damage where we did input the 4. This routine stores the ship # so that R_I already did contain a 4. Now check damage again: 4 f b ----→ 17.40. Now check enemy damage: 1 f b ----→ 41.02 Let's attack again. First we must replentish the shields as they were down to 32.27 units last shot. 100 D ----→132.27 Now attack: 1 E ----→ 77.17. 9 R/S ----→ 2.37*** 13.77*** 31.40*** Damage to enemy = 2.37 units. Damage to Enterprise = 13.77. Check his damage: 1 f b ----→ 43.39. Re-vitalize shields: 75 D ----→ 152.17 Attack: 1 E ----→ 107.08 9 R/S ----→ 26.09*** 11.27*** 31.40*** Damage to enemy = 26.09. Damage to Enterprise was 11.27. 1 E ----→ 99.41 9 R/S ----→ 5.07*** 1.92*** 31.40***. 1 E ----→ 74.84 9 R/S ----→ 48.49*** -1.00 Enemy was finally destroyed! Now you are ready for the other two. ...Or are you? To repair all damage: 4 f b ----→44.37 CHS f c ----→ 31.40

Solution(s) Check damage to enterprise: 4 f b ----→ 0.00 Now try the other two on your own, as an exercise.

Reference(s)



STEP	INSTRUCTIONS	INPUT DATA/UNITS	KEYS	OUTPUT DATA/UNITS
1.	Load program sides 1 & 2.			
2.	To start a new game.			
	a.) Input a seed between 0 & 1	seed	A	Ent.Co**
	b.) Change default values if desired.			
	defaults are:			
	1) Total energy = 20,000 units.			
	2) Range sensors = 25 units.			
	To change total energy	Energy	STO 8	
	To change range of sensors	Range	STO 0	
3.	To run a sensor scan for locating			
	enemy ships.		B	(if in
	Outputs are ship #, distance & bear-			range)
	ing for any ships in range. When			Ship#**
	all ships scanned, damage to the			dist. **
	Enterprise is displayed, then the			brng. **
	coordinates of the enterprise.			:
				dmge **
				Ent.Co**
4.	To move the Enterprise			
	a) Input the angle you want to move	Angle°	↑	
	90°			
	180° —— 0°			
	-90°			
	This may be any angle 0° - 360°			
	(or negative)			
	b) Input the warp number 1 - 8.	warp #	C	NewCo.**
	You cannot move greater than 8			
	and if the Enterprise is damaged			
	you can move proportionately			
	slower. If you are 50% destroyed			
	(50 units damage) you can only			
	move warp 4.			

User Instructions

STEP	INSTRUCTIONS	INPUT DATA/UNITS	KEYS	OUTPUT DATA/UNITS
4.	Move Routine - continued.			
	The speed that the ship travels will			
	be 10 times the warp number.			
	You may move non-integer warp numbers.			
5.	Before going into battle, you should	Energy	D	Shields
	divert energy from main energy to			
	shield energy or you will be destroyed			
	rather quickly. You must also re-			
	plentish this energy throughout the			
	battle, as enemy fire will soon			
	destroy the shields. The maximum you			
	will ever need at one time for			
	shields is 100 units.			
6.	To attack, you must first be within			
	attacking distance. Attacking dis-			
	tance is ½ range of sensors.			
	Battle can be directed at only one			
	klingon at a time.			
	To attack input klingon # to be			
	battled.	Kling.#	E	rem.shld
	Output is remaining shields.			
	Now input phaser power to be used.			
	This should be a number 1 thru 9.	Power	R/S	Kl.Dam**
				En.Dam**
	Outputs are Enemy Damage, Enterprise			Ent.Co**
	Damage, then Enterprise Coordinates.			(Destryd
	If display stops with a negative #			Ship #)
	in the display, the ship # displayed			
	Has been destroyed.			
	If you are not close enough for			
	an attack, the Enterprise coordinates			
	will be output with no other output.			

User Instructions

STEP	INSTRUCTIONS	INPUT DATA/UNITS	KEYS		OUTPUT DATA/UNITS
	NOTE: IF AT ANY TIME YOU ARE GETTING				
	A CONTINUOUSLY PAUSING "0.00" IN THE				
	DISPLAY, IT MEANS YOU HAVE USED ALL				
	THE ENERGY IN THE MAIN BANKS AND DO				
	NOT HAVE ANY LEFT FOR MOVING OR FIRING				
	PHASERS. I USUALLY CONSIDER THIS A				
	LOST GAME, BUT IF YOU WANT TO CONTIN-				
	UE, STOP THE PAUSE WITH R/S, THEN				
	INPUT THE AMOUNT OF ENERGY YOU WANT				
	TO CONTINUE WITH FOLLOWED BY : STO 8				
7.	To recall the damage for any ship:	Ship #	f	b	damage
	Ship #'s:				
	1-3 = Klingons 1 thru 3.				
	4 = Enterprise				
	(This also stores ship# in R_I)				
8.	To repair damage to a ship				
	a.) Store ship # if not already	Ship #	STO	I	
	b.) Input amount of damage to be	Amt-rep.	CHS		
	repaired. AS A NEGATIVE VALUE.		f	c	Ship Co.
	output is ship coordinates				
9.	To recall Enterprise coordinates		GSB	0	Ent.Co**
	(XX.YY)				
10.	To recall total energy remaining		RCL	8	Tot.eng.
11.	To recall shield energy		RCL	6	Shields
12.	To check range of sensors.		RCL	0	
	(Once set, they will never change				
	unless you change them.)				
13.	To refresh seed, simply store a new	seed	STO	E	

033980

Program Listing I

STEP	KEY ENTRY	KEY CODE	COMMENTS
001	*LBLA	21 11	Initialize Routine Store Seed
002	STOE	35 15	
003	2	02	
004	EEX	-23	
005	4	04	Store initial energy = 20,000 units
006	STO8	35 08	
007	EEX	-23	
008	2	02	Store Constant '100' in R_A
009	STOA	35 11	
010	4	04	Initial. for storing of ship locations.
011	STOI	35 46	
012	÷	-24	Create Sensor Range = 25
013	STO0	35 00	
014	0	00	Zero R_6 (Shield Energy)
015	STO6	35 06	
016	*LBL9	21 09	Generate Random # in format XX.XX and Round for Ship Location
017	GSB8	23 08	
018	RCLA	36 11	
019	x	-35	
020	RND	16 24	
021	STOi	35 45	
022	DSZI	16 25 46	Continue to next ship.
023	GTO9	22 09	
024	*LBL0	21 00	Done - Display Enterprise Coordinates
025	RCL4	36 04	
026	DSP2	-63 02	
027	PRTX	-14	
028	R/S	51	Stop - in case pending return.
029	*LBL8	21 08	Recall Seed
030	RCLE	36 15	Random # = FRAC[TAN^{-1} of seed]
031	TAN^{-1}	16 43	
032	FRC	16 44	
033	STOE	35 15	
034	RTN	24	
035	*LBLB	21 12	Sensor Routine
036	3	03	Set up starting ship address @ 3.
037	STOI	35 46	
038	*LBL6	21 06	
039	GSB2	23 02	Compute distance Range of Sensors
040	RCL0	36 00	Is ship in Range?
041	X≤Y?	16-35	No - Continue.
042	GTO5	22 05	
043	R↓	-31	Yes - Display Ship.
044	RCLI	36 46	
045	DSP0	-63 00	First - display I.D. in Print/Flash
046	PRTX	-14	
047	R↓	-31	Roll down distance
048	PRTX	-14	Print/Flash distance
049	X≷Y	-41	
050	PRTX	-14	Print/Flash Angle
051	SPC	16-11	
052	GSB8	23 08	Generate random #
053	LSTX	16-63	Get TAN^{-1} of Seed to be added to damage
054	ST+5	35-55 05	
055	*LBL5	21 05	Decrement to next ship.
056	DSZI	16 25 46	
057	GTO6	22 06	Continue Scanning
058	GSBa	23 16 11	Compute & store damage (if any)
059	GTO0	22 00	
060	*LBL7	21 07	Separate X & Y Coord. Subroutine
061	INT	16 34	
062	F2?	16 23 02	If First Time, store x in R_B
063	STOB	35 12	
064	LSTX	16-63	
065	FRC	16 44	
066	RCLA	36 11	Shift Y coord. to left decimal
067	x	-35	
068	INT	16 34	
069	RTN	24	Return to calling rout.
070	*LBLC	21 13	Move Routine Store warp #
071	STOC	35 13	
072	X≷Y	-41	Store Angle
073	STOB	35 12	Put 8 in Stack
074	8	08	Put 100 in Stack
075	RCLA	36 11	Get Damage
076	RCL4	36 04	
077	GSBd	23 16 14	Subtract damage from 100. & Calc % of 8. = Warp Max.
078	-	-45	
079	%	55	
080	RCLC	36 13	Is warp in > Max?
081	X>Y?	16-34	Yes - Exchange Max
082	X≷Y	-41	
083	1	01	
084	0	00	
085	x	-35	
086	STOC	35 13	Speed = 10 x Warp #
087	X²	53	Energy used = (W# x 10)³
088	GSB1	23 01	
089	RCLB	36 12	
090	RCLC	36 13	Compute X & Y change (ΔXY)
091	+R	44	
092	INT	16 34	
093	ST+4	35-55 04	Update X & Y coord.
094	X≷Y	-41	
095	INT	16 34	
096	RCLA	36 11	
097	÷	-24	Add ΔX.Y to R4
098	ST+4	35-55 04	Go to display coord.
099	GTO0	22 00	
100	*LBL1	21 01	Take energy from R_8 (Total) Routine
101	SF2	16 21 02	Divert Energy to shields routine.
102	*LBLD	21 14	
103	RCL8	36 08	Is there enough energy
104	X≷Y	-41	Yes - Go to 4.
105	X≤Y?	16-35	
106	GTO4	22 04	No - Continuously Pause "0.00"
107	0	00	
108	STOI	35 46	
109	DSZI	16 25 46	
110	PSE	16 51	
111	GTOi	22 45	
112	*LBL4	21 04	

REGISTERS

0 Sensor Range	1 Klingon 1 XX.YYDDD	2 Klingon 2 XX.YYDDD	3 Klingon 3 XX.YYDDD	4 Enterprise XX.YYDDD	5 Damage Accumulator	6 Shield Energy	7	8 Total Energy	9
S0	S1	S2	S3	S4	S5	S6	S7	S8	S9

A	B	C	D	E	I
Constant '100'	-	Used - for warp#		Seed	Ship Pointer

STEP	KEY ENTRY	KEY CODE	COMMENTS
113	ST-8	35-45 08	
114	F2?	16 23 02	
115	RTN	24	
116	ST+6	35-55 06	
117	RCL6	36 06	
118	RTN	24	
119	*LBLE	21 15	
120	STOI	35 46	
121	GSB2	23 02	
122	STOB	35 12	
123	RCL6	36 06	
124	RCLA	36 11	
125	RCLi	36 45	
126	GSBd	23 16 14	
127	-	-45	
128	GSB8	23 06	
129	x	-35	
130	X>Y?	16-34	
131	GSB3	23 03	
132	ST-6	35-45 06	
133	STO5	35 05	
134	RCL6	36 06	
135	R/S	51	
136	STOD	35 14	
137	RCL0	36 00	
138	2	02	
139	÷	-24	
140	RCLB	36 12	
141	X>Y?	16-34	
142	GTO0	22 00	
143	X=	53	
144	RCLD	36 14	
145	x	-35	
146	GSB1	23 01	
147	GSB8	23 08	
148	RCLA	36 11	
149	x	-35	
150	RCLB	36 12	
151	√x	54	
152	÷	-24	
153	PRTX	-14	
154	GSBc	23 16 13	
155	GSBa	23 16 11	
156	GTO0	22 00	
157	*LBL3	21 03	
158	GSB1	23 01	
159	R↑	16-31	
160	-	-45	
161	SF2	16 21 02	
162	RTN	24	
163	*LBL2	21 02	
164	RCLi	36 45	
165	SF2	16 21 02	
166	GSB7	23 07	
167	RCL4	36 04	
168	GSB7	23 07	

Is this only a reduction of R_8? - Yes-RTN

No - it is a divert to shields. - Add R_6 Recall & Return.

Attack Routine.
Store ship # to be attac
Compute distance
Store distance.
Put shields in STK.
Put '100' in STK.
Get damage.

100 - damage x r > Shields ?

Is hit > shields?
Yes - shields not holding

Store damage in R_5
Display remaining shields in Stop.
Store Power of Phase

Compute ½ Sensor Range

Distance.
Is d > ½ S.R.?
Yes - display coord.

Power used = # x d^2
(# is power level)

Damage to Enemy = $\frac{r \times 100}{\sqrt{d}}$

Apply damge to enemy
Apply damge to Ent.
Display Coord.

Shield Power not enough.
Subtract power of hit from R_8
then subtract Shields from Hit.

Compute distance
Recall ship (enemy)
Turn on 1st Time Switch
Separate X & Y (store X)
Recall Enterprise.
Separate X & Y.

STEP	KEY ENTRY	KEY CODE	COMMENTS
169	RCLB	36 12	Recall Enemy x
170	R↓	-31	
171	R↓	-31	Rearrange Stack and subtract
172	X≷Y	-41	
173	R↑	16-31	x and y
174	-	-45	coordinates of
175	R↓	-31	Enterprise and
176	-	-45	Enemy
177	R↑	16-31	Then compute
178	X≷Y	-41	polar coord.
179	→P	34	
180	RTN	24	
181	*LBLe	21 16 15	A Ship has been
182	RCLI	36 46	destroyed.
183	CHS	-22	Identify to
184	STOi	35 45	User.
185	R/S	51	
186	*LBLb	21 16 12	
187	STOI	35 46	Recall damage
188	RCLi	36 45	of R_i ship
189	*LBLd	21 16 14	after storing x
190	RCLA	36 11	in R_i
191	x	-35	
192	FRC	16 44	Recall damage from
193	EEX	-23	data in display
194	3	03	format $_x$ = XX.YYDDD
195	x	-35	
196	RTN	24	
197	*LBLa	21 16 11	Apply damage
198	RCL5	36 05	to Enterprise
199	X=0?	16-43	from R_5.
200	RTN	24	Zero R_5
201	ST-5	35-45 05	
202	4	04	Store 4 for Enter.
203	STOI	35 46	for use in c.
204	F2?	16 23 02	If shields did not
205	1/X	52	hold, multiply
206	÷	-24	damage by 4
207	PRTX	-14	else divide by 4.
208	*LBLc	21 16 13	
209	RCLi	36 45	Apply damage to
210	GSBd	23 16 14	ship and
211	+	-55	see if it has
212	RCLA	36 11	been destroyed.
213	X≤Y?	16-35	
214	GTOe	22 16 15	
215	R↓	-31	
216	EEX	-23	
217	5	05	Shift damage to
218	÷	-24	.00 DDD format.
219	RCLi	36 45	
220	DSP2	-63 02	Round off old
221	RND	16 24	damage & add
222	+	-55	new.
223	STOi	35 45	
224	RTN	24	

LABELS

A Initialize	B Sensor Scan	C Move Rout.	D Divert energy to shields.	E Attack	0
a apply dam. to Enter.	b display damage	c apply damage to R(i).	d Get damage	e Display destroyed ship.	1
0 Used	1 Used	2 Used	3 Used	4 Used	2 Used: Multiple Purp.
5 Used.	6 Used	7 Used	8 Used	9 Used	3

FLAGS

SET STATUS

FLAGS	ON	OFF
0	☐	☒
1	☐	☒
2	☐	☒
3	☐	☒

TRIG	
DEG	☒
GRAD	☐
RAD	☐

DISP	
FIX	☒
SCI	☐
ENG	☐
n	2