Z-100 LifeLine
Since 1989

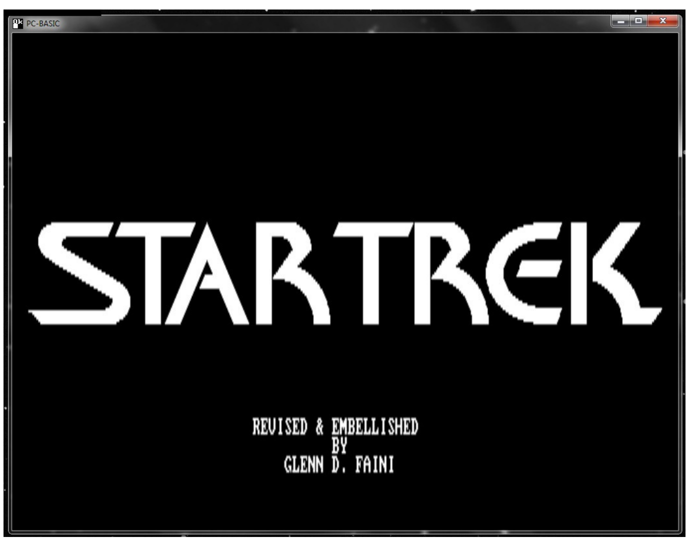
April 2026

#WEB

ZBASIC STARTREK.BAS

by Steven W. Vagts
Editor, "Z-100 LifeLine"

ZBASIC STARTREK.BAS



The game STARTREK.BAS program has been around since the early days of the Z-100. The most recent developer, Glenn D. Faini, has been refining this ZBASIC version since the mid 1980s and has just recently released version 11. He had asked me to test this most recent version and I was so impressed that I decided to share the program here on the website with you.

As with the other Z-BASIC programs, I have provided the full listing. This version runs under Z-BASIC, GW-BASIC, and PC-BASIC. It can easily be converted to run on other BASIC interpreters. On computers with sound capability, such as a PC-clone, it has full color, extended graphics, sound effects, and music.

You can copy this listing into your own editor, make revisions to accommodate the version of BASIC you wish to use, change the listing to the .BAS extension, then run the program with your BASIC Interpreter.

Description:
From the developer, Glenn D. Faini:

STAR TREK is a strategy game based on the 1960s television series.

High school senior Mike Mayfield wrote the original BASIC game in October 1972 to work on an HP2000C minicomputer via printer terminals. Mike Mayfield released his BASIC program to the Public Domain in February 1973 via the HP Contributed Program Library. The game was developed further by David H. Ahl and Mary Cole of Digital Equipment Corp. (DEC) and Bob Leedom.

Countless versions have been released since, including one by William G. Bryson of Texas Instruments in January 1979 for the TI-59 programmable calculator.

The Z-BASIC version that I obtained in 1983 while a junior at Clarkson Memorial College of Technology was a text-based game with no graphics, sound, or color. Between 1983 and 1993 I squashed bugs, and enhanced the game with graphics, color ANSI text, music, and sound effects. In 2025 I updated the program to run on Windows 7 and newer using PC-BASIC v2.0.7.

* Runs on Zenith Z-100 computers running Z-BASIC v1.1, IBM-PC/XT/AT compatibles running BASICA or GW-BASIC, and Windows 7 and newer running PC-BASIC.
* Optional multi-color ANSI text displays.
* Detects the lack of color chips in Z-100s and runs in monochrome mode.
* A Help file, accessible at any time.
* Saves the most recent High Score and a list of all completed games in ASCII text files.

* Color graphics inspired by STAR TREK **The Motion Picture** and THE WRATH OF KAHN.
* Function keys for all common commands.
* Extended color graphics, sound effects, and music.
* Variable time loops to run properly at 5 MHz, 8 MHz, with a math co-processor, and on modern high speed computers.

The game, as distributed, consists of the following files:

STARTREK.PDF - This description file.

STARTREK.BAS - The BASIC game, stored in plain text. You can resave it in tokenized format to speed up loading.

The variable Z100$ in line 140 must be set to "TRUE" to run on the Z-100 and "FALSE" to run other computers.

The variable SCRN% in line 150 is used to set the graphics screen on IBM compatibles. It should be set to "2" for CGA monochrome graphics, or "8" to take advantage of EGA color graphics.

The variable PAUSEFACTOR% in line 160 is used to adjust the pause length on faster computers. It should be "1" for 5 MHz computers, "2" for 8 MHz computers, and "5" for 80286 based computers. You can set it to other values, such as 15 or 20, if you like. The higher the number, the longer the pause will be.

The variable ANSI$ in line 170 enables the ANSI character set. An ANSI driver (ZANSI.DVD) is required on the Z-100 (standard on most other computers). See the section, "**Running STARTREK on the Z-100**" below, for additional information.

If you put your name in line 130, you will only have to press RETURN when asked.

STARTREK.HLP - Help file accessible from the COMMAND prompt.

STARTREK.LOG - BASIC game change log.

STARTREK.VAR - Descriptions of Numeric Variables, Text Strings, and Arrays.

The following files will be created when you run the game.

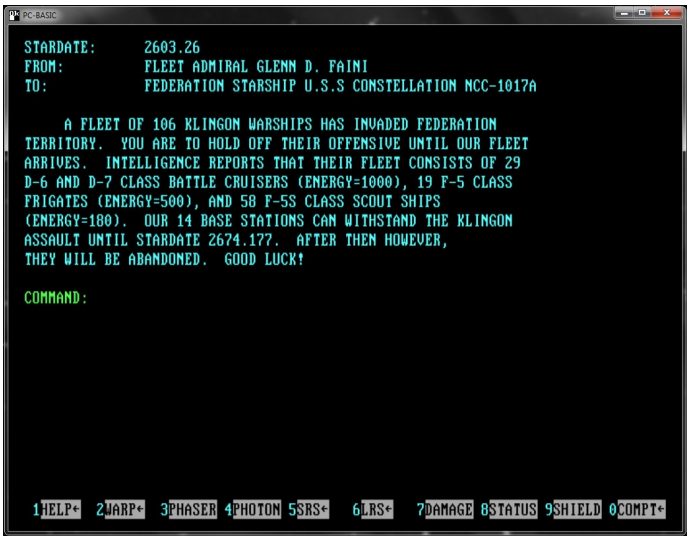
STARTREK.DAT - Data file listing all completed games.

STARTREK.REC - Most recent High Score.

You may make copies and use this game freely. If you have any suggestions for improvements or discover any bugs, please notify me so I can modify the original program.

Glenn D. Faini
gdfaini@gmail.com

Other screens shown include:
At the start of the game, you receive your orders from Starfleet:



It provides an overall picture of the situation, listing the number and composition of Klingon vessels, number of Federation Bases in your patrol area, and the Stardate when all the bases will be abandoned, after which, you are on your own.

Note the function keys at the bottom of the screen. These are how most of the action takes place.

Your patrol corridor is divided into 121 quadrants, each 1 cubic lightyear, in an 11 x 11 array, that represents the playing field (Space). Each quadrant has an array of 100 sectors (10 x 10 grid). These are best represented on the Short Range Sensor display, below.

These Function Keys are defined as:

<F1> - "HELP":
Displays an internal HELP file.

<F2> - "WARP": WARP DRIVE CONTROL
You will receive the prompt "COURSE, WARP-FACTOR: "

Your response must be a set of three numbers separated by commas, as:

Course, Warp: -1,-1,2 {CR}

The course is input in two cartesian coordinates. Each whole number will move you a whole quadrant. Each tenth will move you a sector within a quadrant.

The WARP-FACTOR will determine how fast you will reach your destination. The higher the Warp-factor, the faster you will travel and the less time it will take to reach your destination, but it will consume more energy. The maximum recommended emergency speed is **WARP 8.**

In the example above, the course is the first two digits, moving -1 quadrant left, and -1 quadrant down, at warp 2.

In another example, Course, Warp: -1.5,1.3,1 {CR} would move us 1 quadrant left plus 5 sectors further left in the quadrant, 1 quadrant up plus 3 sectors more in the quadrant, at warp 1.

Examples: 1,1,1 .1,.5,1 0,5,9

Abort by entering 0,0,0 {CR}

<F3> - "PHASER": PHASER CONTROL
You can use the ship's phasers to damage or destroy your enemy. It requests only the power or energy you wish to expend, but the maximum amount of energy that should be channeled into the phasers per volley is **500 units.**

Note: The further the Klingon is from you, the more power that will be required. The energy of the Klingon warship varies with the ship's class; SCOUT = 180, FRIGATE = 500, BATTLE CRUISER = 1000.

The phasers can be overloaded in case of an emergency, to give more firepower, but this will damage them and they must be repaired before they can be fired again.

Abort by entering 0 {CR}

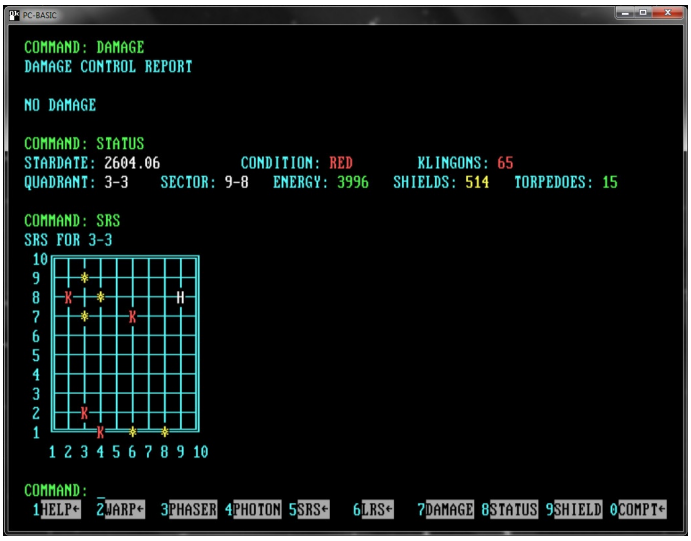
<F4> - "PHOTON": PHOTON TORPEDO CONTROL
The Photon Torpedo Control permits you to shoot up to 6 torpedoes, if available. Target information is listed as cartesian coordinates of the sector grid.

For example, the SRS (Short Range Sensor) for Quadrant 3-3, (shown below) has Klingons at coordinates (column - row) 3-2, 4-1, 6-7, and 2-8.

You will receive the prompt "TARGETS (4-8,...)". Target info becomes: 3-2,4-1,6-7,2-8,... Up to 6 targets can be used, each followed by commas, including after the last target. Each set of coordinates must be followed by a comma for torpedo control to accept the data.

Abort by pressing {RETURN}

<F5> - "SRS": SHORT RANGE SENSOR SCAN
The Short Range Scanner scans the quadrant that you are occupying, and shows the location of star clusters (*), Klingon warships (K), Federation Base Stations (B), and the first initial of the Starship's Name, e.g., (H) for U.S.S. Hood. When the starship is docked at a base station, the letter (B) is changed to (D) for docked. In our SRS picture, each tenth represents the move of one sector.



<F6> - "LRS": LONG RANGE SENSOR SCAN
The Long Range Sensor scans the surrounding quadrants but give less information than the Short Range Scan. The Long Range Sensor shows just three sets of three numbers in a tic-tac-toe grid. For example,

```
107   009   010
302   001   004
004   205   005
```

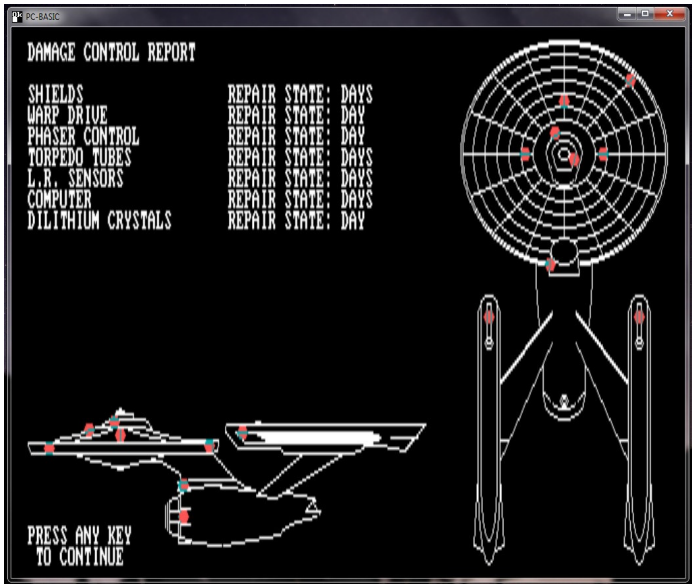
In each set of numbers, e.g. 107:

The first number, the hundred's position, shows the number of Klingon ships in that quadrant.

The second number, the ten's position, shows the number of Federation Bases in that quadrant.

The third number, the one's position, shows the number of star clusters in that quadrant. Your federation starship is located in the center quadrant (or box) of the 3x3 grid.

<F7> - "DAMAGE": DAMAGE CONTROL



Damage Control lists the statement "NO DAMAGE" or provides a graphic showing the ship's outline (top and side view) with flashing dots showing which areas of the ship are damaged.

The damaged systems are listed in the upper left corner of the graphic. The equipment that may be damaged and the time to repair are listed as DAYS:

SHIELDS
WARP DRIVE
PHASER CONTROL
TORPEDO TUBES
LONG RANGE SENSOR
SHIP'S COMPUTER
DILITHIUM CRYSTALS

<F8> - "STATUS": STATUS REPORT
Status Report lists the stardate, ship's condition state, Klingons remaining, quadrant, sector, with energy, shields and torpedoes remaining.

<F9> - "SHIELD": DEFLECTOR SHIELD CONTROL
Deflector Shield Control lists the amount of shield energy available. You can control the strength of the ship's shields, but it is limited to the sum of the present amount of energy in the shields and the amount of ship's energy available. To turn shields OFF, specify the strength of zero. The maximum that can be set is 4000! If you accidentally exceed that number, you lose up to 1000 of what was available!

Abort by pressing {RETURN}

<F10> - "COMPT":
Turns the Ship's Library Computer ON and changes the Function Keys available.

Note: The Menu of commands on line 25 changes when the Library Computer is turned on. The following is a list of computer commands:

<F1> - "OFF":
Turns the Library Computer OFF.

<F2> - "MAP":
Displays a MAP of your patrol corridor.
This map only contains information obtained through your long and short range scans.

<F3> - "KLI":
Scans Klingon vessels' position, distance, and energy. **Note**: Klingon positions are NOT static. This provides the most accurate data for targeting Photon Torpedoes.

<F4> - "BASE":
Reports location of all BASE STATIONS located on map. Additional information available is:

* Distance to Base Station
* Amount of energy available at the base station.
* Number of photon torpedoes available at the station.
* Course to the base station.

<F5> - "TIME":
TIME remaining before the Base Stations must evacuate. When the base stations evacuate, they all disappear from the map and they are no longer available to service your ship.

<F6> - "PHA":
Reports the position of Klingon vessels and the amount of energy needed to destroy the vessels with phasers.

<F7> - "COU":
The Navigation Computer gives the course and distance to a desired location. Enter the coordinates of the quadrant and section as:

7-6,10-2/8-4,1-5/...

More than one desired position can be entered.

<F8> - "SAVE":
SAVES your present location in the Helm Computer. Only one location can be saved at a time.

<F9> - "RET":
Instructs the computer to take control of the helm and return the ship to the saved location immediately. Condition is assumed to be critical.

<F10> - "RAT":
Gives your current performance rating.

You can END the game at any time by typing the Command **'END'**. This will end the game without saving your present status. You can NOT continue after issuing this command.

It should be used instead of {**CTRL-C**}, because it will reset the Function Keys and Color to their default values.

Running STARTREK on the Z-100
As mentioned above, this version of STARTREK uses ANSI Graphics to display the Short Range Scan. ANSI and GRAPHICS can be toggled ON and OFF, if desired.

To be able to use ANSI Graphics, the ZANSI.DVD driver is required to be installed in the Root Directory of the boot disk and the

DRIVER=ZANSI.DVD

statement must be included in the CONFIG.SYS file, also in the Root directory. CONFIG.SYS is run automatically during boot up. Both files are included on the distribution disk from "Z-100 LifeLine".

There is also a STARTREK.BAT Batch file to run the DOS CTTY ANSI command to begin STARTREK directly from ZBASIC, and return DOS to normal console input through the CTTY CON command when STARTREK ends.

Finally, ZBASIC included on the "Z-100 LifeLine" distribution disk is ZBASIC v1.1.

Note: The earliest version of ZBASIC v1.0, dated about 7/15/82, would stall operation of the game about 30-40 minutes into the game. It also included a second key stroke when Function Key <F10> was pressed that caused the "Illegal Request" error upon entry and an extra "COMMAND:" upon exit. Later versions of ZBASIC v1.0 and v1.1 work fine.

I have also created a "Cheat Sheet", STARTREK .RTF for beginners to be able to quickly learn all the Function Keys, without having to review the Help file directly. Just print this WordPad file and refer to it as necessary.

This version of STARTREK.BAS is available from the "Z-100 LifeLine" for the cost of shipping & handling. In addition to the files listed above, it has a few additional files and is bootable to ZDOS v3.10.
The additional files are:

ZANSI.DVD - The ZANSI device driver to display ANSI graphics on the Z-100.

ZBASIC v1.1 - The last version of the ZBASIC Interpreter.

CONFIG.SYS - A sample CONFIG.SYS containing the required driver loader.

STARTREK.BAT - A Batch file to run the DOS CTTY ANSI command to begin STARTREK directly from ZBASIC, and return DOS to normal console input through the CTTY CON command when STARTREK ends.

STARTREK.RTF - A cheat sheet for beginners to be able to quickly learn all the Function Keys, without having to review the Help file directly.

Enjoy,
Steven W. Vagts, Editor, Z-100 LifeLine

Operation

To give this program a try, just mark and bound the BASIC statements of STARTREK>BAS from the listing provided and copy it to your favorite document editor. Modify it as necessary to meet the formatting requirements for your particular version of BASIC. Remove any stray text that does not begin with a line number, then save the document with a .BAS file extension. Load it directly into your version of BASIC.

Remember, if you wish to use ANSI graphics, you must load the ZANSI.DVD device driver with CONFIG.SYS. Also, create the batch file, STARTREK.BAT containing the statements:

```
CTTY ANSI
ZBASIC STARTREK
CTTY CON
```

After booting up, run STARTREK.BAT to load and run the new program. Use {CTRL-C} to stop the program at any point and LIST to view the program, if desired. Use RUN to begin the program again.

If you have any questions or comments, please email me at:
z100lifeline@swvagts.com

Cheers,

Steven W. Vagts

ZBASIC STARTREK.BAS

```
10  REM ******************************************************
20  REM ******************************************************
30  REM **                                                  **
40  REM **      STARTREK - WRITTEN IN Z-BASIC FOR THE       **
50  REM **             ZENITH Z-100 COMPUTER                **
60  REM **     AND BASICA FOR IBM-PC/XT/AT COMPATIBLES      **
70  REM **             UPDATED IN PC-BASIC FOR              **
80  REM **          WINDOWS 7 AND NEWER COMPUTERS           **
90  REM **                                                  **
100 REM ******************************************************
110 REM ******************************************************
120 VERSION$="STARTREK  Version 11.0 of STARDATE 2603.26"
130 DEFAULTPLAYER$="GLENN D. FAINI"
140 Z100$="TRUE"   :REM SET TO "TRUE" FOR Z-100s, "FALSE" FOR PCs
150 SCRN%=2        :REM SET TO 2 FOR STD PC GRAPHICS, 8 FOR EXTD GRAPHICS
160 PAUSEFACTOR%=1 :REM SET TO 1 FOR 5MHz, 2 FOR 8MHz, 5 FOR 80286 COMPUTERS
170 ANSI$="TRUE"   :REM SET TO "FALSE" FOR Z-100 W/O ZANSI.DVD DRIVER.
180 REM ------------------------------------------------------
190 IF Z100$<>"TRUE" THEN 230
200 REM 8 COLOR TEXT
210 BLACK=0:BLUE=1:GREEN=2:CYAN=3
220 RED=4:MAGENTA=5:YELLOW=6:WHITE=7:GOTO 260
230 REM 16 COLOR TEXT
240 BLACK=0:BLUE=9:GREEN=10:CYAN=11
250 RED=12:MAGENTA=13:YELLOW=14:WHITE=15
260 CLS
270 PRINT "DO YOU WANT GRAPHICS (YES OR NO)";
280 A$=INKEY$
290 IF A$="" THEN 280 ELSE GOSUB 15650
300 IF A$="Y" THEN GRAPHICS$="YES":GOTO 330
310 IF A$="N" THEN GRAPHICS$="NO":GOTO 330
320 GOTO 280
330 IF Z100$<>"TRUE" THEN 420
340 REM TEST FOR COLOR CHIPS IN Z-100s
350 CLS:PSET (10,10),1
360 X=POINT(10,10)
370 IF X=1 THEN 400
375 REM MONOCHROME TEXT
380 BLUE=7:GREEN=7:CYAN=7
390 RED=7:MAGENTA=7:YELLOW=7:WHITE=7
400 CLS:KEY OFF:COLOR WHITE
410 GOTO 510
420 PRINT
430 PRINT "DO YOU WANT SOUND EFFECTS (YES OR NO)";
440 A$=INKEY$
450 IF A$="" THEN 440 ELSE GOSUB 15650
460 IF A$="Y" THEN SOUNDEFFECTS$="YES":GOTO 490
470 IF A$="N" THEN SOUNDEFFECTS$="NO":GOTO 490
480 GOTO 440
490 CLS:IF Z100$="TRUE" OR SOUNDEFFECTS$="NO" THEN 510
500 PLAY "MB O2 B. > E8 A2. G# E6 C#6 F#6 B2. B8 > D#1"
510 IF GRAPHICS$="NO" THEN 1350
520 IF Z100$<>"TRUE" THEN SCREEN SCRN%
530 CLS:KEY OFF
540 REM ------------------- S --------------------
550 LINE (0,112)-(75,112)
560 DRAW "m+5,-1m+3,-1m+2,-1m+0,-2m-2,-1m-3,-1m-5,-1"
570 DRAW "m-50,-13m-10,-3m-3,-2m-2,-2m+0,-3m+2,-2m+3,-2m+10,-2"
580 DRAW "m+60,0m-5,5m-40,0m-5,1m-2,2m-1,1m+5,2m+50,13"
590 DRAW "m+10,3m+3,2m+2,2m-0,4m-2,2m-3,2m-10,3m-75,0m-10,-5"
600 PAINT (40,90)
610 REM ------------------- T --------------------
620 PSET (93,75)
630 DRAW "m-5,5m+30,0m+0,37m+15,0m+0,-37m+30,0m+5,-5m-75,0"
640 PAINT (125,90)
650 REM ------------------- A --------------------
660 PSET (138,117)
670 DRAW "m+42,-42m+42,42m-15,0m-15,-15m-19,0m+5,-5"
680 PSET (139,117)
690 DRAW "m+14,0m+27,-27m+7,7m-9,0"
```

```
700 PAINT (190,90)
710 REM -------------------- R --------------------
720 PSET (230,117)
730 DRAW "nm+0,-42m+15,0m+0,-37m+10,0m+5,1m+4,2m+3,3"
740 DRAW "m-20,12m+8,0
750 CIRCLE (250,117),38,,0,1.4,.5
760 PSET (288,117)
770 DRAW "m+15,0"
780 CIRCLE (254,117),50,,0,1.2,.5
790 PSET (230,75)
800 DRAW "m+35,0
810 PSET (270,93)
820 DRAW "m+15,-7
830 CIRCLE (255,85),30,,0,1.2,.36
840 PAINT (240,90)
850 REM -------------------- T --------------------
860 PSET (313,75)
870 DRAW "m-5,5m+30,0m+0,37m+15,0m+0,-37m+30,0m+5,-5m-75,0"
880 PAINT (345,90)
890 REM -------------------- R --------------------
900 PSET (390,117)
910 DRAW "nm+0,-42m+15,0m+0,-37m+10,0m+5,1m+4,2m+3,3"
920 DRAW "m-20,12m+8,0
930 CIRCLE (410,117),38,,0,1.4,.5
940 PSET (448,117)
950 DRAW "m+15,0"
960 CIRCLE (414,117),50,,0,1.2,.5
970 PSET (390,75)
980 DRAW "m+35,0
990 PSET (430,93)
1000 DRAW "m+15,-7
1010 CIRCLE (415,85),30,,0,1.2,.36
1020 PAINT (400,90)
1030 REM -------------------- E --------------------
1040 CIRCLE (522,96),60,,1.571,4.712,.35
1050 CIRCLE (522,96),43,,1.571,4.712,.35
1060 PSET (523,75)
1070 DRAW "m+15,0m-5,6m-10,0
1080 PSET (523,111)
1090 DRAW "m+15,0m-5,6m-10,0
1100 PAINT (475,90)
1110 PSET (523,91)
1120 DRAW "m+15,0m-5,6m-40,0m+5,-6m+25,0
1130 PAINT (523,93)
1140 REM -------------------- K --------------------
1150 PSET (548,117)
1160 DRAW "m+0,-42m+15,0m+0,42m-15,0
1170 PAINT (550,90)
1180 PSET (570,98)
1190 DRAW "m+0,-8m+27,-15m+20,0m-30,18
1200 CIRCLE (569,117),50,,0,1.2,.5
1210 PSET (618,117)
1220 DRAW "m-17,0"
1230 CIRCLE (550,117),50,,0,1.15,.41
1240 PAINT (580,90)
1250 PSET(639,111)
1260 DRAW "m-30,0m+0,+6m+20,0m+10,-6"
1270 PAINT (630,114)
1280 LOCATE 21,29:PRINT "REVISED & EMBELLISHED"
1290 LOCATE 22,39:PRINT "BY"
1300 LOCATE 23,33:PRINT "GLENN D. FAINI"
1310 FOR X=1 TO 1000*PAUSEFACTOR%
1320 NEXT
1330 REM ----------------------------------------
1340 REM MARK# STORES RED DAMAGE MARKER DOTS
1350 IF Z100$<>"TRUE" THEN 1450
1360 COLOR CYAN:CLS
1370 DIM MARK#(17)
1380 FOR I%=0 TO 5
1390     CIRCLE (10,10),I%,RED
1400 NEXT I%
1410 GET (4,0)-(16,16),MARK#
```

```
1420 CLS
1430 GOTO 1540
1440 REM ----------------------------------------
1450 IF GRAPHICS$="NO" THEN 1510
1460 DIM MARK#(21):CLS
1470 FOR I%=0 TO 5
1480     CIRCLE (10,10),I%,RED
1490 NEXT I%
1500 GET (4,0)-(16,16),MARK#
1510 SCREEN 0:COLOR WHITE:CLS
1520 REM ----------------------------------------
1530 REM ASSIGN COMMANDS TO FUNCTION KEYS
1540 KEY 1,"HELP"+CHR$(13)
1550 KEY 2,"WARP"+CHR$(13)
1560 KEY 3,"PHASER"+CHR$(13)
1570 KEY 4,"PHOTON"+CHR$(13)
1580 KEY 5,"SRS"+CHR$(13)
1590 KEY 6,"LRS"+CHR$(13)
1600 KEY 7,"DAMAGE"+CHR$(13)
1610 KEY 8,"STATUS"+CHR$(13)
1620 KEY 9,"SHIELD"+CHR$(13)
1630 KEY 10,"COMPT"+CHR$(13)
1640 IF Z100$<>"TRUE" THEN 1690
1650 REM ----------------------------------------
1660 KEY 11,"SOUND"+CHR$(13)
1670 KEY 12,"GRAPHICS"+CHR$(13)
1680 REM ----------------------------------------
1690 PRINT "INSTRUCTIONS (YES OR NO)";
1700 A$=INKEY$
1710 IF A$="" THEN 1700 ELSE GOSUB 15650
1720 IF A$="Y" THEN 1750
1730 IF A$="N" THEN 1940
1740 GOTO 1700
1750 ON ERROR GOTO 1880
1760 OPEN "I",#1,"STARTREK.HLP"
1770 CLS:COLOR CYAN:KEY OFF
1780 LINE INPUT #1,HELP$
1790 IF LEFT$(HELP$,1) = "|" THEN GOTO 1820
1800 COLOR WHITE:PRINT HELP$
1810 GOTO 1780
1820 LOCATE 25,1 : PRINT "PRESS 'E' TO EXIT HELP, ANY OTHER KEY TO CONTINUE";
1830 HELP$=INKEY$
1840 IF HELP$="" THEN 1830
1850 IF HELP$="E" OR HELP$="e" THEN ERROR 62 :ELSE 1770
1860 REM ERROR 62 INPUT PAST END
1870 REM ERROR 53 FILE NOT FOUND
1880 IF ERR=62 THEN CLOSE #1 : RESUME 1940
1890 IF ERR=53 THEN PRINT "NO HELP FILE AVAILIBLE"
1900 FOR PAUSE=1 TO 1000*PAUSEFACTOR%:NEXT PAUSE:RESUME 1940
1910 REM END PROGRAM GRACEFULLY ON UNEXPECTED ERROR
1920 COLOR RED:PRINT "ERROR";ERR;"AT LINE";ERL : GOTO 15430
1930 REM ----------------------------------------
1940 REM START OF GAME PROGRAMMING
1950 KEY ON
1960 RESTORE: REM RESETS DATA POINTER
1970 REM SEED RANDOM NUMBER GENERATOR WITH CURRENT TIME SECONDS+MINUTES
1980 RANDOMIZE (VAL(MID$(TIME$,7,2)+MID$(TIME$,4,2)))
1990 COLOR WHITE:CLS
2000 PRINT VERSION$
2010 PRINT "          Revised & Embellished by Glenn D. Faini":PRINT
2020 COLOR CYAN:INPUT "PLEASE ENTER YOUR FULL NAME FOR THE LOG: ",PLAYER$
2030 IF LEN(PLAYER$)=0 THEN PLAYER$=DEFAULTPLAYER$:LOCATE 4,42:PRINT PLAYER$
2040 REM E(1) IS KLINGON PERCENTAGE
2050 E(1)=.25:PRINT
2060 REM ALLOCATE MEMORY FOR ARRAYS
2070 DIM X(4),U(10),N(121,9),A(3)
2080 REM U(8) IS HIGH SCORE, U(9) IS KLINGONS DESTROYED
2090 U(8)=0 : U(9)=0
2100 ON ERROR GOTO 2180
2110 OPEN "I",#1,"STARTREK.REC"
2120 ON ERROR GOTO 0 : REM ERROR TRAPPING TURNED OFF
2130 REM U(8)=HIGH SCORE, U(9)=KLINGONS DESTROYED, T=STARDATES
```

```
2140 INPUT#1,U(8),U(9),T
2150 IF U(8)<>0 THEN 2230
2160 CLOSE #1
2170 GOTO 2200
2180 RESUME 2200
2190 REM HIGH SCORE FILE STARTREK.REC DOES NOT EXIST
2200 PRINT "THERE IS NO RECORD RATING CURRENTLY ON FILE":PRINT
2210 COMMAND$=" STARFLEET COMMAND"
2220 GOTO 2340
2230 PRINT PLAYER$;", YOU ARE TRYING TO BEAT AN INDEX RATING OF";U(8);"GOTTEN BY"
2240 REM RANK$=RANK AND RECORD HOLDER NAME
2250 LINE INPUT#1,RANK$
2260 COMMAND$=" "+RANK$
2270 REM SHIP$=RECORD SHIP, D$=RECORD DATE, T$=RECORD TIME
2280 INPUT#1,SHIP$,D$,T$
2290 CLOSE #1
2300 PRINT RANK$;" OF THE STARSHIP U.S.S. ";SHIP$
2310 PRINT "AGAINST A FLEET OF";U(9);"KLINGON WARSHIPS"
2320 PRINT "IN";(INT(T*100))/100;"STARDATES AT ";
2330 PRINT T$;" HOURS ON STARDATE ";D$
2340 PRINT:PRINT "DO YOU WANT TO ALTER THE KLINGON CREATION PROBABILITIES?  ";
2350 A$=INKEY$
2360 IF A$="" THEN 2350 ELSE GOSUB 15650
2370 ON ERROR GOTO 15200: REM DETECTOR SECTION
2380 IF A$="Y" THEN 2430
2390 IF A$="N" THEN 2410
2400 GOTO 2350
2410 PRINT A$:PRINT:COLOR GREEN:PRINT "WORKING . . .":COLOR CYAN
2420 GOTO 2490
2430 PRINT A$ : PRINT
2440 PRINT "WHAT PERCENTAGE OF THE QUADRANTS SHOULD RECEIVE KLINGONS?"
2450 PRINT "NOTE:  %=0.25.  ENTER A VALUE BETWEEN 0.03 AND 1.0."
2460 INPUT"PERCENTAGE = ",E(1):PRINT:COLOR GREEN:PRINT"WORKING . . .":COLOR CYAN
2470 IF E(1)<.03 GOTO 2450
2480 IF E(1)>1 THEN E(1)=1
2490 DIM DESTROYED$(20),KLINGONCAUSE$(7),KLINGONRESULT$(4),COMMAND$(20)
2500 DIM QUADRANT$(121),SRS$(7),PROMOTION$(4),DAMAGE$(8)
2510 DIM KLINGONPREFIX$(5),KLINGONSUFFIX$(5),SHIP$(14)
2520 DIM F(20),B(25,6),W(11),Z(3),P(60),G(11,11),Q(10,10),D(8),K(9,4),C(2)
2530 DEF FND(A,B,C,D)=SQR((A-C)^2+(B-D)^2)
2540 DEF FNZ(E,A,B,C,D)=INT(E*(RND+2)/FND(A,B,C,D))
2550 DEF FNS(I,J)=FNM(G(I,J),10)
2560 DEF FNK(I,J)=FNM(G(I,J)/100,10)
2570 DEF FNA=D(1)+D(2)+D(3)+D(4)+D(5)+D(6)+D(7)+D(8)
2580 DEF FNB(I,J)=FNM(G(I,J)/10,10)
2590 DEF FNM(I,J)=INT(I)-INT(INT(I)/J)*J
2600 DEF FNH(E,A,B,C,D)=INT(E/FND(A,B,C,D)^2)-40
2610 DEF FNX=INT(RND*10+1)
2620 FOR L%=1 TO 121
2630     FOR K%=1 TO 9
2640         N(L%,K%)=0
2650 NEXT K%,L%
2660 A(1)=0 : A(2)=0 : A(3)=0
2670 FOR L%=1 TO 20
2680     READ DESTROYED$(L%),COMMAND$(L%)
2690 NEXT L%
2700 DATA "DESTROYS","HEL"
2710 DATA "INCINERATES","WAR"
2720 DATA "DISINTEGRATES","PHA"
2730 DATA "VAPORIZES","PHO"
2740 DATA "KILLS","SRS"
2750 DATA "DEMOLISHES","LRS"
2760 DATA "BLOWS UP","DAM"
2770 DATA "ANNIHILATES","STA"
2780 DATA "WRECKS","SHI"
2790 DATA "EXTINGUISHES","COM"
2800 DATA "SMASHES","MAP"
2810 DATA "OBLITERATES","KLI"
2820 DATA "ELIMINATES","BAS"
2830 DATA "ATOMIZES","OFF"
2840 DATA "PULVERIZES","TIM"
2850 DATA "CRUSHES","COU"
```

```
2860 DATA "DOES IN","SAV"
2870 DATA "LIQUIDATES","RET"
2880 DATA "ENDS","RAT"
2890 DATA "CRUNCHES","PHA"
2900 FOR L%=1 TO 8
2910     READ DAMAGE$(L%)
2920     D(L%)=0
2930 NEXT L%
2940 DATA "SHIELDS","WARP DRIVE","PHASER CONTROL","TORPEDO TUBES"
2950 DATA "S.R. SENSORS","L.R. SENSORS","COMPUTER","DILITHIUM CRYSTALS"
2960 B0=2 : U(10)=2 : REM INITIAL RANK IS ENSIGN
2970 REM T IS STARDATE = CURRENT DATE YYMM.DD
2980 T=VAL(RIGHT$(DATE$,2)+LEFT$(DATE$,2)+"."+MID$(DATE$,4,2)) : T0=T : T9=T
2990 FOR L%=1 TO 7
3000     READ KLINGONCAUSE$(L%),SRS$(L%)
3010 NEXT L%
3020 DATA "OVERLOADED ITS ENGINES","."
3030 DATA "CAME TOO CLOSE TO A PULSAR","E"
3040 DATA "HIT A STAR","K"
3050 DATA "HIT A NUCLEAR SPACE MINE","B"
3060 DATA "WAS SABOTAGED","D"
3070 DATA "STRAYED INTO KZINTI SPACE","*"
3080 DATA "STRUCK SPACE DEBRIS","O"
3090 IF ANSI$="TRUE" AND Z100$<>"TRUE" THEN SRS$(6)=CHR$(15)
3100 IF ANSI$="TRUE" THEN SRS$(1)=CHR$(197)
3110 E=INT((1530*RND+3000)*(E(1)*RND*1.4+1)) : E0=E
3120 FOR L%=1 TO 14
3130     READ SHIP$(L%)
3140 NEXT L%
3150 DATA "CONSTELLATION NCC-1017A"
3160 DATA "REPUBLIC      NCC-1371A"
3170 DATA "CONSTITUTION  NCC-1700A"
3180 DATA "ENTERPRISE    NCC-1701A"
3190 DATA "FARRAGUT      NCC-1702A"
3200 DATA "LEXINGTON     NCC-1703A"
3210 DATA "YORKTOWN      NCC-1704A"
3220 DATA "EXCALIBUR     NCC-1705A"
3230 DATA "EXETER        NCC-1706A"
3240 DATA "HOOD          NCC-1707A"
3250 DATA "INTREPID      NCC-1708A"
3260 DATA "VALIANT       NCC-1709A"
3270 DATA "CONGO         NCC-1710A"
3280 DATA "POTEMPKIN     NCC-1711A"
3290 REM CHOOSE RANDOM SHIP
3300 L%=INT(RND*14+1)
3310 SHIP$=SHIP$(L%)
3320 IF LEN(SHIP$)=0 THEN 3300
3330 Z2=INT(200*RND+250)
3340 FOR I%=1 TO 121
3350     QUADRANT$(I%)="   "
3360 NEXT I%
3370 FOR L%=1 TO 4
3380     READ PROMOTION$(L%),KLINGONRESULT$(L%)
3390 NEXT L%
3400 DATA "DUE TO VALOR UNDER FIRE","AND WAS DESTROYED"
3410 DATA "SINCE YOU ARE DOING SO WELL","AND IS NOW JUNK"
3420 DATA "BECAUSE OF YOUR BRILLIANT STRATEGY","AND WAS VAPORIZED"
3430 DATA "DUE TO ROUTINE PROMOTION PROCEDURES","AND WAS DISINTEGRATED"
3440 FOR L%=1 TO 5
3450     READ KLINGONPREFIX$(L%),KLINGONSUFFIX$(L%)
3460 NEXT L%
3470 DATA "KAR","NOV","GLAR","NOK","LLA","NVAR","PIER","MOS","VIER","NVAL"
3480 DIM RANK$(8)
3490 FOR L%=1 TO 8
3500     READ RANK$(L%)
3510 NEXT L%
3520 DATA "CREWMAN","ENSIGN","LIEUTENANT","COMMANDER"
3530 DATA "CAPTAIN","COMMODORE","ADMIRAL","FLEET ADMIRAL"
3540 RANK$=RANK$(U(10))
3550 P=INT((10*RND+5)*(E(1)*RND*1.7+1)) : P0=P
3560 FOR L%=1 TO 11
3570     FOR K%=1 TO 11
```

```
3580         G(L%,K%)=0
3590 NEXT K%,L%
3600 I=INT(RND*11+1) : J=INT(RND*11+1) : G(I,J)=10 : B9=1 : G7=1
3610 F1=0 : F2=0 : K5=0 : K9=0
3620 FOR I=1 TO 25
3630     B(I,5)=INT((16000+RND*1600)*(E(1)*RND*8+1))
3640     B(I,6)=INT((20+RND*30)*(E(1)*RND*7+1))
3650 NEXT I
3660 FOR I=1 TO 11
3670     FOR J=1 TO 11
3680         K3=0 : B4=0
3690         IF RND>E(1) THEN 3790 :ELSE K3=1
3700         Q=11*(I-1)+J : N(Q,1)=180
3710         FOR K=2 TO 9
3720             IF RND>(((E(1)^2+1)-.1)-K*.1) THEN 3780 :ELSE K3=K3+1
3730             IF RND<.25 THEN N(Q,K)=1000:GOTO 3750
3740             IF RND>.46789 THEN N(Q,K)=180 :ELSE N(Q,K)=500
3750             IF N(Q,K)=1000 THEN A(1)=A(1)+1
3760             IF N(Q,K)=500 THEN A(2)=A(2)+1
3770         NEXT K
3780         K9=K9+K3
3790         IF RND<=.0654321+(E(1)/20) AND G(I,J)=0 THEN B3=1 :ELSE B3=0
3800         B9=B9+B3 : G(I,J)=G(I,J)+K3*100+B3*10+INT(RND*9+1)
3810 NEXT J,I
3820 K1=K9 : P6=K1-(K1/5+.01) : T9=INT((T0+K9+2-RND*K9/2)*10000)/10000
3830 CLS
3840 PRINT "STARDATE:",T0
3850 PRINT "FROM:",COMMAND$
3860 J=A(1) : K=A(2)
3870 PRINT "TO:"," FEDERATION STARSHIP U.S.S ";SHIP$
3880 SRS$(2)=LEFT$(SHIP$,1) : PRINT
3890 PRINT "     A FLEET OF";K1;"KLINGON WARSHIPS HAS INVADED FEDERATION"
3900 PRINT "TERRITORY.  YOU ARE TO HOLD OFF THEIR OFFENSIVE UNTIL OUR FLEET"
3910 PRINT "ARRIVES.  INTELLIGENCE REPORTS THAT THEIR FLEET CONSISTS OF";J
3920 PRINT "D-6 AND D-7 CLASS BATTLE CRUISERS (ENERGY=1000),";K;"F-5 CLASS"
3930 PRINT "FRIGATES (ENERGY=500), AND";K1-(J+K);"F-5S CLASS SCOUT SHIPS"
3940 PRINT "(ENERGY=180).  OUR";B9;"BASE STATIONS CAN WITHSTAND THE KLINGON"
3950 PRINT "ASSAULT UNTIL STARDATE";STR$(T9);".  AFTER THEN HOWEVER,"
3960 PRINT "THEY WILL BE ABANDONED.  GOOD LUCK!"
3970 K=0
3980 Q1=INT(RND*11+1) : K=K+1 : Q2=INT(RND*11+1)
3990 IF K=200 THEN 4030
4000 IF FNK(Q1,Q2)<>0 THEN 3980
4010 A(3)=1
4020 GOTO 4040
4030 A(3)=0
4040 S1=FNX : S2=FNX
4050 REM -------------------- SET UP QUADRANT --------------------
4060 K3=0 : R3=0 : S3=0
4070 FOR L%=1 TO 9
4080     FOR K%=1 TO 4
4090         K(L%,K%)=0
4100 NEXT K%,L%
4110 FOR L%=1 TO 10
4120     FOR K%=1 TO 10
4130         Q(L%,K%)=0
4140 NEXT K%,L%
4150 K0=FNK(Q1,Q2) : K3=K0
4160 B3=FNB(Q1,Q2)
4170 S3=FNS(Q1,Q2)
4180 IF B3=0 THEN 4290
4190 IF B4=0 THEN 4230
4200 FOR I=1 TO B4
4210     IF B(I,1)=Q1 AND B(I,2)=Q2 THEN 4270
4220 NEXT I
4230 R1=FNX : R2=FNX
4240 IF Q(R1,R2)<>0 THEN PRINT"ERROR: NO SUCH LINE":GOTO 15430
4250 B1=R1:B2=R2:B4=B4+1:B(B4,1)=Q1:B(B4,2)=Q2:B(B4,3)=B1:B(B4,4)=B2
4260 GOTO 4280
4270 B1=B(I,3) : B2=B(I,4)
4280 Q(B1,B2)=3
4290 Q(S1,S2)=Q(S1,S2)+1
```

```
4300 FOR I=1 TO S3
4310     R1=FNX : R2=FNX
4320     IF Q(R1,R2)<>0 THEN 4310
4330     Q(R1,R2)=5
4340 NEXT I
4350 K=1
4360 FOR I=1 TO K3
4370     R1=FNX : R2=FNX
4380     IF Q(R1,R2)<>0 THEN 4370
4390     Q(R1,R2)=2 : K(I,1)=R1 : K(I,2)=R2 : G7=11*(Q1-1)+Q2
4400     IF N(G7,K)>0 THEN 4420 :ELSE K=K+1
4410     IF N(G7,K)<>0 THEN 4400 :ELSE PRINT "ERROR K=";K;"G7=";G7;"N()=";N(G7,K)
4420     K(I,3)=N(G7,K) : K=K+1
4430     IF K(I,3)>500 THEN VESSEL$="BATTLE CRUISER"
4440     IF K(I,3)<=500 THEN VESSEL$="FRIGATE"
4450     IF K(I,3)<=180 THEN VESSEL$="SCOUT"
4460     PRINT "SENSORS SHOW A KLINGON ";VESSEL$;" OF ENERGY";K(I,3);"AT";STR$(R1);-R2
4470     G7=G7+1 : K(I,4)=G7-1
4480 NEXT I
4490 FOR PAUSE=1 TO 500*PAUSEFACTOR%:NEXT PAUSE
4500 GOTO 11620 : REM PROMOTION/ABANDON BASES
4510 GOSUB 14510 : REM STATUS REPORT
4520 GOSUB 15130 : REM END OF GAME CHECK
4530 IF U(5)=1 THEN 14430
4540 REM PRINT
4550 IF A(3)=0 OR K3=0 OR E(2)=0 THEN 4570
4560 GOSUB 4800 : REM SRS
4570 E(2)=0 : A(3)=1
4580 COLOR GREEN:INPUT "COMMAND: ",A$:COLOR CYAN
4590 A$=A$+"###" : A$=LEFT$(A$,3) : GOSUB 15650
4600 IF A$="END" THEN 15430
4610 IF A$="QUI" THEN 15430
4620 IF A$="SOU" THEN 4730
4630 IF A$="GRA" THEN 4750
4640 IF A$="ANS" THEN 4770
4650 FOR I=1 TO 10
4660     IF A$=COMMAND$(I) THEN 4700 : REM VALID COMMAND
4670 NEXT I
4680 COLOR RED:PRINT "ILLEGAL COMMAND ";RANK$:COLOR CYAN:PRINT
4690 GOTO 4550
4700 ON I GOTO 11950,5290,6460,7140,4710,6230,8410,4510,6950,12560
4710 GOSUB 4800 : REM SRS
4720 GOTO 4550
4730 IF SOUNDEFFECTS$="YES" THEN SOUNDEFFECTS$="NO" :ELSE SOUNDEFFECTS$="YES"
4740 GOTO 4550
4750 IF GRAPHICS$="YES" THEN GRAPHICS$="NO" :ELSE GRAPHICS$="YES"
4760 GOTO 4550
4770 IF ANSI$="TRUE" THEN ANSI$="FALSE":SRS$(1)="." :ELSE ANSI$="TRUE":SRS$(1)=CHR$(197)
4780 GOTO 4550
4790 REM -------------------- SHORT RANGE SCAN --------------------
4800 IF D(5)>=0 THEN 4840
4810 COLOR RED:PRINT "**SHORT RANGE SENSORS ARE OUT**" : K5=K5+1:COLOR CYAN
4820 PRINT
4830 RETURN
4840 PRINT "SRS FOR";STR$(Q1);-Q2
4850 FOR J=10 TO 1 STEP -1
4860     IF J=10 THEN TYPE1=201:TYPE2=209:TYPE3=187:TYPE4=205
4870     IF J=1  THEN TYPE1=200:TYPE2=207:TYPE3=188:TYPE4=205
4880     IF J>1 AND J<10 THEN TYPE1=199:TYPE2=197:TYPE3=182:TYPE4=196
4890     IF J=10 THEN PRINT STR$(J); :ELSE PRINT STR$(J);" ";
4900     K=Q(1,J)+Q(2,J)+Q(3,J)+Q(4,J)+Q(5,J)+Q(6,J)+Q(7,J)
4910     K=K+Q(8,J)+Q(9,J)+Q(10,J)
4920     IF K=0 THEN 5170
4930     FOR I=1 TO 10
4940         ON Q(I,J)+1 GOTO 4950,4960,4970,4980,4990,5000,5010
4950         COLOR CYAN:GOTO 5020    :REM EMPTY
4960         COLOR WHITE:GOTO 5020   :REM SHIP
4970         COLOR RED:GOTO 5020     :REM KLINGON
4980         COLOR MAGENTA:GOTO 5020 :REM BASE
4990         COLOR GREEN:GOTO 5020   :REM DOCKED
5000         COLOR YELLOW:GOTO 5020  :REM STAR
5010         COLOR BLUE              :REM
```

```
5020         IF ANSI$<>"TRUE" OR SRS$(Q(I,J)+1)<>CHR$(197) THEN 5110
5030         IF J=10 AND I=1 THEN PRINT CHR$(201);CHR$(205);:GOTO 5140
5040         IF J=10 AND I=10 THEN PRINT CHR$(187):GOTO 5140
5050         IF J=10 THEN PRINT CHR$(209);CHR$(205);:GOTO 5140
5060         IF J=1 AND I=1 THEN PRINT CHR$(200);CHR$(205);:GOTO 5140
5070         IF J=1 AND I=10 THEN PRINT CHR$(188):GOTO 5140
5080         IF J=1 THEN PRINT CHR$(207);CHR$(205);:GOTO 5140
5090         IF I=1 THEN PRINT CHR$(199);CHR$(196);:GOTO 5140
5100         IF I=10 THEN PRINT CHR$(182):GOTO 5140
5110         IF ANSI$<>"TRUE" THEN DASH$=" ":GOTO 5130
5120         IF J=10 OR J=1 THEN DASH$=CHR$(205) :ELSE DASH$=CHR$(196)
5130         PRINT SRS$(Q(I,J)+1);:COLOR CYAN:IF I=10 THEN PRINT :ELSE PRINT DASH$;
5140     NEXT I
5150     COLOR CYAN
5160     GOTO 5250
5170     IF ANSI$<>"TRUE" THEN 5240
5180     PRINT CHR$(TYPE1);
5190     FOR I%=1 TO 8
5200         PRINT CHR$(TYPE4);CHR$(TYPE2);
5210     NEXT I%
5220     PRINT CHR$(TYPE4);CHR$(TYPE3)
5230     GOTO 5250
5240     PRINT ". . . . . . . . . ."
5250 NEXT J
5260 PRINT "   1 2 3 4 5 6 7 8 9 10"
5270 PRINT
5280 RETURN
5290 REM -------------------- WARP DRIVE --------------------
5300 INPUT "COURSE,WARP-FACTOR: ";X,Y,W
5310 X=INT(X*10+.001) : Y=INT(Y*10+.001)
5320 IF W>=0 THEN 5480
5330 COLOR RED:PRINT "ATTEMPT TO USE A NEGATIVE WARP FACTOR HAS RESULTED IN"
5340 PRINT "DAMAGE TO THE WARP DRIVE, AND HULL DAMAGE ON DECKS 4 AND 7."
5350 COLOR CYAN
5360 IF RND>(.1*(W-8)) THEN 5440
5370 COLOR RED
5380 PRINT "WARP DRIVE HAS OVERLOADED.....    .....   ... ..  ..    ."
5390 COLOR CYAN
5400 R=-1
5410 GOSUB 15230 : REM SAVE GAME
5420 CLS
5430 GOTO 15430 : EXIT GAME
5440 K5=K5+1000 : D(2)=-5
5450 IF U(10)=1 THEN 5470 :ELSE U(10)=U(10)-1 : RANK$=RANK$(U(10))
5460 PRINT "YOU ARE DEMOTED TO ";RANK$
5470 GOTO 4520 : REM END OF GAME CHECK
5480 IF ABS(X)+ABS(Y)=0 THEN 4520 : REM END OF GAME CHECK
5490 N$="IT'S RUNNING OUT OF ENERGY"
5500 IF D(2)>=0 THEN 5560
5510 IF W<=.5 THEN 5560
5520 COLOR RED:PRINT "WARP ENGINES DAMAGED."
5530 PRINT "IMPULSE ENGINES AVAILABLE FOR ONLY 1/2 OF LIGHT":COLOR CYAN
5540 K5=K5+1
5550 GOTO 5290
5560 IF W<=8 THEN 5600
5570 COLOR RED:PRINT "MAXIMUM EMERGENCY SPEED OF WARP 8 SURPASSED!"
5580 K5=K5+1
5590 GOTO 5360
5600 V4=1
5610 GOTO 12120 : REM FIND PATH BLOCKAGE
5620 IF Q(X5,Y5)=3 THEN 5660
5630 PRINT SHIP$;" COURSE BLOCKED AT";STR$(X5);-Y5
5640 GOTO 4520 : REM END OF GAME CHECK
5650 REM -------------------- ENTRY FOR RETURN --------------------
5660 Q(S1,S2)=Q(S1,S2)-1 : I=SQR((X/10)^2+(Y/10)^2)
5670 CLS
5680 IF W<>0 THEN 5710
5690 T=T+1825 : REM 5 YEARS ELAPSED
5700 GOTO 5720
5710 T=T+I/W+.1 : REM ELAPSED TIME
5720 E=E-INT(9*W*I) : REM ENERGY USED
5730 IF E>0 THEN 5750
```

```
5740 Z2=Z2+E : E=0
5750 S1=S1+X : S2=Y+S2
5760 IF S1<11 AND S1>0 AND S2<11 AND S2>0 THEN 5920
5770 S1=S1-X : S2=S2-Y : Q1=10*Q1+(S1-1)+X : Q2=10*Q2+(10-S2)-Y
5780 IF Q1>120 OR Q1<10 OR Q2>120 OR Q2<10 THEN 5820
5790 X=10*INT(Q1/10) : Y=10*INT(Q2/10) : S1=INT(Q1-X)+1 : S2=10-INT(Q2-Y)
5800 Q1=INT(X/10) : Q2=INT(Y/10)
5810 GOTO 4060 : REM SET UP QUADRANT
5820 COLOR RED:PRINT SHIP$;" REBOUNDS FROM THE GALAXY BARRIER!"
5830 IF U(10)=1 THEN 5850 :ELSE U(10)=U(10)-1 : RANK$=RANK$(U(10))
5840 PRINT:PRINT "YOU ARE DEMOTED TO ";RANK$:PRINT
5850 N$="A NAVIGATION ERROR":K5=K5+100:Q1=INT(RND*11+1):Q2=INT(RND*11+1)
5860 COLOR CYAN
5870 T=T+INT(RND*8-5)
5880 IF RND>.75 THEN 4060 : REM SET UP QUADRANT
5890 Z1=INT(RND*1000+1)
5900 GOSUB 13610 : REM DAMAGE SECTION
5910 GOTO 4040 : REM SECTOR CALCULATION
5920 Q(S1,S2)=Q(S1,S2)+1
5930 GOTO 11620 : REM PROMOTION/ABANDON BASES
5940 REM -------------------- DAMAGE SECTION --------------------
5950 GOSUB  13610
5960 REM -------------------- KLINGON ACCIDENT --------------------
5970 IF RND>.06 THEN 6210 : REM NO ACCIDENT
5980 I=INT(RND*11+1) : J=INT(RND*11+1)
5990 IF INT(G(I,J)/100)=0 THEN 6210 : REM NO ACCIDENT
6000 IF I=Q1 AND J=Q2 THEN 6210 : REM NO ACCIDENT
6010 G7=11*(I-1)+J : K=FNK(I,J) : ACCIDENT$=" SCOUT "
6020 FOR R=K TO 1 STEP -1
6030     IF N(G7,R)>0 THEN 6060
6040 NEXT R
6050 GOTO 6210
6060 IF N(G7,R)=500 THEN ACCIDENT$=" FRIGATE "
6070 IF N(G7,R)=1000 THEN ACCIDENT$=" BATTLE CRUISER "
6080 N(G7,R)=-1
6090 PRINT "STARDATE:",T
6100 PRINT "TO:","FEDERATION STARSHIP U.S.S. ";SHIP$
6110 PRINT "FROM:",COMMAND$
6120 PRINT "     WE HAVE LEARNED THAT A KLINGON";ACCIDENT$;"IN"
6130 PRINT "QUADRANT";STR$(I);-J;KLINGONCAUSE$(INT(RND*7+1));" ";
6140 PRINT KLINGONRESULT$(INT(RND*4+1));"."
6150 PRINT "THIS LEAVES";K9-1;"KLINGON VESSELS.":PRINT
6160 Z(1)=VAL(LEFT$(QUADRANT$((J-1)*11+I),1))
6170 IF Z(1)=0 THEN 6200
6180 Z(1)=Z(1)-1
6190 QUADRANT$((J-1)*11+I)=RIGHT$(STR$(Z(1)),1)+RIGHT$(QUADRANT$((J-1)*11+I),2)
6200 G(I,J)=G(I,J)-100 : K9=K9-1
6210 IF A$="WAR" THEN 4510 : REM STATUS REPORT
6220 GOTO 4520 : REM END OF GAME CHECK
6230 REM -------------------- LONG RANGE SCAN --------------------
6240 IF D(6)>=0 THEN 6270
6250 COLOR RED:PRINT "LONG RANGE SENSORS ARE INOPERABLE" : K5=K5+1:COLOR CYAN
6260 GOTO 4520 : REM END OF GAME CHECK
6270 PRINT "LRS FOR";STR$(Q1);-Q2
6280 FOR I=Q2-1 TO Q2+1
6290     FOR J=Q1-1 TO Q1+1
6300         IF I<=11 AND J<=11 AND I>=1 AND J>=1 THEN 6330
6310         PRINT " <*>";
6320         GOTO 6420
6330         Z$=STR$(G(J,I))
6340         Z(0)=LEN(Z$)
6350         IF Z(0)=>4 THEN 6400
6360         IF Z(0)=3 THEN 6390
6370         Z$="00"+RIGHT$(Z$,1)
6380         GOTO 6400
6390         Z$="0"+RIGHT$(Z$,2)
6400         Z$=RIGHT$(Z$,3)
6410         PRINT " ";Z$; : QUADRANT$((I-1)*11+J)=Z$
6420     NEXT J
6430     PRINT
6440 NEXT I
6450 GOTO 4520 : REM END OF GAME CHECK
```

```
6460 REM -------------------- PHASER CONTROL --------------------
6470 IF Q(S1,S2)=4 THEN 6910 : REM DOCKED
6480 IF D(3)>=0 THEN 6510
6490 COLOR RED:PRINT "PHASERS NOT OPERATIONAL." : K5=K5+1:COLOR CYAN
6500 GOTO 4520 : REM END OF GAME CHECK
6510 IF K3>0 THEN 6570
6520 PRINT "NO KLINGONS"
6530 GOTO 4520 : REM END OF GAME CHECK
6540 COLOR RED:PRINT "TOO MUCH ENERGY, PHASERS OVERLOADED.":COLOR CYAN:D(3)=-2
6550 K5=K5+10 : E=E-X:FOR PAUSE=1 TO 1000*PAUSEFACTOR%:NEXT
6560 GOTO 6630
6570 PRINT "PHASERS LOCKED ON.  ENERGY AVAILABLE: ";E;
6580 INPUT X
6590 IF X<=0 THEN 4520 : REM END OF GAME CHECK
6600 IF E-X<0 THEN 6510 : REM NOT ENOUGH ENERGY
6610 IF X>500 THEN 6540 : REM PHASERS OVERLOADED
6620 E=E-X
6630 CLS:PRINT"PHASERS . . . FIRE!"
6640 IF Z100$="TRUE" OR SOUNDEFFECTS$="NO" THEN 6740
6650 FOR SHOT=1 TO 2
6660     FOR PHASER=1 TO 30
6670         SOUND  987,.11
6680         SOUND 1318,7.500001E-02
6690         SOUND 1975,.15
6700         SOUND 2975,.09
6710     NEXT PHASER
6720     FOR PAUSE=1 TO 250*PAUSEFACTOR%:NEXT PAUSE
6730 NEXT SHOT
6740 FOR I=1 TO K0
6750     IF K(I,3)<=0 THEN 6870
6760     H=FNZ(X,S1,S2,K(I,1),K(I,2)) : K(I,3)=K(I,3)-H
6770     IF K(I,3)<=0 THEN 6810
6780     PRINT H;"HIT REDUCES KLINGON AT";STR$(K(I,1));-K(I,2);
6790     PRINT "TO";K(I,3);"ENERGY"
6800     GOTO 6870
6810     PRINT H;"HIT ";DESTROYED$(INT(RND*20+1));" KLINGON AT";STR$(K(I,1));-K(I,2)
6820     N((11*(Q1-1)+Q2),I)=-1
6830     V$=QUADRANT$((Q2-1)*11+Q1)
6840     IF "0"=LEFT$(V$,1) THEN GOTO 6860
6850     QUADRANT$((Q2-1)*11+Q1)=RIGHT$(STR$(VAL(LEFT$(V$,1))-1),1)+RIGHT$(V$,2)
6860     K3=K3-1:K9=K9-1:V=K(I,1):W=K(I,2):Q(V,W)=0:G(Q1,Q2)=G(Q1,Q2)-100
6870 NEXT I
6880 GOTO 10740 : REM KLINGON ATTACK
6890 GOTO 4520 : REM END OF GAME CHECK
6900 REM -------------------- DOCKED FIRING --------------------
6910 COLOR RED:PRINT "BASE STATION PERSONNEL ENDANGERED ";RANK$:COLOR CYAN
6920 K5=K5+1
6930 GOTO 4520 : REM END OF GAME CHECK
6940 REM -------------------- SHIELDS --------------------
6950 IF D(1)>=0 THEN 6980 :ELSE COLOR RED:PRINT "SHIELDS ARE DAMAGED":COLOR CYAN
6960 K5=K5+1
6970 GOTO 4520 : REM END OF GAME CHECK
6980 PRINT "SHIELD STRENGTH (";Z2;"), ENERGY (";E;").  NEW STRENGTH";
6990 INPUT STRENGTH$:IF LEN(STRENGTH$)=0 THEN 4520 :ELSE J6=VAL(STRENGTH$)
7000 IF J6<=(E+Z2) THEN 7040
7010 COLOR RED:PRINT "SHIELDS REQUEST EXCEEDS ENERGY RESERVES ";RANK$:COLOR CYAN
7020 K5=K5+1
7030 GOTO 6980
7040 IF J6<0 THEN 4520 : REM END OF GAME CHECK
7050 IF J6<=4000 THEN 7100
7060 COLOR RED:PRINT "TOO MUCH ENERGY ";RANK$;",";J6-4000;"UNITS LOST!":COLOR CYAN
7070 FOR PAUSE=1 TO 1000*PAUSEFACTOR%:NEXT PAUSE
7080 K5=K5+1 : E=E-(J6-Z2) : Z2=4000
7090 GOTO 4520 : REM END OF GAME CHECK
7100 IF J6<Z2 THEN E=E+(Z2-J6) :ELSE E=E-(J6-Z2)
7110 Z2=J6
7120 GOTO 4520 : REM END OF GAME CHECK
7130 REM -------------------- PHOTON TORPEDOES --------------------
7140 IF Q(S1,S2)=4 THEN 6910 : REM DOCKED FIRING
7150 ON ERROR GOTO 15200
7160 IF D(4)>=0 THEN 7220
7170 COLOR RED:PRINT "TORPEDO TUBES ARE DAMAGED.":COLOR CYAN
```

```
7180 K5=K5+1
7190 GOTO 4520 : REM END OF GAME CHECK
7200 COLOR RED:PRINT "ALL PHOTON TORPEDOES EXPENDED":COLOR CYAN
7210 GOTO 10740 : REM KLINGON ATTACK
7220 IF P>5 THEN TUBE$="ALL SIX":GOTO 7290
7230 IF P=5 THEN TUBE$="FIVE":GOTO 7290
7240 IF P=4 THEN TUBE$="FOUR":GOTO 7290
7250 IF P=3 THEN TUBE$="THREE":GOTO 7290
7260 IF P=2 THEN TUBE$="TWO":GOTO 7290
7270 IF P=1 THEN TUBE$="ONE":GOTO 7290
7280 GOTO 7310
7290 PRINT TUBE$;" TUBE(S) REPORT READY."
7300 LINE INPUT "TARGETS (4-8, ...)",P$
7310 REM CLS
7320 P(0)=LEN(P$):N$="A SUICIDAL WEAPONS OFFICER"
7330 D8=6
7340 J4=0:TORP=0
7350 L(0)=0 : T%=0
7360 IF P=0 THEN 7200:IF P(0)=0 THEN 4520 : REM END OF GAME CHECK
7370 J4=J4+1
7380 IF J4>P(0) THEN 8370 : REM KLINGON ATTACK
7390 V$=MID$(P$,J4,1)
7400 IF " "=V$ THEN 7370
7410 IF V$<"0" OR V$>"9" THEN 7440
7420 L(0)=L(0)+1 : L(L(0))=VAL(V$)
7430 GOTO 7370
7440 IF ","=V$ THEN TORP=TORP+1 :ELSE 7460
7450 IF TORP=7 THEN 10740 :ELSE 7500
7460 IF "-"<>V$ THEN 8370
7470 FOR I%=1 TO L(0) : T%=T%+L(I%)*10^(L(0)-I%) : NEXT I%
7480 X=T%
7490 GOTO 7350
7500 FOR I%=1 TO L(0) : T%=T%+L(I%)*10^(L(0)-I%) : NEXT I%
7510 PRINT:COLOR WHITE:PRINT "FIRE ";TORP:COLOR CYAN
7520 Y5=T% : Y=T% : X5=X : P=P-1
7530 IF X>10 OR X<1 OR Y>10 OR Y<1 THEN 8380
7540 IF Q(X,Y)=0 THEN 8380
7550 IF Q(X,Y)=5 THEN 7920
7560 U(6)=X : U(7)=Y : X=X-S1 : Y=Y-S2
7570 IF X=0 AND Y=0 THEN 11170
7580 V4=2 : R=RND
7590 IF R>.05 THEN 7620
7600 COLOR RED:PRINT "TORPEDO EXPLODES ON SPACE DEBRIS":COLOR CYAN
7610 GOTO 8350
7620 IF R>.985 THEN 8380
7630 GOTO 12120
7640 ON Q(X5,Y5)-1 GOTO 7650,7870,11170,7930,7600
7650 FOR I=1 TO K0
7660     IF X5<>K(I,1) THEN 7680
7670     IF Y5=K(I,2) THEN 7700
7680 NEXT I
7690 GOTO 7600
7700 K(I,3)=INT(K(I,3)-(350+(RND*250+1))) : N((11*(Q1-1)+Q2),I)=K(I,3)
7710 IF K(I,3)>0 THEN 7790
7720 PRINT "KLINGON AT";STR$(X5);-Y5;"DESTROYED"
7730 Z(1)=VAL(LEFT$(QUADRANT$((Q2-1)*11+Q1),1))
7740 IF Z(1)=0 THEN 7770
7750 Z(1)=Z(1)-1
7760 QUADRANT$((Q2-1)*11+Q1)=RIGHT$(STR$(Z(1)),1)+RIGHT$(QUADRANT$((Q2-1)*11+Q1),2)
7770 K3=K3-1 : K9=K9-1
7780 GOTO 8330
7790 PRINT "TORPEDO HIT REDUCES KLINGON AT";STR$(K(I,1));-K(I,2);
7800 PRINT "TO";K(I,3);"ENERGY."
7810 IF RND>.5 THEN 7350
7820 K=K(I,1) : J=K(I,2)
7830 IF RND>.5 THEN K=INT(RND*10+1) :ELSE J=INT(RND*10+1)
7840 IF Q(K,J)<>0 THEN 8340
7850 Q(X5,Y5)=0 : K(I,1)=K : K(I,2)=J : Q(K,J)=2
7860 GOTO 8340
7870 COLOR RED:PRINT "A BASE STATION HAS BEEN DESTROYED BY THE U.S.S ";SHIP$
7880 U(10)=1 : RANK$=RANK$(U(10))
7890 PRINT:PRINT "YOU ARE DEMOTED TO ";RANK$:PRINT:COLOR CYAN
```

```
7900 K5=K5+1000 : B3=0 :B9=B9-1
7910 GOTO 8330
7920 U(6)=X : U(7)=Y : X=X-S1 : Y=Y-S2
7930 IF RND<.8 THEN GOTO 7960
7940 W9=FNH(40000!,S1,S2,U(6),U(7)):NOVAFACTOR=10
7950 COLOR RED:PRINT "STAR SUPERNOVAS!":COLOR CYAN:GOTO 7980
7960 W9=FNH(4000,S1,S2,U(6),U(7)):NOVAFACTOR=1
7970 COLOR RED:PRINT "STAR NOVAS":COLOR CYAN
7980 IF W9>0 THEN Z1=W9 :ELSE Z1=0
7990 FOR I=1 TO K0
8000     IF K(I,3)<=0 THEN 8210
8010     IF FND(U(6),U(7),K(I,1),K(I,2))<>0 THEN 8040
8020     W9=4000*NOVAFACTOR
8030     GOTO 8050
8040     W9=FNH(4000*NOVAFACTOR,U(6),U(7),K(I,1),K(I,2))
8050     IF W9>0 THEN H=W9 :ELSE H=0
8060     K(I,3)=K(I,3)-H
8070     IF K(I,3)<=0 THEN 8140
8080     IF H<>0 THEN 8110
8090     PRINT "NOVA HAS NO AFFECT ON KLINGON AT";STR$(K(I,1));-K(I,2)
8100     GOTO 8210
8110     PRINT "NOVA IMPACT OF";H;"REDUCES KLINGON AT";STR$(K(I,1));-K(I,2);
8120     PRINT "TO";K(I,3);"ENERGY."
8130     GOTO 8210
8140     PRINT "FORCE OF";H;DESTROYED$(INT(RND*20+1));" KLINGON AT";STR$(K(I,1));-K(I,2)
8150     Z(1)=VAL(LEFT$(QUADRANT$((Q2-1)*11+Q1),1))
8160     IF Z(1)=0 THEN 8190
8170     Z(1)=Z(1)-1
8180     QUADRANT$((Q2-1)*11+Q1)=RIGHT$(STR$(Z(1)),1)+RIGHT$(QUADRANT$((Q2-1)*11+Q1),2)
8190     K3=K3-1:K9=K9-1:V=K(I,1):W=K(I,2):Q(V,W)=0:G(Q1,Q2)=G(Q1,Q2)-100
8200     N((11*(Q1-1)+Q2),I)=-1
8210 NEXT I
8220 FOR PAUSE=1 TO 250*PAUSEFACTOR%:NEXT PAUSE
8230 IF Z1>0 THEN 8260
8240 PRINT "SENSORS SHOW NO AFFECT BY THE NOVA ON THE ";SHIP$;" ";RANK$"."
8250 GOTO 8320
8260 COLOR RED:PRINT "NOVA IMPACTS THE ";SHIP$;" WITH AN ENERGY OF";Z1:COLOR CYAN
8270 IF U(10)=1 THEN 8290 :ELSE U(10)=U(10)-1 : RANK$=RANK$(U(10))
8280 PRINT:PRINT "YOU ARE DEMOTED TO ";RANK$:PRINT
8290 K5=K5+100
8300 N$="THE EFFECTS OF A NOVA"
8310 GOSUB 13610
8320 S3=S3-1
8330 Q(X5,Y5)=0
8340 G(Q1,Q2)=(K3*100)+(B3*10)+S3
8350 IF J4<P(0) THEN 7350
8360 D8=5
8370 GOTO 10740 : REM KLINGON ATTACK
8380 PRINT "OFF TARGET"
8390 GOTO 8350 : REM KLINGON ATTACK
8400 REM -------------------- DAMAGE REPORT --------------------
8410 KEY OFF:J6=0
8420 FOR I=1 TO 8
8430     IF D(I)<0 THEN J6=J6+1
8440 NEXT I
8450 IF J6=0 OR GRAPHICS$="NO" THEN GOTO 8490 :ELSE J6=0
8460 IF Z100$<>"TRUE" THEN SCREEN SCRN%
8470 GOSUB 8880 : REM DRAW TOP VIEW
8480 GOSUB 9730 : REM DRAW SIDE VIEW
8490 PRINT "DAMAGE CONTROL REPORT":PRINT
8500 FOR I=1 TO 8
8510     IF D(I)=-1 THEN DAYS$="DAY" ELSE DAYS$="DAYS"
8520     IF D(I)<0 THEN PRINT DAMAGE$(I);TAB(26);"REPAIR STATE: ";ABS(D(I));DAYS$
8530     IF D(I)<0 THEN J6=J6+1
8540 NEXT I
8550 IF J6=0 THEN PRINT "NO DAMAGE":KEY ON:GOTO 4520 :ELSE GOTO 8610
8560 IF Z100$<>"TRUE" THEN 8590
8570 FOR I=1 TO 1000*PAUSEFACTOR%:NEXT I:LOCATE 24:FOR I=1 TO 25:PRINT:NEXT I:LOCATE 1
8580 KEY ON:GOTO 4520 : REM COMMAND INPUT
8590 FOR I=1 TO 1000*PAUSEFACTOR%:NEXT I:CLS
8600 SCREEN 0:KEY ON:GOTO 4520 : REM COMMAND INPUT
8610 IF GRAPHICS$="NO" THEN 8580
```

```
8620 Y%=0
8630 IF Z100$<>"TRUE" THEN Y%=-15
8640 IF D(1)<0 THEN PUT (460,Y%+20),MARK#:PUT (595,Y%+20),MARK#:PUT (15,Y%+160),MARK#
8650 IF D(2)<0 THEN PUT (604,Y%+110),MARK#:PUT (208,Y%+154),MARK#
8660 IF D(2)<0 THEN PUT (454,Y%+110),MARK#
8670 IF D(3)<0 THEN PUT (490,Y%+48),MARK#:PUT (529,Y%+28),MARK#:PUT (568,Y%+48),MARK#
8680 IF D(3)<0 THEN PUT (86,Y%+155),MARK#:PUT (55,Y%+153),MARK#
8690 IF D(4)<0 THEN PUT (149,Y%+174),MARK#
8700 IF D(5)<0 THEN PUT (86,Y%+167),MARK#
8710 IF D(6)<0 THEN PUT (149,Y%+186),MARK#
8720 IF D(7)<0 THEN PUT (520,Y%+40),MARK#:PUT (539,Y%+50),MARK#:PUT (80,Y%+150),MARK#
8730 IF D(8)<0 THEN PUT (515,Y%+90),MARK#:PUT (175,Y%+159),MARK#
8740 IF D(1)<0 THEN PUT (460,Y%+20),MARK#:PUT (595,Y%+20),MARK#:PUT (15,Y%+160),MARK#
8750 IF D(2)<0 THEN PUT (604,Y%+110),MARK#:PUT (208,Y%+154),MARK#
8760 IF D(2)<0 THEN PUT (454,Y%+110),MARK#
8770 IF D(3)<0 THEN PUT (490,Y%+48),MARK#:PUT (529,Y%+28),MARK#:PUT (568,Y%+48),MARK#
8780 IF D(3)<0 THEN PUT (86,Y%+155),MARK#:PUT (55,Y%+153),MARK#
8790 IF D(4)<0 THEN PUT (149,Y%+174),MARK#
8800 IF D(5)<0 THEN PUT (86,Y%+167),MARK#
8810 IF D(6)<0 THEN PUT (149,Y%+186),MARK#
8820 IF D(7)<0 THEN PUT (520,Y%+40),MARK#:PUT (539,Y%+50),MARK#:PUT (80,Y%+150),MARK#
8830 IF D(8)<0 THEN PUT (515,Y%+90),MARK#:PUT (175,Y%+159),MARK#
8840 LOCATE 24,1 : PRINT "PRESS ANY KEY";
8850 LOCATE 25,2 : PRINT "TO CONTINUE";
8860 IF INKEY$="" THEN 8620
8870 GOTO 8560 : REM EXIT DAMAGE REPORT
8880 REM -------------------- PRIMARY HULL --------------------
8890 CLS:PI=3.1415927#:X%=535:Y%=58
8900 IF Z100$<>"TRUE" THEN Y%=43
8910 FOR I%=31 TO 103 STEP 12
8920     CIRCLE (X%,Y%),I%
8930 NEXT I%
8940 FOR ANGLE=0 TO -2.125*PI STEP -PI/8
8950     CIRCLE (X%,Y%),101,,0,ANGLE
8960 NEXT ANGLE
8970 FOR I%=0 TO 30
8980     CIRCLE (X%,Y%),I%,0
8990 NEXT I%
9000 REM -------------------- BRIDGE --------------------
9010 CIRCLE (X%,Y%),31
9020 CIRCLE (X%,Y%),5
9030 CIRCLE (X%,Y%),6
9040 CIRCLE (X%,Y%),18,,0,PI
9050 LINE (X%+18,Y%)-(X%+11,Y%+10)
9060 LINE (X%-18,Y%)-(X%-11,Y%+10)
9070 CIRCLE (X%,Y%+10),9,,PI,0,.2
9080 CIRCLE (X%,Y%),12,,0,PI
9090 LINE (X%+12,Y%)-(X%+7,Y%+6)
9100 LINE -(X%-7,Y%+6)
9110 LINE -(X%-12,Y%)
9120 REM -------------------- IMPULSE ENGINES --------------------
9130 FOR I%=0 TO 13
9140     CIRCLE (X%,Y%+37),I%,0
9150 NEXT I%
9160 LINE (X%-15,Y%+42)-(X%+15,Y%+43),0,B
9170 CIRCLE (X%,Y%+37),13
9180 LINE (X%+13,Y%+37)-(X%+15,Y%+46)
9190 LINE -(X%-15,Y%+46)
9200 LINE -(X%-13,Y%+37)
9210 REM -------------------- WARP ENGINES --------------------
9220 X%=X%-75:Y%=Y%+40
9230 GOSUB 9300
9240 X%=X%+150
9250 GOSUB 9300
9260 GOTO 9460
9270 LINE (X%-11,Y%+17)-(X%+11,Y%+82),0,BF
9280 PRESET (X%,Y%+16),0
9290 PRESET (X%,Y%+15),0
9300 CIRCLE (X%,Y%+17),8,,0,PI
9310 LINE (X%+8,Y%+17)-(X%+8,Y%+82)
9320 LINE (X%-8,Y%+17)-(X%-8,Y%+82)
9330 CIRCLE (X%,Y%+82),25,,PI,0,3
```

```
9340 LINE (X%+8,Y%+17)-(X%+11,Y%+20)
9350 LINE -(X%+11,Y%+82)
9360 LINE (X%-8,Y%+17)-(X%-11,Y%+20)
9370 LINE -(X%-11,Y%+82)
9380 CIRCLE (X%,Y%+82),34,,PI,0,3
9390 LINE (X%-2,Y%+17)-(X%+2,Y%+32),,B
9400 CIRCLE (X%,Y%+32),4
9410 LINE (X%+11,Y%+82)-(X%+15,Y%+109)
9420 LINE -(X%+6,Y%+111)
9430 LINE (X%-11,Y%+82)-(X%-15,Y%+109)
9440 LINE -(X%-6,Y%+111)
9450 RETURN
9460 REM -------------------- ENGINE PYLONS --------------------
9470 X%=X%-75
9480 LINE (X%-64,Y%+47)-(X%-12,Y%+20)
9490 LINE (X%-64,Y%+48)-(X%-12,Y%+21)
9500 LINE (X%-64,Y%+77)-(X%-12,Y%+32)
9510 LINE (X%-64,Y%+69)-(X%-55,Y%+69)
9520 LINE (X%+64,Y%+47)-(X%+12,Y%+20)
9530 LINE (X%+64,Y%+48)-(X%+12,Y%+21)
9540 LINE (X%+64,Y%+77)-(X%+12,Y%+32)
9550 LINE (X%+64,Y%+69)-(X%+55,Y%+69)
9560 REM -------------------- SECONDARY HULL --------------------
9570 LINE (X%-28,Y%+2)-(X%-28,Y%+9)
9580 LINE -(X%-25,Y%+25)
9590 LINE (X%-21,Y%+40)-(X%-21,Y%+50)
9600 LINE -(X%-14,Y%+54)
9610 CIRCLE (X%-12,Y%+50),9,,PI,1.5*PI,1
9620 LINE (X%+28,Y%+2)-(X%+28,Y%+9)
9630 LINE -(X%+25,Y%+25)
9640 LINE (X%+21,Y%+40)-(X%+21,Y%+50)
9650 LINE -(X%+14,Y%+54)
9660 CIRCLE (X%+12,Y%+50),9,,1.5*PI,2*PI,1
9670 CIRCLE (X%,Y%+51),16,,1.25*PI,1.75*PI,.35
9680 CIRCLE (X%,Y%+55),16,,1.25*PI,1.75*PI,.35
9690 CIRCLE (X%,Y%+54),4
9700 CIRCLE (X%,Y%+54),2
9710 RETURN
9720 REM -------------------- SIDE VIEW --------------------
9730 X%=92:Y%=170
9740 IF Z100$<>"TRUE" THEN Y%=155
9750 REM -------------------- PRIMARY HULL --------------------
9760 LINE (X%-92,Y%-2)-(X%+92,Y%-2)
9770 LINE -(X%+92,Y%+2)
9780 LINE -(X%-87,Y%+2)
9790 CIRCLE (X%-87,Y%-2),5,,PI,1.5*PI,.9
9800 CIRCLE (X%-102,Y%-26),120,,1.6*PI,1.74*PI,.2
9810 CIRCLE (X%-82,Y%-26),120,,1.6*PI,1.71*PI,.2
9820 CIRCLE (X%+102,Y%-26),120,,1.29*PI,1.4*PI,.2
9830 CIRCLE (X%+82,Y%-26),120,,1.29*PI,1.4*PI,.2
9840 LINE (X%,Y%-8)-(X%,Y%-2)
9850 LINE (X%-20,Y%-8)-(X%+20,Y%-8)
9860 LINE -(X%+30,Y%-7)
9870 LINE (X%+65,Y%-3)-(X%+98,Y%-3)
9880 LINE -(X%+97,Y%+1)
9890 LINE -(X%+94,Y%+1)
9900 LINE (X%-20,Y%-8)-(X%-10,Y%-11)
9910 LINE -(X%+20,Y%-11)
9920 CIRCLE (X%+20,Y%-8),10,,.1*PI,.5*PI,.3
9930 LINE (X%+30,Y%-8)-(X%+31,Y%-8)
9940 LINE (X%-9,Y%-11)-(X%-4,Y%-13)
9950 LINE -(X%+13,Y%-13)
9960 LINE -(X%+13,Y%-11)
9970 LINE (X%-4,Y%-14)-(X%+4,Y%-14)
9980 LINE (X%-1,Y%-15)-(X%+1,Y%-15)
9990 CIRCLE (X%-72,Y%+27),120,,.3*PI,.4*PI,.2
10000 LINE (X%-43,Y%+3)-(X%-38,Y%+3)
10010 CIRCLE (X%+72,Y%+27),120,,.6*PI,.7*PI,.2
10020 LINE (X%+43,Y%+3)-(X%+38,Y%+3)
10030 LINE (X%-5,Y%+7)-(X%+5,Y%+7)
10040 LINE (X%-3,Y%+8)-(X%+3,Y%+8)
10050 REM -------------------- CONNECTING DORSAL --------------------
```

```
10060 LINE (X%+25,Y%+5)-(X%+40,Y%+5)
10070 LINE -(X%+60,Y%+10)
10080 LINE -(X%+60,Y%+13)
10090 LINE -(X%+75,Y%+12)
10100 LINE -(X%+115,Y%+12)
10110 LINE -(X%+92,Y%+2)
10120 REM -------------------- SECONDARY HULL --------------------
10130 LINE (X%+57,Y%+13)-(X%+60,Y%+16),,BF
10140 PRESET (X%+57,Y%+14):PRESET (X%+57,Y%+15)
10150 LINE (X%+60,Y%+16)-(X%+70,Y%+16)
10160 CIRCLE (X%+70,Y%+18),10,,.5*PI,PI,.2
10170 LINE (X%+45,Y%+22)-(X%+45,Y%+30)
10180 LINE (X%+47,Y%+21)-(X%+47,Y%+31)
10190 LINE (X%+49,Y%+20)-(X%+49,Y%+32)
10200 LINE (X%+49,Y%+20)-(X%+60,Y%+18)
10210 LINE (X%+49,Y%+32)-(X%+60,Y%+34)
10220 LINE (X%+60,Y%+18)-(X%+60,Y%+34)
10230 LINE (X%+60,Y%+25)-(X%+60,Y%+27),0
10240 LINE (X%+49,Y%+24)-(X%+70,Y%+23)
10250 LINE (X%+49,Y%+28)-(X%+70,Y%+29)
10260 CIRCLE (X%+110,Y%+17),50,,.555*PI,PI,.05
10270 CIRCLE (X%+110,Y%+33),50,,PI,1.44*PI,7.900001E-02
10280 LINE (X%+58,Y%+20)-(X%+58,Y%+36)
10290 PRESET (X%+58,Y%+25):PRESET (X%+58,Y%+27)
10300 LINE (X%+58,Y%+36)-(X%+70,Y%+36)
10310 LINE (X%+115,Y%+12)-(X%+125,Y%+15)
10320 LINE -(X%+135,Y%+16)
10330 LINE -(X%+100,Y%+15)
10340 LINE (X%+103,Y%+37)-(X%+160,Y%+33)
10350 CIRCLE (X%+260,Y%+33),100,,.7200001*PI,.99*PI,.095
10360 LINE (X%+199,Y%+25)-(X%+200,Y%+24)
10370 LINE -(X%+185,Y%+24)
10380 LINE -(X%+193,Y%+20)
10390 LINE -(X%+195,Y%+20)
10400 LINE -(X%+193,Y%+19)
10410 LINE -(X%+165,Y%+17)
10420 LINE (X%+192,Y%+20)-(X%+197,Y%+24)
10430 REM -------------------- ENGINE PYLONS --------------------
10440 LINE (X%+136,Y%+18)-(X%+183,Y%+2)
10450 LINE -(X%+223,Y%+2)
10460 LINE -(X%+156,Y%+19)
10470 LINE -(X%+136,Y%+18)
10480 REM -------------------- WARP ENGINE NACELLES --------------------
10490 LINE (X%+183,Y%+2)-(X%+163,Y%+2)
10500 LINE -(X%+160,Y%)
10510 LINE -(X%+118,Y%)
10520 LINE -(X%+105,Y%-6)
10530 LINE -(X%+105,Y%-8)
10540 LINE -(X%+107,Y%-9)
10550 LINE -(X%+305,Y%-9)
10560 LINE -(X%+285,Y%+1)
10570 LINE -(X%+260,Y%-1)
10580 LINE -(X%+245,Y%-1)
10590 LINE -(X%+240,Y%-2)
10600 LINE -(X%+125,Y%-2)
10610 LINE -(X%+115,Y%-6)
10620 LINE -(X%+115,Y%-6)
10630 LINE -(X%+135,Y%-6)
10640 LINE -(X%+150,Y%-2)
10650 LINE (X%+135,Y%-6)-(X%+240,Y%-6)
10660 CIRCLE (X%+224,Y%-1),20,,1.5*PI,1.95*PI,.14
10670 CIRCLE (X%+240,Y%-4),20,,1.5*PI,.5*PI,.12
10680 PAINT (X%+170,Y%-4)
10690 LINE (X%+305,Y%-9)-(X%+283,Y%+1)
10700 LINE (X%+297,Y%-4)-(X%+270,Y%-4)
10710 LINE (X%+135,Y%-10)-(X%+140,Y%-10)
10720 RETURN
10730 REM -------------------- KLINGON ATTACK --------------------
10740 IF K3>0 THEN 10770
10750 IF A$="WAR" THEN 5970 : REM KLINGON ACCIDENT
10760 GOTO 4520 : REM DETECTOR
10770 IF Q(S1,S2)<>4 THEN 10800
```

```
10780 PRINT "BASE STATION SHIELDS PROTECT THE ";SHIP$
10790 GOTO 5970
10800 COLOR RED:PRINT "","RED ALERT!!!!!!!!":COLOR CYAN
10810 IF Z100$="TRUE" OR SOUNDEFFECTS$="NO" THEN 10880
10820 FOR ALERT=1 TO 4
10830 FOR PAUSE=1 TO 300*PAUSEFACTOR%:NEXT PAUSE
10840     FOR FREQUENCY= 440 TO 900 STEP 15
10850     SOUND FREQUENCY,.5
10860     NEXT FREQUENCY
10870 NEXT ALERT
10880 IF A(3)=0 THEN 4510
10890 Z1=0
10900 PRINT "KLINGON ATTACKS FROM: ";
10910 N$="THE OVERWHELMING SUPERIORITY OF OUR FORCES"
10920 FOR I=1 TO K0
10930     IF K(I,3)<=0 THEN 10970
10940     PRINT STR$(K(I,1));-K(I,2);
10950     IF K(I,1)<>10 AND K(I,2)<>10 THEN PRINT " ";
10960     IF K(I,1)<>10 AND K(I,2)<>10 THEN PRINT "";
10970 NEXT I
10980 J=0
10990 PRINT : PRINT "WITH ENERGY HITS OF:"; : H1=0
11000 FOR I=1 TO K0
11010     K6=K(I,1) : K7=K(I,2)
11020     IF K(I,3)<=0 THEN 11120
11030     J=J+1 : H=FNZ(K(I,3),S1,S2,K(I,1),K(I,2)) : Z1=Z1+H
11040     PRINT TAB(17+(J*6));H;
11050     K(I,3)=K(I,3)-INT(H/10) : H1=H1+H
11060     IF K(I,3)<=10 THEN 11120
11070  REM -------------------- KLINGON MOVEMENT --------------------
11080     X5=INT(RND*5-2)+K6 : Y5=INT(RND*5-2)+K7
11090     IF X5<1 OR X5>10 OR Y5<1 OR Y5>10 THEN 11120
11100     IF Q(X5,Y5)<>0 THEN 11120
11110     Q(K6,K7)=0 : K(I,1)=X5 : K(I,3)=K(I,3)-5 : K(I,2)=Y5 : Q(X5,Y5)=2
11120 NEXT I
11130 PRINT
11140 REM -------------------- END OF GAME MESSAGE --------------------
11150 GOSUB 13610 : REM DAMAGE SECTION
11160 GOTO 5970 : REM KLINGON ACCIDENT
11170 FOR PAUSE=1 TO 1000*PAUSEFACTOR%:NEXT PAUSE:CLS
11180 COLOR RED:PRINT "TO KLINGON CENTRAL COMMAND, FROM COMMANDER ";
11190 PRINT KLINGONPREFIX$(INT(RND*5+1));KLINGONSUFFIX$(INT(RND*5+1))
11200 PRINT : PRINT "    DUE TO ";N$
11210 PRINT "THE U.S.S. ";SHIP$;" HAS BEEN DESTROYED"
11220 PRINT "LEAVING THE FEDERATION OPEN TO INVASION!":PRINT:COLOR CYAN
11230 R=-1
11240 GOSUB 15230 : REM SAVE GAME
11250 GOTO 15430 : EXIT GAME
11260 CLS:COLOR GREEN:PRINT "STARDATE:",T
11270 U(10)=U(10)+1 : RANK$=RANK$(U(10))
11280 PRINT "FROM:"," FEDERATION STARSHIP U.S.S. ";SHIP$
11290 PRINT "TO:",COMMAND$
11300 PRINT:PRINT "     I AM PLEASED TO REPORT THAT THE ";SHIP$
11310 PRINT "HAS COMPLETED ITS MISSION, AT A COST OF";
11320 PRINT C(1);"DEAD, AND";C(2);"INJURED." : PRINT:PRINT
11330 PRINT "STARDATE:",T
11340 PRINT "FROM:",COMMAND$
11350 PRINT "TO"," FEDERATION STARSHIP U.S.S. ";SHIP$
11360 PRINT:PRINT "     WE ACKNOWLEDGE YOUR MESSAGE, AND REPORT THAT"
11370 PRINT "THE KLINGON EMPIRE HAS REQUESTED THAT PEACE NEGOTIATIONS"
11380 PRINT "BE INITIATED WITH ALL DUE HASTE!"
11390 PRINT:PRINT "     DUE TO YOUR BRILLIANCE IN BATTLE, THAT ENABLED"
11400 PRINT "YOU TO DESTROY";K1;"KLINGON SHIPS IN THE"
11410 PRINT "TIME SPAN OF";T-T0;"STARDATES, YOU HAVE BEEN PROMOTED"
11420 PRINT "TO ";RANK$;", WITH AN INDEX RATING OF";
11430 R=INT(K1/(T-T0)*1000)-K5
11440 PRINT R:COLOR CYAN
11450 IF K1=U(9) THEN 11480 : REM ALL KLINGONS DESTROYED
11460 IF K1<=U(9)-15 THEN 11590 : REM SAVE GAME
11470 IF K1>U(9)+25 THEN 11490 : REM HIGH SCORE
11480 IF R<=U(8) THEN 11590 : REM SAVE GAME
11490 OPEN "O",#1,"STARTREK.REC"
```

```
11500 STARDATE$=RIGHT$(DATE$,2)+LEFT$(DATE$,2)+"."+MID$(DATE$,4,2)
11510 T$=LEFT$(TIME$,2)+MID$(TIME$,4,2)
11520 COLOR WHITE:PRINT:PRINT "THAT IS A NEW RECORD":COLOR CYAN
11530 U(10)=8:RANK$=RANK$(U(10))
11540 PRINT:PRINT "YOU HAVE BEEN PROMOTED TO ";RANK$;"!"
11550 PRINT#1,R,K1,T-T0
11560 PRINT#1,RANK$+" "+PLAYER$
11570 PRINT#1,SHIP$+","+STARDATE$+","+T$
11580 CLOSE #1
11590 GOSUB 15230 : REM SAVE GAME
11600 GOTO 15430 : REM END GAME
11610 REM -------------------- PROMOTION/ABANDON BASES --------------------
11620 IF K9>=P6 THEN 11700
11630 U(10)=U(10)+1 : RANK$=RANK$(U(10)) : J6=INT(RND*4+1)
11640 COLOR GREEN:PRINT "STARDATE:",T
11650 PRINT "FROM:",COMMAND$
11660 PRINT "TO:","FEDERATION STARSHIP U.S.S. ";SHIP$
11670 PRINT "     ";PROMOTION$(J6);"."
11680 PRINT "     YOU ARE PROMOTED TO ";RANK$;"!!!!"
11690 P6=P6-(K1/5+.01) : PRINT:COLOR CYAN
11700 IF T<T9 THEN 11890 : REM DAMAGE REPAIR
11710 IF B9<=0 THEN 11890 : REM DAMAGE REPAIR
11720 COLOR RED:PRINT "STARDATE:",T
11730 PRINT "FROM:",COMMAND$
11740 PRINT "TO:","FEDERATION STARSHIP U.S.S. ";SHIP$
11750 PRINT:PRINT "     YOU ARE HEREBY NOTIFIED THAT ALL BASE STATIONS"
11760 PRINT "IN YOUR SECTOR HAVE BEEN ABANDONED.  GOOD LUCK!!":COLOR CYAN
11770 B9=0 : B4=0
11780 FOR I=1 TO 11
11790     FOR J=1 TO 11
11800         G(I,J)=G(I,J)-(FNB(I,J)*10)
11810         V$=QUADRANT$((I-1)*11+J)
11820         V2$=MID$(V$,2,1)
11830         IF V2$=" " THEN 11860
11840         V2$="0"
11850         QUADRANT$((I-1)*11+J)=LEFT$(V$,1)+V2$+RIGHT$(V$,1)
11860 NEXT J,I
11870 IF Q(S1,S2)<>4 THEN 11890 : REM DAMAGE REPAIR
11880 Q(S1,S2)=1
11890 FOR I=1 TO 8
11900     IF D(I)=0 THEN 11920
11910     D(I)=D(I)+1
11920 NEXT I
11930 GOTO 10740 : REM KLINGON ATTACK
11940 REM -------------------- HELP --------------------
11950 ON ERROR GOTO 12060
11960 OPEN "I",#1,"STARTREK.HLP"
11970 KEY OFF:CLS:COLOR CYAN
11980 LINE INPUT #1,HELP$
11990 IF LEFT$(HELP$,1) = "|" THEN GOTO 12020
12000 COLOR WHITE:PRINT HELP$
12010 GOTO 11980
12020 LOCATE 25,1 : PRINT "PRESS 'E' TO EXIT HELP, ANY OTHER KEY TO CONTINUE";
12030 HELP$=INKEY$
12040 IF HELP$="" THEN 12030
12050 IF HELP$="E" OR HELP$="e" THEN ERROR 62 :ELSE 11970
12060 IF ERR=62 THEN CLOSE #1:CLS:RESUME 12100
12070 IF ERR=53 THEN PRINT "NO HELP FILE AVAILIBLE"
12080 FOR PAUSE=1 TO 1000*PAUSEFACTOR%:NEXT PAUSE:CLS:RESUME 12100
12090 COLOR RED:PRINT "ERROR";ERR;"AT LINE";ERL : GOTO 15430
12100 ON ERROR GOTO 15200:KEY ON:GOTO 4540
12110 REM -------------------- FIND PATH BLOCKAGE --------------------
12120 IF Y=0 THEN 12360
12130 IF X=0 THEN 12440
12140 M=Y/X
12150 IF ABS(M)>1 THEN 12260
12160 IF X<0 THEN M1=-1 :ELSE M1=1
12170 IF X=0 THEN M1=0
12180 B=S2-M*S1
12190 FOR X5=S1+M1 TO S1+X STEP M1
12200     IF X5>10 OR X5<1 THEN 12520
12210     Y5=INT(M*X5+B+.5)
```

```
12220     IF Y5>10 OR Y5<1 THEN 12520
12230     IF Q(X5,Y5)>1 THEN 12540
12240 NEXT X5
12250 GOTO 12520
12260 M=1!/M : B=S1-M*S2
12270 IF Y<0 THEN M1=-1 :ELSE M1=1
12280 IF Y=0 THEN M1=0
12290 FOR Y5=S2+M1 TO S2+Y STEP M1
12300     IF Y5>10 OR Y5<0 THEN 12520
12310     X5=INT(M*Y5+B+.5)
12320     IF X5>10 OR X5<1 THEN 12520
12330     IF Q(X5,Y5)>1 THEN 12540
12340 NEXT Y5
12350 GOTO 12520
12360 Y5=S2
12370 IF X<0 THEN M1=-1 :ELSE M1=1
12380 IF X=0 THEN M1=0
12390 FOR X5=S1+M1 TO S1+X STEP M1
12400     IF X5>10 OR X5<1 THEN 12520
12410     IF Q(X5,Y5)>1 THEN 12540
12420 NEXT X5
12430 GOTO 12520
12440 X5=S1
12450 IF Y<0 THEN M1=-1 :ELSE M1=1
12460 IF Y=0 THEN M1=0
12470 FOR Y5=S2+M1 TO S2+Y STEP M1
12480     IF Y5>10 OR Y5<1 THEN 12520
12490     IF Q(X5,Y5)>1 THEN 12540
12500 NEXT Y5
12510 REM -------------------- ON BLOCKAGE --------------------
12520 ON V4 GOTO 5660,7640
12530 REM -------------------- PATH BLOCKED --------------------
12540 ON V4 GOTO 5620,7640
12550 REM -------------------- COMPUTER --------------------
12560 IF D(7)>=0 THEN 12590
12570 COLOR RED:PRINT "COMPUTER IS DAMA%$&()$.....ED":COLOR CYAN
12580 GOTO 4520 : REM END OF GAME DETECTOR
12590 COLOR WHITE:PRINT "COMPUTER ON"
12600 KEY 1,"OFF"+CHR$(13)
12610 KEY 2,"MAP"+CHR$(13)
12620 KEY 3,"KLINGO"+CHR$(13)
12630 KEY 4,"BASES"+CHR$(13)
12640 KEY 5,"TIME"+CHR$(13)
12650 KEY 6,"PHASER"+CHR$(13)
12660 KEY 7,"COURSE"+CHR$(13)
12670 KEY 8,"SAVE"+CHR$(13)
12680 KEY 9,"RETURN"+CHR$(13)
12690 KEY 10,"RATING"+CHR$(13)
12700 A$="":INPUT "READY . . . ",A$
12710 T=T+.05 : A$=A$+"###" : A$=LEFT$(A$,3) : GOSUB 15650
12720 FOR I=11 TO 20
12730     IF A$=COMMAND$(I) THEN 12780
12740 NEXT I
12750 COLOR RED:PRINT "ILLEGAL REQUEST ";RANK$:COLOR WHITE:PRINT
12760 K5=K5-1
12770 GOTO 12700
12780 ON I-10 GOTO 13230,12930,13050,13420,13550,14040,14300,14340,13380,12800
12790 REM -------------------- PHASERS --------------------
12800 IF K3>0 THEN 12840
12810 PRINT "NO KLINGONS IN THIS QUADRANT"
12820 K5=K5+1
12830 PRINT:GOTO 12700
12840 PRINT "KLINGON DESTRUCT ENERGIES"
12850 PRINT "LOCATION","MAX","MIN"
12860 FOR I=1 TO K0
12870     IF K(I,3)<=0 THEN 12900
12880     W=FND(S1,S2,K(I,1),K(I,2))
12890     PRINT " ";STR$(K(I,1));-K(I,2),INT(W*K(I,3)/2+.5),INT(W*K(I,3)/3+.5)
12900 NEXT I
12910 PRINT:GOTO 12700
12920 REM -------------------- KLINGONS --------------------
12930 IF K3>0 THEN 12970
```

```
12940 PRINT "NO KLINGONS IN THIS QUADRANT"
12950 K5=K5+1
12960 PRINT:GOTO 12700
12970 PRINT "LOCATION","DISTANCE","ENERGY"
12980 FOR I=1 TO K0
12990     IF K(I,3)<=0 THEN 13020
13000     W=INT(100*FND(S1,S2,K(I,1),K(I,2)))/100
13010     PRINT " ";STR$(K(I,1));-K(I,2)," ";W,K(I,3)
13020 NEXT I
13030 PRINT:GOTO 12700
13040 REM -------------------- BASES --------------------
13050 IF B4>0 THEN 13110
13060 IF T9-T<=0 THEN 13090
13070 PRINT "EXACT LOCATION OF BASES NOT KNOWN"
13080 PRINT:GOTO 12700
13090 COLOR RED:PRINT "BASES ABANDONED!!":COLOR WHITE
13100 PRINT:GOTO 12700
13110 PRINT "KNOWN BASE LOCATIONS"
13120 PRINT "QUADRANT";TAB(13);"SECTOR";TAB(24);"COURSE";TAB(35);
13130 PRINT "DISTANCE";TAB(48);"PHOTON & ENERGY"
13140 FOR I=1 TO B4
13150     X=(10*B(I,1)+B(I,3)-1)-(10*Q1+S1-1)
13160     Y=(10*B(I,2)+10-B(I,4))-(10*Q2+10-S2)
13170     W=INT(SQR(X^2+Y^2))/10 : X=INT(X)/10 : Y=INT(Y)/10
13180     PRINT " ";STR$(B(I,1));-B(I,2);TAB(13);STR$(B(I,3));-B(I,4);TAB(23);
13190     PRINT STR$(X);",";-Y;TAB(36);W;TAB(49);B(I,6);"     ";B(I,5)
13200 NEXT I
13210 PRINT:GOTO 12700
13220 REM -------------------- MAP --------------------
13230 PRINT : PRINT "     1   2   3   4   5   6   7   8   9   10  11"
13240 FOR I=0 TO 10
13250     IF I<9 THEN PRINT 1+I;" ";:ELSE PRINT 1+I;
13260     FOR J=11 TO 1 STEP -1
13270         IF QUADRANT$(I*11+J)<>"   " THEN 13290
13280     NEXT J
13290     IF I=Q2-1 AND J<Q1 THEN J=Q1
13300     FOR J=1 TO J
13310         PRINT QUADRANT$(I*11+J);
13320         IF I<>Q2-1 OR J<>Q1 THEN PRINT " "; :ELSE PRINT "*";
13330     NEXT J
13340     PRINT
13350 NEXT I
13360 PRINT:GOTO 12700
13370 REM -------------------- RATING --------------------
13380 PRINT "YOUR RATING AS OF NOW IS ";
13390 PRINT INT((K1-K9)/(T-T0)*1000)-K5;RANK$
13400 PRINT:GOTO 12700
13410 REM -------------------- OFF --------------------
13420 PRINT "COMPUTER OFF":COLOR CYAN
13430 KEY 1,"HELP"+CHR$(13)
13440 KEY 2,"WARP"+CHR$(13)
13450 KEY 3,"PHASER"+CHR$(13)
13460 KEY 4,"PHOTON"+CHR$(13)
13470 KEY 5,"SRS"+CHR$(13)
13480 KEY 6,"LRS"+CHR$(13)
13490 KEY 7,"DAMAGE"+CHR$(13)
13500 KEY 8,"STATUS"+CHR$(13)
13510 KEY 9,"SHIELD"+CHR$(13)
13520 KEY 10,"COMPT"+CHR$(13)
13530 GOTO 4520 : REM END OF GAME DETECTOR
13540 REM -------------------- TIME --------------------
13550 IF T9-T>0 THEN 13580
13560 PRINT "BASE STATIONS ABANDONED AS OF:";T9;", TIME IS NOW:";T
13570 PRINT:GOTO 12700
13580 PRINT "STARDATES LEFT UNTIL EVACTUATION OF BASES:";T9-T
13590 PRINT:GOTO 12700
13600 REM -------------------- DAMAGE SECTION --------------------
13610 Z1=Z2-Z1
13620 IF Z1<0 THEN 13690
13630 PRINT:PRINT "SHIELDS HELD ";RANK$;", POWER REMAINING IN THEM:";Z1
13640 Z2=Z1
13650 IF D(5)>=0 THEN E(2)=1
```

```
13660 FOR PAUSE=1 TO 1000*PAUSEFACTOR%:NEXT PAUSE
13670 PRINT
13680 RETURN
13690 IF D(1)>=0 THEN 13760
13700 COLOR RED
13710 PRINT "THE SHIP ABSORBED ";-Z1;" UNITS OF ENERGY."
13720 FOR PAUSE=1 TO 1000*PAUSEFACTOR%:NEXT PAUSE
13730 PRINT
13740 COLOR CYAN
13750 GOTO 13790
13760 COLOR RED
13770 PRINT "SHIELDS HAVE BUCKLED ";RANK$;", OVERLOADED BY";STR$(-Z1);"."
13780 COLOR CYAN
13790 Z2=0
13800 E=E+Z1
13810 D(1)=-(INT(RND*5+1))
13820 FOR I=1 TO 7
13830     W(I)=I+1
13840 NEXT I
13850 FOR I=1 TO 10
13860     J6=INT(RND*11+1) : J7=INT(RND*11+1) : J8=W(J6) : W(J6)=W(J7) : W(J7)=J8
13870 NEXT I
13880 J6=INT(-Z1/85)+1
13890 IF J6>7 THEN J6=7
13900 FOR I=1 TO J6
13910     D(W(I))=-INT(RND*3+1)
13920 NEXT I
13930 IF D(8)<0 AND E>200 THEN E=200
13940 COLOR RED
13950 PRINT "SICKBAY REPORTS CASUALTIES OF";
13960 J7=INT(RND*5*J6) : C(1)=C(1)+J7:K5=K5+25*J7
13970 PRINT J7;"KILLED, AND"; : J7=INT(RND*10*J6) : C(2)=C(2)+J7
13980 PRINT J7;"INJURED." : K5=K5+J7:COLOR CYAN
13990 IF D(5)>=0 THEN E(2)=1
14000 FOR PAUSE=1 TO 1000*PAUSEFACTOR%:NEXT PAUSE
14010 PRINT
14020 RETURN
14030 REM -------------------- COURSE --------------------
14040 LINE INPUT "DESTINATIONS (QUA,SEC/..8-5,3-10/....)";P$
14050 P(0)=LEN(P$) : D8=7 : J4=0
14060 PRINT "QUADRANT";TAB(15);"SECTOR";TAB(30);"COURSE";TAB(45);"DISTANCE"
14070 X(0)=0
14080 L(0)=0 : T%=0
14090 J4=J4+1
14100 IF J4>P(0) THEN 14220
14110 V$=MID$(P$,J4,1)
14120 IF " "=V$ THEN 14090
14130 IF V$<"0" OR V$>"9" THEN 14160
14140 L(0)=L(0)+1 : L(L(0))=VAL(V$)
14150 GOTO 14090
14160 IF V$<>"," AND V$<>"/" AND V$<>"-" THEN 14210
14170 FOR I%=1 TO L(0) : T%=T%+L(I%)*10^(L(0)-I%) :NEXT I%
14180 X(0)=X(0)+1 : X(X(0))=T%
14190 IF X(0)=4 THEN 14230
14200 GOTO 14080
14210 COLOR RED:PRINT "DESTINATION ERROR":COLOR WHITE
14220 PRINT:GOTO 12700
14230 X=(10*X(1)+X(3)-1)-(10*Q1+S1-1) : Y=(10*X(2)+10-X(4))-(10*Q2+10-S2)
14240 W=SQR(X^2+Y^2)/10 : X=INT(X)/10 : Y=INT(Y)/10
14250 PRINT " ";STR$(X(1));-X(2);TAB(15);STR$(X(3));-X(4);TAB(29);STR$(X);",";
14260 PRINT -Y;TAB(45);W
14270 IF J4<P(0) THEN 14070
14280 PRINT:GOTO 12700
14290 REM -------------------- SAVE --------------------
14300 U(1)=Q1 : U(2)=Q2 : U(3)=S1 : U(4)=S2
14310 PRINT "PRESENT LOCATION (";STR$(Q1);-Q2;",";STR$(S1);-S2;") SAVED."
14320 PRINT:GOTO 12700
14330 REM -------------------- RETURN --------------------
14340 IF U(1)=0 GOTO 14470
14350 COLOR MAGENTA:PRINT "COMPUTER ASSUMING CONTROL OF HELM . . .":COLOR WHITE
14360 N$="IT'S RUNNING OUT OF ENERGY BEFORE REACHING BASE"
14370 FOR PAUSE = 1 TO 1000*PAUSEFACTOR%:NEXT
```

```
14380 W=8 : U(5)=1
14390 X=INT((10*U(1)+U(3)-1)-(10*Q1+S1-1)+.001)
14400 Y=-INT((10*U(2)+10-U(4))-(10*Q2+10-S2)+.001)
14410 COLOR CYAN
14420 GOTO 5660
14430 GOSUB 14510
14440 U(5)=0
14450 COLOR WHITE
14460 PRINT:GOTO 12700
14470 COLOR RED:PRINT "NO STORED LOCATION EXISTS":COLOR WHITE
14480 K5=K5+1
14490 PRINT:GOTO 12700
14500 REM -------------------- STATUS REPORT --------------------
14510 IF Q(S1,S2)<>4 THEN 14600
14520 CONDITION$="DOCKED"
14530 CONDCOLOR=WHITE:ENERCOLOR=WHITE:SHICOLOR=WHITE:TORPCOLOR=WHITE
14540 FOR I=1 TO B4
14550     IF Q1=B(I,1) AND Q2=B(I,2) THEN 14580
14560 NEXT I
14570 COLOR RED:PRINT "ERROR ON BASE GENERATION - NOTIFY CHIEF ENGINEER FAINI":COLOR CYAN
14580 B8=I
14590 GOTO 14690
14600 CONDITION$="GREEN"
14610 CONDCOLOR=GREEN:ENERCOLOR=GREEN:SHICOLOR=GREEN:TORPCOLOR=GREEN
14620 IF Z2<1000 THEN SHICOLOR=YELLOW
14630 IF Z2<500 THEN SHICOLOR=RED
14640 IF P=0 THEN TORPCOLOR=RED
14650 IF E<=500 THEN ENERCOLOR=YELLOW
14660 IF FNA<>0 THEN CONDITION$="YELLOW":CONDCOLOR=YELLOW:SHICOLOR=RED
14670 IF E<=200 THEN CONDITION$="EMERG PWR":CONDCOLOR=YELLOW:ENERCOLOR=RED
14680 IF K3>0 THEN CONDITION$="RED":CONDCOLOR=RED
14690 IF CONDITION$<>"DOCKED" THEN 14890
14700 IF P>=15 THEN K=0 :ELSE K=15-P
14710 IF B(B8,5)<>0 THEN 14750 :ELSE COLOR RED:PRINT "NO ENERGY AVAILABLE"
14720 COLOR CYAN
14730 GOTO 14810
14740 IF E>4000 THEN 14770
14750 IF B(B8,5)<4000-E THEN I=B(B8,5) :ELSE I=4000-E
14760 B(B8,5)=B(B8,5)-I : E=E+I
14770 IF B(B8,5)<1000-Z2 THEN I=B(B8,5) :ELSE I=1000-Z2
14780 B(B8,5)=B(B8,5)-I : Z2=Z2+I
14790 IF B(B8,5)<>0 THEN 14810
14800 COLOR RED:PRINT "ALL BASE STATION ENERGY NOW CONSUMED":COLOR CYAN
14810 IF B(B8,6)<>0 THEN 14840
14820 COLOR RED:PRINT "NO TORPEDOES AVAILABLE":COLOR CYAN
14830 GOTO 14890
14840 IF K<B(B8,6) THEN P=P+K :ELSE P=P+B(B8,6)
14850 IF K<B(B8,6) THEN V9=K :ELSE V9=B(B8,6)
14860 B(B8,6)=B(B8,6)-V9
14870 IF B(B8,6)<>0 THEN 14890
14880 COLOR RED:PRINT "ALL BASE STATION TORPEDOES NOW CONSUMED":COLOR CYAN
14890 PRINT "STARDATE:";:COLOR WHITE:PRINT T;:COLOR CYAN
14900 PRINT TAB(28)"CONDITION: ";
14910 COLOR CONDCOLOR:PRINT CONDITION$;:COLOR CYAN
14920 PRINT TAB(50)"KLINGONS:";:COLOR RED:PRINT K9:COLOR CYAN
14930 PRINT "QUADRANT:";:COLOR WHITE:PRINT STR$(Q1);-Q2;:COLOR CYAN
14940 PRINT TAB(18)"SECTOR:";:COLOR WHITE:PRINT STR$(S1);-S2;:COLOR CYAN
14950 PRINT TAB(32)"ENERGY:";:COLOR ENERCOLOR:PRINT E;:COLOR CYAN
14960 PRINT TAB(47)"SHIELDS:";:COLOR SHICOLOR:PRINT Z2;:COLOR CYAN
14970 PRINT TAB(62)"TORPEDOES:";:COLOR TORPCOLOR:PRINT P:COLOR CYAN
14980 IF FNB(Q1,Q2)=0 OR CONDITION$="DOCKED" THEN 15000
14990 PRINT "BASE STATION IN JUXTAPOSITION AT";STR$(B1);-B2
15000 IF Q(S1,S2)<>4 THEN 15110
15010 J6=0
15020 FOR I=1 TO 8
15030     IF D(I)<J6 THEN J6=D(I)
15040 NEXT I
15050 FOR I%=1 TO 8 : D(I%)=0 : NEXT I%
15060 IF J6=0 THEN 15110
15070 T=T-J6
15080 COLOR GREEN
15090 PRINT"REPAIRS COMPLETED ";RANK$;". ";-J6;"STARDATES HAVE ELAPSED."
```

```
15100 COLOR CYAN
15110 RETURN
15120 REM -------------------- END OF GAME --------------------
15130 IF (E+Z2)<=0 THEN 11170
15140 REM -------------------- DETECTOR --------------------
15150 IF K9<=0 THEN 11260 : REM MISSION COMPLETE
15160 PRINT
15170 RETURN
15180 COLOR RED:PRINT "INCORRECT DATA":COLOR CYAN
15190 K5=K5+.5
15200 ON D8 GOTO 5290,6570,6980,2450,15420,8360,12700
15210 RESUME 15220
15220 GOTO 15420 : REM END GAME
15230 ON ERROR GOTO 15410
15240 OPEN "I",#1,"STARTREK.DAT"
15250 OPEN "O",#2,"STARTREK.TMP"
15260 IF EOF(1) THEN 15300
15270 LINE INPUT#1,A$
15280 PRINT#2,A$
15290 GOTO 15260
15300 CLOSE #1
15310 KILL "STARTREK.DAT"
15320 STARDATE$=RIGHT$(DATE$,2)+LEFT$(DATE$,2)+"."+MID$(DATE$,4,2)
15330 DATAFILE$="\        \ \                  \  \           \"
15340 DATAFILE$=DATAFILE$+"  \           \ ####  #####"
15350 PRINT#2,USING DATAFILE$;STARDATE$,PLAYER$,RANK$,SHIP$,K1,R
15360 CLOSE
15370 NAME "STARTREK.TMP" AS "STARTREK.DAT"
15380 REM
15390 RETURN
15400 REM ERROR 53 FILE NOT FOUND
15410 IF ERR=53 AND ERL=15240 THEN OPEN "O",#2,"STARTREK.TMP":RESUME 15320
15420 ON ERROR GOTO 0 : REM ERROR TRAPPING TURNED OFF
15430 KEY 1,"LIST "
15440 KEY 2,"RUN"+CHR$(13)
15450 KEY 3,"LOAD"+CHR$(34)
15460 KEY 4,"SAVE"+CHR$(34)
15470 KEY 5,"CONT"+CHR$(13)
15480 IF Z100$<>"TRUE" THEN 15580
15490 KEY 6,"PRINT "
15500 KEY 7,"AUTO "
15510 KEY 8,"FOR "
15520 KEY 9,"NEXT "
15530 KEY 10,"GOSUB "
15540 KEY 11,""
15550 KEY 12,""
15560 COLOR WHITE
15570 GOTO 15640
15580 KEY 6,","+CHR$(34)+"LPT1:"
15590 KEY 7,"TRON"+CHR$(13)
15600 KEY 8,"TROFF"+CHR$(13)
15610 KEY 9,"KEY "
15620 KEY 10,"SCREEN 0,0,0"+CHR$(13)
15630 COLOR WHITE
15640 END
15650 REM ---------- CONVERT LOWERCASE TO UPPERCASE ----------
15660 FOR I=1 TO LEN(A$)
15670      C=ASC(MID$(A$,I,1))
15680      IF C>=97 AND C<=122 THEN MID$(A$,I,1)=CHR$(C-32)
15690 NEXT I
15700 RETURN
```