

## QBITS Galaxy Introduction

The nineteen eighties and I had perhaps an overly enthusiastic interest in Science Fiction and high on my to do list was write a Space Adventure Prog. However, in June 1987 Sinclair QL World Magazine published **Stellaris** by David Cormona. For what I was trying to achieve at the time this delayed any further development. Reviewing my early aspiration, I was struck by the thought as to whether there was any connection between the nineteen-eighties QL Game and the release of **Stellaris Galaxy** by Paradox Development Studio in 2016.

## QBITS Galaxy Concepts

In designing QBITS Galaxy, the challenge was getting the right balance between ease of play while maintaining an acceptable level of difficulty. To keep the Player engaged with presented opportunities and maintain the satisfaction of completing a series of events.

In review of previous releases, the basic screen layout worked well enough, but there were a number of modifications and additions I wished to make. The opposing sides are now established as Alliance and Republic. The Status Panels are more Graphically presented and now with a Fuel bar. For Star Ship to Planet and visa verse Transfers, there are more options. The Encounter section has improved Phaser & Shield Graphics as well as Space Craft Images.

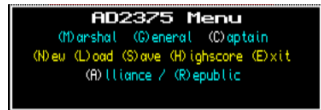
## QBITS Galaxy Game

After choosing which side to play the aim is to take the Galaxy Star System by Star System and reduce your Opponent's ownership to zero. Those Stars encircled on the Galaxy Grid are either in Player's or Opponent colours, other Star Systems are not highlighted and remain Independent until acquired. Selecting a Star System activates a Hyperspace jump to Orbit the primary Planet.

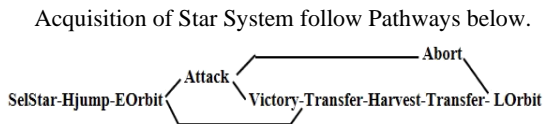
If an Independent or an Opponent's Star System you commence with an attack. If successful or Star System already acquired, then Transfers take place with Planet to advance the Population and develop its Technology. The results are Harvested to replenish and/or increase a Star Ships Tactical Stock especially WMD for future missions. There are hazards to encounter that may disrupt or even reverse your progress.

## QBITS AD2375 - New Game

Selection of Side (Alliance or Republic) and Skill level (Marshal or General or Captain) are made from the Main Menu. The side played by the current Player is held by variable **Gp**.



Side chosen by Gamer is held by variable **Gp1** and Computer Opponent held by **Gp2**.



Star Ships Tactical Information displayed with Galaxy Grid reference on the left side of the screen with Star System and Planet details on the right.



Attack Mode uses a Random Throw of the Dice as shown in the Circles placed below Star Ship and Planet Information.



The Outcome, a **Victory** and acquisition of the Star System or an **Abort** due to insufficient WMD's to continue the attack. Victory secures an increase in Star Ships Tactical Stock adding to or replacing lost items. The next move is to **Transfer** from Star Ship to Planet.

## QBITS AD2375 - Harvest

By transferring Star Ship Food stocks, refined Ores, Technology and opening Trade facilitates increases Population and Production. Activating **(H)**arvest generates results which can be transferred from Planet to Star Ship to replenish and build up Tactical Stocks WMDs.etc

## QBITS AD2375 - Simulation

Computer generated moves are based on Tactical assessment. Engagement requires sufficient WMD's to defeat a defending Star System. As Planet acquisitions and Harvests increase Tactical Stocks can grow. For the opening stages random selection of any Star system gives an even chance of success. Once the number of Star Ships and WMD have been increase then selection of opponent Star Systems are targeted.

However, in the event of heavy losses of Star Ships and WMD's, only home Star Systems are sought so as to restock for future missions. A simple REPEAT Loop makes random selection based on values of a Sides WMD Stocks. See Page 97 **Opponent & Simulation Mode**, this generates inputs to the screen displays showing changes to the Star Ship and Star System Information Tables as the Game progresses.

## QBITS AD2375 - Hazards

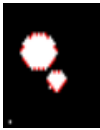
View a passing



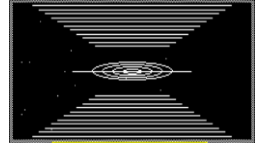
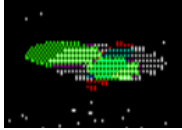
Pulsar



Twin Star



or Nebula these just create Time delays.



A **Black Hole** if not avoided costs a Star Ship + 20WMDs.

**Time Slip** happens rarely, here Tactical Status of each side is Swapped. You may have been winning or losing now the Tables are turned.

**Encounter** is attacks by the Opponents Fighters. Each time this happens you lose a Fuel Cell to charge Shields and Phasers.

However, if you are successful in destroying an Enemy Fighter your Opponent loses the Fighter Plus 3 WMD and 1 Fuel Cell.



## QBITS AD2375 - Control Keys

Prompts are displayed in the Message window relative to the actions taking place. Extensive use is made of the Cursor keys and Spacebar. For Selection of Main Menu items and some other actions use Letter Keys such as **N/n** for (N)ew, **L/l** for (L)oad, **H/h** for (H)ighScore etc.



## QBITS AD2375SE - End Game

League Table					
Skill	Time h:m:s	WMD	Stars	Gamer	
Marshal	01:24:32	1024	58	SPlay	
Admiral	01:02:45	1185	62	CPPlay	
Captain	00:57:32	1296	66	QBITS	



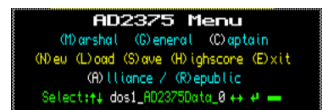
## QBITS AD2375 - League Table

Default is set with first use. To Reset Table, press Key 'd' from Menu Panel, enter High Score at End of Game. Pressing key 'D' (UPPERCASE) will reset Default.

## QBITS AD2375 - File Management

The set up for loading and running the AD2375 Prog is one of a group of QBITS Progs selected from a QBITSProgs Menu. On start up the AD2375 Prog therefore expects to set a number of variables with information held by a QBITConfig file. These are the gx,gy offsets to run under a QPCII environment, the default return to QBITSProgs Menu and a number of other variables holding Device names.

(L)oad and (S)ave are able to offer Data storage to a range of Devices & Set Files 'AD2375Data\_0' to '9'.

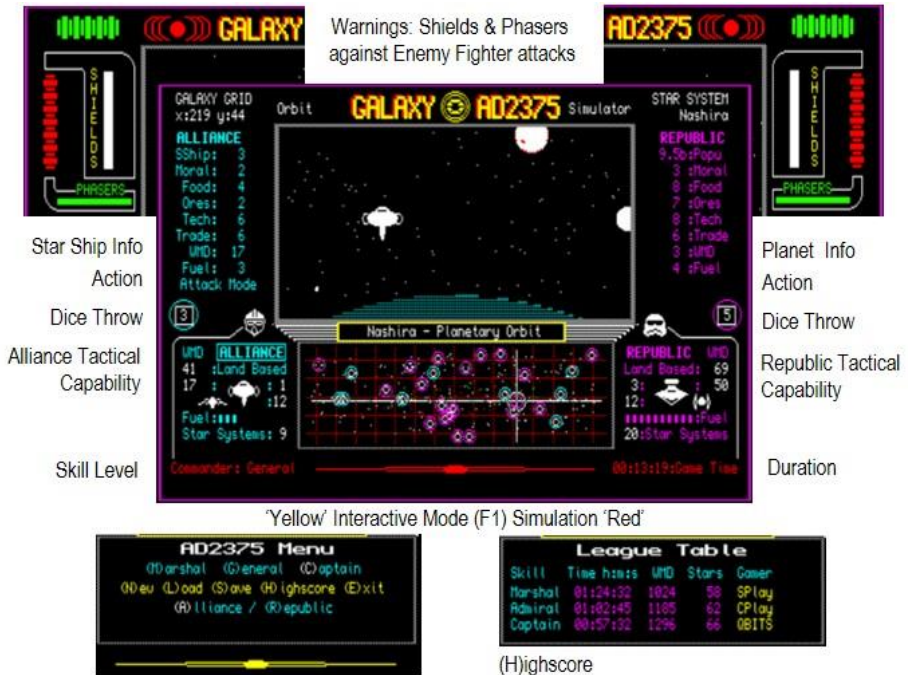


The File Management carries out basic checks to validate a Device and Filename returning '**File NOT Found**' or '**DEVICE ERROR**' when Saving and will prompt for '**Overwrite Y/N**' if File is found to already exist. If no problems are found the Prog will LOAD from or SAVE to Selected Device & Filename.

## QBITS GALAXY Procedures

- GMenu** - Start Menu - Side : Player Level : New/ Load/Save/Exit
- GChoice** - Highlights selected Side & Player Level.
- HigScore** - League Table
- QExit** - Exit Game
- Plus** - **Init\_New, GLoad, GSave, FCheck, SelPath**
- QBold** - Bold Lettering,
  
- QAD2375** - Main Menu
- GTime** - Game Timer
- Write** - Prints Messages as prompts of actions to be taken
- GMapInit** - Draws the Galaxy Grid and Stars distributed across it
- GSide** - Displays - Gamer - Opponent or Simulation Mode
  
- Hazards** - Blackhole can trap a Starship : Timeslip Swaps Players Stats  
**Pulsar, Binary, Nebula, BlackHole-Bhole, TimeSlip**
- Encounter** - Enemy Fighters Attack
  - EAS** Early Warning Alarm System
  - DPS** Draws Shields & Phasers Graphics
  - RChange** Charges up Shields & Phasers
  - Defender Phaser **PSights, PFire, Phaser**
  - Enemy Phaser Fire **FFire**
  - Enemy Fighters **AF1, AF2, AF3, RF1, RF2, RF3**
  
- LMap** - Draws Star System of Alliance & Republic
- PMap** - Draws Crosswires of selected Star system
  
- HJump** - Hyper Space Jump to Selected Start system **StarMove, StarView, Sfuel**
- EOrbit** - Drop from Warp to Impulse and Orbit above Prime Planet of selected Star System
- LOrbit** - Warp out of Orbit into deep space
  
- GPlay** - Main Game loop - Info / Moral of StarShips and Planet of selected Star System.  
**SSInfo SSNoral, PTInfo, PTMoral, SStats, WMD**
  - Attack actions **RollDice - DiceA, DiceB, DiceCLS, GPnt, Victory,**
  - Transfers **TranSP, Harvest, TranPS**
  - Game End **Gend, DeathStar, DSAttack, DSDefeat**
  - League Table **LScore, LTDefault, LTRreset, Lname, LSave, LLoad**
  
- GStarSel** - Computer Simulation Mode - Opponents Moves
  - Attack actions **GSAtt**
  - Transfers actions **GSTrans**
  - Clears for next player **GSwap**
  
- Note:** - **F1** runs Simulation Mode for both Player [Gamer] & Opponent
  
- Init\_Win, GTitle, Emblem, GIcon, Init\_Stats, Alliance, Republic, StarFleeth**
  
- GIntro, GNotes,** - Background and Basic Instructions
  
- GameTest** - **F2** runs Tests the Various Screen Graphics Displays

## QBITS AD2375SE Code



1000 REMark **QBITS\_AD2375SE\_bas** [QBITS GALAXY AD2375 SE Review 2023 - QPC2]

1002 dev\$='win1\_':MODE 4:gx=0:gy=0 :REMark Basic Screen Settings

1004 **WHEN ERrOr** :eck=1:**CONTINUE:END WHEN**

1006 REMark **Import QBITSConfig Settings - QPC2**

1007 OPEN \_IN#9,dev\$&'QBITSConfig':INPUT#9,gx\gy\dn\$'\dev\$\dn%\dm%

1008 DIM Drv\$(15,5):FOR d=0 TO 15:INPUT#9,Drv\$(d):END FOR d:CLOSE#9

1010 REMark **Setup Arrays**

1011 DIM astro\$(70,10),astro(70,12)

:REMark Star/Planet 1-70 Info

1012 DIM side(2,6),min(6),max(6)

:REMark Players Info : GTS h:m:s

1013 DIM Gclk\$(20),name\$(3,10),score(3,3)

:REMark GTS h:m:s League Table

1015 REMark **Setup Variables**

1016 wck=0:nck=1:pck=0:eck=0

:REMark WMD/New/Load/Save/ERRor checks

1017 Gp=0:Gp1=1:Gp2=2:Gs=0:Gsk=3

:REMark Player/Side/Skill

1018 AWMD=0:ASSn=1:Alnt=12:APs=0

:REMark Alliance WMD/SShips/Fighters/Planets

1019 RWMD=0:RSSn=1:RInt=12:RPs=0

:REMark Republic WMD/SShips/Fighters/Planets

1020 dt=0:dw=3:ack=1:vck=0:t=1

:REMark Dicethrow:Del WMD:Attack/Transfer

1021 df%=0:ws=1:Mes\$=''

:REMark device file:Warp Speed:Messages

1022 GTS=0:TD1=20:TD2=60:LTck=0

:REMark Game Time:Delays:League Table check

1024 Init\_Win:GMenu



1026 REMark Part 1 SetUp - Intro - Menu

1028 DEFine PROCEDURE GMenu

1029 BLOCK#2,62,10,94,6,0:CLS#13:CLS#16

1030 INK#20,2:CURSOR#20,390,230:PRINT#20,' :Game Time'

1031 CSIZE#13,2,0:INK#13,7:QBold 13,12,56,2,'AD2375 Menu':CSIZE#13,0,0

1032 INK#13,6:CURSOR#13,16,26:PRINT#13,'(N)ew (L)oad (S)ave (H)ighscore (E)xit'

1033 ch=13:GChoice:SPck=0:GIcon:Alliance:Republic

1034 REPEAT lp

1035 k=CODE(INKEY\$(-1)):ch=13

1036 SELECT ON k

1037 =77,109:TD2=20 :GChoice :REMark Marshal

1038 =71,103:TD2=40 :GChoice :REMark General

1039 =67,99:TD2=60 :GChoice :REMark Captain

1040 =65,97:Gp1=1:Gp2=2 :GChoice:Init\_Stats :REMark Alliance

1041 =82,114:Gp1=2:Gp2=1 :GChoice:Init\_Stats :REMark Republic

1042 =78,110:mch=1:nck=0 :Init\_New:GAD2375 :REMark (N)ew

1043 =76,108: GLoad :REMark (L)oad

1044 =83,115: IF nck=0 :GSave :REMark (S)ave

1045 =72,104:LScore:PAUSE :GMenu :REMark (H)ighscore

1046 =68 :LTDefault:LSave :LScore:PAUSE:GMenu :REMark (D)efault Highscore

1047 =100:LTRreset :LSave :LScore:PAUSE:GMenu :REMark (d) Reset highscore

1048 =69,101 :QExit :REMark (E)xit

1049 =32 : IF nck=0 :GMapInit:GAD2375 :REMark Continue Game

1050 END SELECT

1051 END REPEAT lp

1052 END DEFine

1054 DEFine PROCEDURE QBold(ch,cs,cx,cy,str\$)

1055 OVER#ch,1:sl=LEN(str\$)

1056 FOR c=1 TO sl

1057 FOR b=0 TO 1:CURSOR#ch,cx+b+c\*cs,cy:PRINT#ch,str\$(c)

1058 END FOR c:OVER#ch,0

1059 END DEFine

1061 DEFine PROCEDURE GChoice

1062 IF TD2=20:Tm\$='Marshal':GSk=1:xsk=38:sk\$='(M)':mx=4:my=2

1063 IF TD2=40:Tm\$='General':GSk=2:xsk=104:sk\$='(G)':mx=6:my=3

1064 IF TD2=60:Tm\$='Captain':GSk=3:xsk=170:sk\$='(C)':mx=8:my=4

1065 IF Gp1=1:xsd=62:sd\$='(A)':ELSE Gp1=2:xsd=140:sd\$='(R)'

1066 INK#20,2:CURSOR#20,8,230:PRINT#20,'Commander: ',Tm\$

1067 INK#ch,5:CURSOR#ch,62,38:PRINT#ch,'(A)lliance / (R)epublic'

1068 CURSOR#ch,38,14:PRINT#ch,'(M)arshai (G)eneral (C)aptain':INK#ch,7

1069 CURSOR#ch,xsk,14:PRINT#ch,sk\$:CURSOR#ch,xsd,38:PRINT#ch,sd\$

1070 IF nck=0:Write 'To Continue Press Spacebar'

1071 END DEFine

1073 DEFine PROCEDURE QExit

1074 Write 'Exit Y/N':PAUSE:IF KEYROW(5)=64:LRUN dn\$:STOP:ELSE Write "

1075 END DEFine

## 1077 REMark Part 2 Game AD2370

### 1079 DEFine PROCEDURE GAD2375

```

1080 ch=1:CLS#ch:INK#ch,7:FOR i=1 TO 120:POINT#ch,RND(10 TO 200),RND( 5 TO 95)
1081 Gclk=DATE:clk1=DATE+TD1:clk2=DATE+TD2:clk3=DATE:haz=0:haz2=0
1082 SStats:LMap:n=RND(20 TO 50):bx=-9:by=-9:PMap
1083 REPEAT lp
1084 GTime:StarMove:BEEP:k=CODE(INKEY$(10))
1085 Write 'Use ← → to Select Star System':BLOCK#16,16,3,34,4,7
1086 IF DATE>clk1:Hazards :clk1=DATE+TD1
1087 IF DATE>clk2:Gp=Gp2 :GSide:clk2=DATE+TD2 :REMark Opponent
1088 IF DATE>clk3 AND SPck=1 :Gp=Gp1:GSide:clk3=DATE+TD2 :REMark Simulator
1089 SElect ON k
1090 =32 :IF SPck=0 :Gp=Gp1:GSide :REMark [SBar] Select Star
1091 =71,103 :GNotes :REMark (G)ame Notes
1092 =77,109:GTS=(DATE-Gclk+GTS) :GMenu :REMark Skill/side/NEW/Load/Save
1093 =192:IF SPck=0 AND n> 1 :n=n-1:Gp=Gp1:SSInfo :REMark ¼ Star Left
1094 =200:IF SPck=0 AND n<70:n=n+1:Gp=Gp1:SSInfo :REMark ½ Star Right
1095 =232:IF SPck=0:SPck=1 :GIcon :ELSE SPck=0:GIcon :REMark F1 Simulator
1096 =236:sn=n:GameTest :REMark F2 Graphics Test
1097 =240 :Blackhole:TimeSlip :REMark F3 cheat-Test
1098 =244 :Encounter:CLS#16 :REMark F4 Cheat-Test
1099 =248:side(Gp1,5)=side(Gp1,5)+10:SStats :REMark F5 Cheat-Tec/MODS
1100 END SElect
1101 END REPEAT lp
1102 END DEFine

```

### 1104 DEFine PROCEDURE GTime

```

1105 Gclk$=DATE$(DATE-Gclk+GTS):INK#20,2
1106 CURSOR#20,390,230:PRINT#20,Gclk$(13 TO 20)
1107 END DEFine

```

### 1109 DEFine PROCEDURE Write(str\$)

```

1110 PRINT#16,FILL$(' ',16-(LEN(str$) DIV 2))&str$
1111 END DEFine

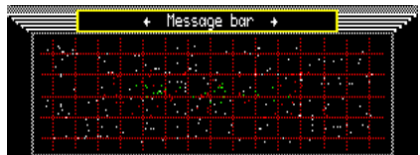
```

### 1113 DEFine PROCEDURE GMapInit

```

1114 ch=13:CLS#ch:INK#ch,0,2,3:SCALE#ch,100,0,0
1115 FOR h=12 TO 96 STEP 18:LINE#ch,6,h TO 310,h
1116 FOR v=16 TO 330 STEP 20:LINE#ch,v,3 TO v,98
1117 INK#ch,2:FOR i=1 TO 60 :POINT#ch,RND(60 TO 260),RND(30 TO 70)
1118 INK#ch,4:FOR i=1 TO 30 :POINT#ch,RND(80 TO 240),RND(40 TO 60)
1119 INK#ch,7:FOR i=1 TO 180:POINT#ch,RND(10 TO 300),RND(10 TO 90)
1120 END DEFine

```



**Note:** Mode of Play Interactive or Simulation is set by Sim Player check 'SPck' and with 'Gp' set to be Player or Opponent side.

### 1122 DEFine PROCEDURE GSide

```

1123 WMD=side(Gp,5):ack=1:t=1:vc=1:Mes$=' ':INK#2,7:CURSOR#2,348,6
1124 IF SPck=1 :PRINT#2,'Simulator' :GStarSel:GSwap:RETurn
1125 IF Gp=Gp2:PRINT#2,' Opponent' :GStarSel:GSwap:RETurn
1126 IF Gp=Gp1:PRINT#2,' Gamer' :HJump:StarView:EOrbit:GPlay:GSwap
1127 END DEFine
:

```

### 1129 DEFINE PROCEDURE Hazards

```

1130 haz=RND(1 TO 6):IF haz=haz2 :RETurn :ELSE haz2=haz
1131 IF haz=1 :BEEP 0, 3, 6,6000,5,0,0,0 :Pulsar :PAUSE 20
1132 IF haz=2 :BEEP 0, 9,18,3000,5,0,0,15:Binary :PAUSE 20
1133 IF haz=3 :Encounter
1134 IF haz=4 :BEEP 0,30,33,6000,5,2,5,15:Nebula :PAUSE 20
1135 IF haz=5 :BEEP 0,12,24,6000,8,6,8,15:Blackhole:PAUSE 20
1136 IF haz=6 AND DATE>clk2
1137 IF Gp1=1 AND APs<RPs+20:RETurn
1138 IF Gp1=2 AND RPs<APs+20:RETurn
1139 BEEP 0,12,32,3000,1,0,11,0:TimeSlip:BEEP 0,1,19,30,8,4,11,0
1140 PAUSE 20:clk2=DATE+TD2
1141 END IF
1142 END DEFINE

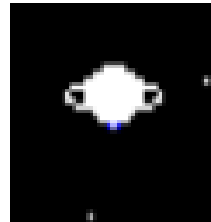
```

### 1144 DEFINE PROCEDURE Pulsar

```

1145 LOCal px,py,pr:pr=.3:ws=1:Write 'Passing Pulsar'
1146 FOR i=1 TO 32
1147 px=90+i*3:py=90-i*2:pr=i/6:GTime:StarMove
1148 INK 241:FILL 1:CIRCLE px,py,pr :FILL 0:PAUSE 3
1149 INK 7:FILL 1:CIRCLE px,py,pr*1.1:FILL 0:PAUSE 2
1150 CIRCLE px,py,pr*2,.3,PI/2:PAUSE 2
1151 INK 0:FILL 1:CIRCLE px,py,pr*2 :FILL 0
1152 END FOR i
1153 END DEFINE

```



### 1155 DEFINE PROCEDURE Binary

```

1156 LOCal a1,a2,bx,by,rx,ry:ws=1:Write 'Passing Binary Star System'
1157 a1=180:a2=0:rx=12:ry=4
1158 FOR i=1 TO 40
1159 bx=72-i*2:by=50+i/2
1160 x1=bx-rx*SIN(RAD(a1)):y1=by-ry*COS(RAD(a1))
1161 x2=bx-rx*SIN(RAD(a2)):y2=by-ry*COS(RAD(a2))
1162 sx=x1:sy=y1:sr=i/12:Star:a1=a1+6:IF a1>360:a1=0
1163 sx=x2:sy=y2:sr=i/6 :Star:a2=a2+6:IF a2>360:a2=0
1164 PAUSE 2+i/4:INK 0:FILL 1:CIRCLE x1,y1,i/4:FILL 0
1165 FILL 1:CIRCLE x2,y2,i/2:FILL 0:GTime:StarMove
1166 END FOR i
1167 END DEFINE

```

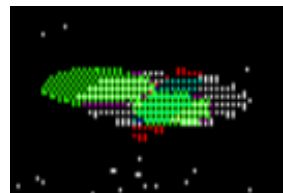


### 1169 DEFINE PROCEDURE Nebula

```

1170 LOCal x1,y1,x2,y2,x3,y3,x4,y4:ws=1:Write 'Passing Nebula'
1171 FOR a=1 TO 40
1172 x1=92-a*3:y1=50-a*1.5:x2=90-a*4:y2=48-a*1.1
1173 x3=94-a*3:y3=48-a*1.5:x4=92-a*3:y4=50-a*1.5:OVER -1
1174 INK 0,7,0:FILL 1:CIRCLE x1,y1,a*1.5,.3,PI/2:FILL 0
1175 INK 0,4,3:FILL 1:CIRCLE x2,y2,a,.3,PI/2:FILL 0
1176 INK 0,4,3:FILL 1:CIRCLE x3,y3,a*.7,.4,PI/2:FILL 0
1177 INK 0,2,0:FILL 1:CIRCLE x4,y4,a*.8,.4,-PI/4:FILL 0:OVER 0
1178 PAUSE 4+a/10:INK 0:FILL 1:CIRCLE x1,y1,a*4,.3,PI/2:FILL 0
1179 GTime:StarMove:INK 7:FOR i=1 TO 5:POINT RND(30 TO 80),RND(5 TO 40)
1180 END FOR a
1181 END DEFINE

```





```

1183 DEFine PROCEDURE Blackhole
1184 x=170:y=92:ws=1:Write "WARNING! >>> Blackhole <<<"
1185 FOR a=8 TO 20:BHole:x=x-a/3:y=y-a/5:GTime:StarMove:PAUSE 5
1186 IF side(Gp1,5)>40
1187   side(Gp1,5)=side(Gp1,5)-20:Write 'One Ship and Crew Lost'
1188 ELSE
1189   Write 'Taking Evasive Action'
1190 END IF
1191 FOR a=20 TO 15 STEP -1:BHole:x=x+a:y=y-a/2:GTime:StarMove:PAUSE 2
1192 StarMove:SStats
1193 END DEFine

```

```

1195 DEFine PROCEDURE BHole
1196 OVER -1:INK 2:CIRCLE x,y,a*.3,.8,PI/2
1197 ARC x,y+a*.2 TO x-a*.3,y,PI/2 TO x,y-a*.3,PI/2 TO x+a*.4,y,PI/2
1198 ARC TO x,y+a*.4,PI/2 TO x-a*.6,y-a*.6,PI/2
1199 ARC x,y+a*.2 TO x-a*.4,y,PI/2 TO x,y-a*.4,PI/2 TO x+a*.5,y,PI/2
1200 ARC TO x,y+a*.5,PI/2 TO x-a*.7,y-a*.7,PI/2
1201 OVER 0:PAUSE 5:INK 0:FILL 1:CIRCLE x,y,a*.4:FILL 0
1202 END DEFine

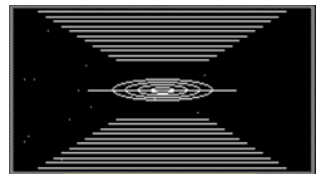
```



```

1204 DEFine PROCEDURE TimeSlip
1205 Write 'Time Slip':BLOCK#2,60,10,96,6,0:INK 7
1206 FOR a=2 TO 8 STEP 2:CIRCLE 96,50,a*.4,.2,PI/2:PAUSE 3
1207 FOR n=1 TO 70
1208   IF astro(n,7)=1:astro(n,7)=2:ELSE IF astro(n,7)=2:astro(n,7)=1
1209 END FOR n
1210 FOR i=1 TO 5
1211   a=side(1,i):side(1,i)=side(2,i):side(2,i)=a
1212 END FOR i
1213 FOR b=16 TO 2 STEP -1
1214   LINE 96-b*3,50 TO 96+b*3,50:LINE 96-b*5,50+b*3 TO 96+b*5,50+b*3
1215   LINE 96-b*5,50-b*3 TO 96+b*5,50-b*3:PAUSE 5
1216 END FOR b
1217 FOR a=1 TO 2:side(a,7)=10:IF side(a,5)<20:side(a,5)=20:END IF
1218 GTime:GMapInit:LMap:bx=-9:by=-9:PMap:SStats
1219 END DEFine

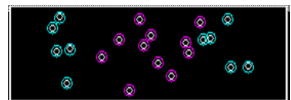
```



```

1221 DEFine PROCEDURE LMap
1222 LOCAL n,nx,ny:ch=13
1223 FOR n=1 TO 70
1224   nx=astro(n,8):ny=astro(n,9):INK#ch,7
1225   IF astro(n,7)=1:CIRCLE#ch,nx,ny,3:INK#ch,5:CIRCLE#ch,nx,ny,6
1226   IF astro(n,7)=2:CIRCLE#ch,nx,ny,3:INK#ch,3:CIRCLE#ch,nx,ny,6
1227 END FOR n
1228 END DEFine

```



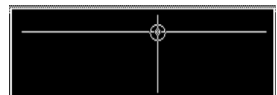
Star List

```

1230 DEFine PROCEDURE PMap
1231 ch=13:INK#ch,7:OVER#ch,-1 :REMark bx by Base Coordinates
1232 LINE#ch,bx,5 TO bx,95:LINE#ch,12,by TO 306,by
1233 CIRCLE#ch,bx,by,3:INK#ch,7:CIRCLE#ch,bx,by,9:OVER#ch,0
1234 END DEFine

```

Star Position Crosswires



```

1236 DEFINE PROCEDURE Encounter
1237 BEEP 0,1,9,100,1,0,0,0:PMap:bx=-9:by=-9:PMap
1238 Write 'Enemy Fighters Approaching'
1239 ch=2:BLOCK#ch,62,18,94,0,0:BLOCK#ch,62,18,346,0,0
1240 FOR alm=1 TO 12
1241   INK#2,2:IF alm MOD 2=0:INK#2,0
1242   EAS 41,96:EAS 123,96:PAUSE 5
1243 END FOR alm
1244 BEEP 0,60,3,900,1,0,0,0:INK 7:DPS 11:DPS 12
1245 Write 'Abort<Esc>:Target ← → Fire':BLOCK#16,16,3,130,4,7
1246 IF side(Gp1,6)>5:side(Gp1,6)=side(Gp1,6)-2:ELSE side(Gp1,6)=3
1247 RCharge:cs=0:cp=0:SStats:BEEP:turns=RND(3 TO 5):des=0
1248 FOR att=1 TO turns
1249   fx=RND(1 TO 3):IF Gp1=1:CG=fx:ELSE CG=fx+3
1250   xx=fx*60-24+8*RND(-1 TO 1) :yy=95:tx=95
1251 REPEAT att Ip
1252   IF side(1,5)<9 OR side(2,5)<9 OR cs>64 OR cp>64 :EXIT att
1253   IF side(1,6)<2 OR side(2,6)<2 OR Alnt<1 OR RInt<1 :EXIT att
1254   IF xx<10 OR xx>190 OR yy<5 OR des=1:des=0 :EXIT att Ip
1255   IF fx=1:xx=xx+RND(-3 TO 12)
1256   IF fx=2:xx=xx+RND(-12 TO 12)
1257   IF fx=3:xx=xx+RND(-12 TO 3)
1258   IF SPck=1 AND xx>90 AND xx<100:PFire tx,xx,yy
1259   yy=yy-RND(2 TO 5)
1260   INK 7:Enemy CG:INK 4:PSights tx,yy
1261   k=CODE(INKEY$(5)) :IF yy>30 AND RND(2)=1:FFire xx,yy
1262   INK 0:Enemy CG:INK 0:PSights tx,yy
1263   SELECT ON k
1264     = 27:EXIT att
1265     = 32:PFire tx,xx,yy:INK 7
1266     =192:tx=tx-5:IF tx< 20:tx= 20
1267     =200:tx=tx+5:IF tx>160:tx=160
1268   END SELECT
1269 END REPEAT att Ip
1270 END FOR att
1271 BLOCK#2,62,18,94,0,0:BLOCK#2,62,18,344,0,0
1272 sn=RND(1 TO 70):Gp=Gp1:HJump:DPS 11:DPS 12
1273 END Define

```



```

1275 DEFINE PROCEDURE EAS(ax,ay)
1276   FILL#ch,1:CIRCLE#ch,ax,ay,2:FILL#ch,0
1277   FOR a=0 TO .25 STEP .25
1278     FOR b=1 TO 3:ARC#ch,ax-b*2-a,ay+3 TO ax-b*2-a,ay-3,PI
1279     FOR b=1 TO 3:ARC#ch,ax+b*2+a,ay+3 TO ax+b*2+a,ay-3,-PI
1280   END FOR a
1281 END Define

```

Note: Early Warning Alarm System



```

1283 DEFINE PROCEDURE RCharge
1284   FOR i=1 TO 8
1285     BLOCK#11,i*8,4,20,90,4 :BLOCK#12,i*8,4,70-i*8,90,4:PAUSE 5
1286     BLOCK#11,8,i*8,60,72-i*8,7:BLOCK#12,8,i*8,18,72-i*8,7:PAUSE 5
1287   END FOR i
1288 END Define

```

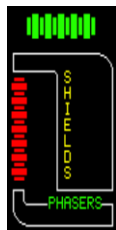
Note: Re-Charge Phasers & Shields



```

1290 DEFINE PROCEDURE DPS(ch) #14
1291 IF ch=11:CLS#11:INK#11,7:ch1=14:RESTORE 1304:fx1=36:fx2=48:fx4=8
1292 IF ch=12:CLS#12:INK#12,7:ch1=15:RESTORE 1306:fx1=12:fx2=36:fx4=70 #11
1293 CLS#ch1:FOR i=1 TO 5:BLOCK#ch1,5,14,8+i*10,2,4
1294     FOR i=1 TO 6:BLOCK#ch1,3,10,4+i*10,4,4
1295 FOR i=1 TO 2
1296     READ x1,y1,x2,y2,x3,y3,x4,y4,x5,y5,x6,y6,x7,y7,x8,y8
1297     LINE#ch,x1,y1 TO x2,y2 TO x3,y3:ARC#ch TO x4,y4,PI/2
1298     LINE#ch TO x5,y5 TO x6,y6 TO x7,y7:ARC#ch TO x8,y8,-PI/2:LINE#ch TO x1,y1
1299 END FOR i
1300 FOR i=1 TO 8:BLOCK#ch,8,2,fx4+2,6+i*8,2 #15
1301 FOR i=1 TO 7:BLOCK#ch,12,4,fx4,9+i*8,2
1302 INK#ch,4:CUSOR#ch,fx1,80:PRINT#ch,"PHASERS":str$="SHIELDS" #12
1303 INK#ch,6:FOR i=1 TO 7:CUSOR#ch,fx2,i*9:PRINT#ch,str$(i)
1304 DATA 7,22,13,22,13,16,16,14,64,14,64,2,17,2,7,12 :REMark Phaser #11
1305 DATA 32,22,64,22,64,90,54,100,7,100,7,90,29,90,32,87 :REMark Shield
1306 DATA 2,2,14,51,14,54,17,54,22,60,22,60,12,50,2 :REMark Phasor #12
1307 DATA 60,100,60,90,38,90,35,87,35,22,2,2,90,12,100 :REMark Shield
1308 END DEFINE

```



```

1310 DEFINE PROCEDURE FFire(xx,yy) Fighter Fire
1311 BEEP 3000,1,19,69,81,4,2,0:INK 2:cs=cs+2:tm=RND(-15 TO 15)
1312 BLOCK#11,8,cs,60,8,0:BLOCK#12,8,cs,18,8,0
1313 Phaser 2,xx,yy-2,95+tm,2:PAUSE 2:Phaser 0,xx,yy-2,95+tm,2
1314 END DEFINE

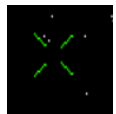
```



```

1316 DEFINE PROCEDURE PSights(tx,y) Phaser Sights
1317 LINE tx-8,y+8 TO tx-3,y+3:LINE tx+3,y+3 TO tx+8,y+8
1318 LINE tx-8,y-8 TO tx-3,y-3:LINE tx+3,y-3 TO tx+8,y-8
1319 END DEFINE

```



```

1321 DEFINE PROCEDURE PFire(tx,xx,yy) Phaser Fire
1322 BEEP 2000,1,255,200,4,2,0,0:INK 4:cp=cp+4
1323 BLOCK#11,cp,4,84-cp,90,0:BLOCK#12,cp,4,6,90,0
1324 Phaser 4,95,2,tx,yy:PAUSE 2:Phaser 0,95,2,tx,yy
1325 IF tx<xx-1 OR tx>xx+1:RETURN
1326 BEEP 20000,5,12,5,4,0,0,0:des=1 :REMark Enemy Destroyed
1327 INK 4:FOR i=3 TO 9 :CIRCLE xx,yy,i :PAUSE 2
1328 INK 0:PAUSE 5:FILL 1:CIRCLE xx,yy,15:FILL 0
1329 FOR i=1 TO 8:POINT RND(xx-9 TO xx+9),RND(yy-9 TO yy+9)
1330 IF Gp1=1:side(2,5)=side(2,5)-3:ELSE side(1,5)=side(1,5)-3
1331 IF Gp1=1:side(2,6)=side(2,6)-1:ELSE side(1,6)=side(1,6)-1
1332 IF Gp1=1:RInt=RInt-1:SStats :ELSE AInt=AInt-1:SStats
1333 END DEFINE

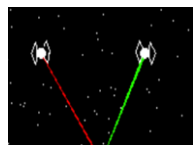
```



```

1335 DEFINE PROCEDURE Phaser(col,xx,yy,xt,yt)
1336 INK col:FILL 1:LINE xx,yy TO xt-.5,yt TO xt+.5,yt TO xx,yy:FILL 0
1337 END DEFINE

```



1339 **DEfIne PROCEDURE Enemy(CG)** Note: Select Fighter Image

1340 **SElect ON CG**

1341 =1:RF1 1,xx,yy

1342 =2:RF2 1,xx,yy

Republic Fighter

1343 =3:RF3 1,xx,yy

1344 =4:AF1 1,xx,yy

Alliance Fighter

1345 =5:AF2 1,xx,yy

1346 =6:AF3 1,xx,yy

1347 **END SElect**

1348 **END DEfIne**

1350 **DEfIne PROCEDURE AF1(ch,x,y)**

1351 **CIRCLE#**ch,x,y+2,1.2,.6,PI:**LINE#**ch,x-.5,y+3 TO x-4,y+5

1352 **FILL#**ch,1:**LINE#**ch,x,y TO x+4,y-4.5 TO x+3,y-5 TO x-4,y-1 TO x,y:**FILL#**ch,0

1353 **FILL#**ch,1:**CIRCLE#**ch,x-4.5,y+3,3,-PI/3:**FILL#**ch,0

1354 **FILL#**ch,1:**LINE#**ch,x-5,y TO x-9,y-2 TO x-11,y TO x-9,y+3 TO x-5,y:**FILL#**ch,0

1355 **CIRCLE#**ch,x-4,y,1.2,.6,PI:**LINE#**ch,x+3,y TO x,y+2

1356 **END DEfIne**



1358 **DEfIne PROCEDURE AF2(ch,x,y)**

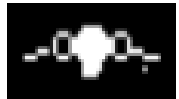
1359 **FILL#**ch,1:**LINE#**ch,x-2,y-1 TO x-1,y-4 TO x+1,y-4 TO x+2,y-1 TO x-2,y-1

1360 **FILL#**ch,0:**LINE#**ch,x-10,y-1.5 TO x-6,y-.5:**CIRCLE#**ch,x-4.5,y,1.6

1361 **FILL#**ch,1:**CIRCLE#**ch,x,y,2.5

1362 **FILL#**ch,0:**LINE#**ch,x+10,y-1.5 TO x+6,y-.5:**CIRCLE#**ch,x+4.5,y,1.6

1363 **END DEfIne**



1365 **DEfIne PROCEDURE AF3(ch,x,y)**

1366 **CIRCLE#**ch,x,y+2,1.2,.6,PI:**LINE#**ch,x+.5,y+3 TO x+4,y+5

1367 **FILL#**ch,1:**LINE#**ch,x,y TO x-4,y-4.5 TO x-3,y-5 TO x+4,y-1 TO x,y:**FILL#**ch,0

1368 **FILL#**ch,1:**CIRCLE#**ch,x+4.5,y+3,3,-PI/3:**FILL#**ch,0

1369 **FILL#**ch,1:**LINE#**ch,x+5,y TO x+9,y-2 TO x+11,y TO x+9,y+3 TO x+5,y:**FILL#**ch,0

1370 **CIRCLE#**ch,x+4,y,1.2,.6,PI:**LINE#**ch,x-3,y TO x,y+2

1371 **END DEfIne**



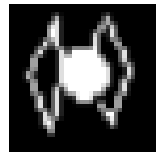
1373 **DEfIne PROCEDURE RF1(ch,x,y)**

1374 **LINE#**ch,x+2,y-3 TO x+3.5,y-5 TO x+5,y+1 TO x+3,y+6.4 TO x+1,y+3

1375 **FILL#**ch,1:**CIRCLE#**ch,x,y,2.5:**FILL#**ch,0

1376 **LINE#**ch,x-6,y TO x-3,y+6 TO x-2,y TO x-3.5,y-6 TO x-6,y

1377 **END DEfIne**



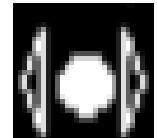
1379 **DEfIne PROCEDURE RF2(ch,x,y)**

1380 **LINE#**ch,x-6,y TO x-4,y+5 TO x-4,y-5 TO x-6,y :REMark L Wing

1381 **FILL#**ch,1:**CIRCLE#**ch,x,y,2.5:**FILL#**ch,0 :REMark Sphere

1382 **LINE#**ch,x+6,y TO x+4,y+5 TO x+4,y-5 TO x+6,y :REMark R Wing

1383 **END DEfIne**



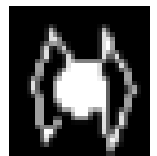
1385 **DEfIne PROCEDURE RF3(ch,x,y)**

1386 **LINE#**ch,x-2,y-3 TO x-3.5,y-5 TO x-5,y+1 TO x-3,y+6.4 TO x,y+1

1387 **FILL#**ch,1:**CIRCLE#**ch,x,y,2.5:**FILL#**ch,0

1388 **LINE#**ch,x+6,y TO x+3,y+6 TO x+2,y TO x+3.5,y-6 TO x+6,y

1389 **END DEfIne**



```

1391 DEFine PROCEDURE HJump
1392 Write 'Hyperspace Jump to '&astro$(sn):CURSOR 0,43
1393 GTime:wx=96:wy=50:ws=3:INK 241:BEEP 0,9,36,1000,1,0,5,5
1394 FOR a=42 TO 24 STEP -3:CIRCLE wx,wy,a:PAUSE a/4
1395 FOR a=1 TO 15
1396   FOR b=0 TO 3:CIRCLE wx,wy,20+b*15,.6,PI/2
1397   StarMove:BLOCK#2,60,10,96,6,0:INK 241:PAUSE 2
1398 END FOR a
1399 BEEP 0,60,90,3000,1,5,6,6:FOR a=12 TO 1 STEP -1:ws=a:StarMove:PAUSE 14-a
1400 GTime:SSInfo:INK 7
1401 END DEFine

```



```

1403 DEFine PROCEDURE StarMove
1404 hoz=ws:ver=ws:INK#2,7
1405 CURSOR#2,96,6:PRINT#2,'Warp:':FILL$(' ',2-LEN(ws))&ws
1406 IF ws>0:INK#1,7:FOR w=1 TO ws*2:POINT#1,RND(40 TO 160),RND(30 TO 70)
1407 SCROLL#4,-ver*2:SCROLL#9,ver*2:PAN#6,-hoz*2:PAN#7,hoz*2
1408 PAN#3,-hoz*2:SCROLL#3,-ver:PAN#5,hoz*2:SCROLL#5,-ver
1409 PAN#8,-hoz*2:SCROLL#8, ver:PAN#10,hoz*2:SCROLL#10,ver
1410 END DEFine

```



```

1412 DEFine PROCEDURE StarView
1413 Write 'Approaching '&astro$(sn):sy=80:sx=16+(astro(sn,11)*40)
1414 FOR r=1 TO 9:INK 0:FILL 1:CIRCLE sx,sy,r*2:FILL 0:sr=r:Star:StarMove:PAUSE 2
1415 END DEFine

```

```

1417 DEFine PROCEDURE Star
1418 INK 7:FILL 1:CIRCLE sx,sy,sr:FILL 0:INK 2,7,3:CIRCLE sx,sy,sr
1419 INK 2:FILL 1:CIRCLE sx-sr/2,sy-sr/3,sr/6,.3,RAD(45):FILL 0
1420 END DEFine

```

```

1422 DEFine PROCEDURE ASS(ch,col,x,y)
1423 INK#ch,col:FILL#ch,1:CIRCLE#ch,x,y,8,.5,PI/2:FILL#ch,0
1424 FILL#ch,1:CIRCLE#ch,x,y+4,4,.6,PI/2:FILL#ch,0
1425 CIRCLE#ch,x-9,y,3,.6,PI:CIRCLE#ch,x+9,y,3,.6,PI
1426 FILL#ch,1:CIRCLE#ch,x,y-5,6,.2,PI:FILL#ch,0
1427 INK#ch,0:LINE#ch,x-2.5,y+4 TO x+2.5,y+4:INK#ch,7
1428 END DEFine

```



```

1430 DEFine PROCEDURE RSS(ch,col,x,y)
1431 INK#ch,col:LINE#ch,x-9,y-2 TO x,y-7.5 TO x+9,y-2
1432 FILL#ch,1:LINE#ch,x-2.5,y+5 TO x+2.5,y+5 TO x+4,y+2.5 TO x+12,y-5
1433 LINE#ch TO x,y-4 TO x-12,y-5 TO x-4,y+2.5 TO x-2,y+5:FILL#ch,0
1434 FILL#ch,1:CIRCLE#ch,x,y+8,6,.2,PI/2:FILL#ch,0
1435 INK#ch,0:LINE#ch,x-2.8,y+2 TO x+3,y+2:INK#ch,7
1436 END DEFine

```



1438 **DEFINE PROCEDURE EOrbit**

1439 IF side(Gp,6)>3:side(Gp,6)=side(Gp,6)-RND(1 TO 3):ELSE side(Gp,6)=3

1440 **SFuel:Write astro\$(sn)&' - Planetary Orbit'**

1441 x=96:INK#2,7:CURSOR#2,96,6:PRINT#2,'Impulse '

1442 **FOR a=1 TO 18**

1443 POINT RND(10 TO 180),RND(60 TO 80):SCROLL -1:GTime

1444 IF astro(sn,11)=1 :x=18+a\*3

1445 IF astro(sn,11)=3 :x=182-(a\*3)

1446 INK astro(sn,10) :y=24-a\*3

1447 FILL 1:CIRCLE x,y,a\*2:FILL 0:PAUSE a/3:INK 0

1448 FILL 1:CIRCLE x,y,a\*2:FILL 0

1449 INK 7 :POINT RND(40 TO 160),RND(30 TO 40)

1450 **END FOR a**

1451 **GTime:rx=astro(sn,10)-20**

1452 IF rx<8 AND astro(n,11)<>3:c1=5:c3=2.5:ELSE c1=1:c3=1.2:INK#1,7

1453 IF rx>18 AND astro(n,11)=3:c2=6:ELSE c2=1

1454 FILL#1,1:CIRCLE#1,rx,15,c1 :FILL#1,0

1455 FILL#1,1:CIRCLE#1,170+rx,52,c2 :FILL#1,0

1456 FILL#1,1:CIRCLE#1,118+rx,110-rx,c3:FILL#1,0

1457 IF rx<6:INK#1,2:ARC#1,114+rx,112-rx TO 122+rx,108-rx,PI/2:INK#1,7

1458 **FOR b=1 TO 14**

1459 INK 7:POINT RND(10 TO 180),RND(10 TO 20)

1460 IF Gp=1:x=96-40:y=50:ELSE x=96+40:y=50

1461 INK astro(sn,10):FILL 1:CIRCLE 96,-200+b,198:FILL 0

1462 INK 241:CIRCLE x,y,b-2:LINE x-b,y TO x+b,y:LINE x,y-b TO x,y+b:PAUSE b/4

1463 INK 0:CIRCLE x,y,b-2:LINE x-b,y TO x+b,y:LINE x,y-b TO x,y+b

1464 **END FOR b**

1465 IF Gp=1:**ASS 1,7,56,52:ELSE RSS 1,7,136,52**

1466 INK#2,7:CURSOR#2,96,6:PRINT#2,'Orbit ' :BEEP

1467 **END DEFINE**



1469 **DEFINE PROCEDURE SFuel**

1470 BLOCK#17,60,5,32,45,0:FOR i=1 TO side(1,6):BLOCK#17,3,5,26+i\*6,45,5

1471 BLOCK#18,60,5, 6,45,0:FOR i=1 TO side(2,6):BLOCK#18,3,5,68 -i\*6,45,3

1472 **END DEFINE**



1474 **DEFINE PROCEDURE LOrbit**

1475 **Write 'Leaving Orbit':INK#2,7:CURSOR#2,96,6:PRINT#2,'Impulse '**

1476 BEEP 0,60,18,900,1,0,0,0:IF Gp=1:**ASS 1,0,56,52:ELSE RSS 1,0,136,52**

1477 IF Gp=1:x=60+42-(b\*3):y=50:ELSE x=140-42+(b\*3):y=55

1478 **FOR b=15 TO 1 STEP -2**

1479 INK 241:CIRCLE x,y,b-4:LINE x-b,y TO x+b,y:LINE x,y-b TO x,y+b:PAUSE a/2

1480 INK 0:CIRCLE x,y,b-4:LINE x-b,y TO x+b,y:LINE x,y-b TO x,y+b:INK 7

1481 CURSOR 0,80:SCROLL -1,1:PAN#6, -1

1482 CURSOR 0,40:SCROLL 2,2:PAN#7, 1

1483 **FOR i=1 TO 3:POINT RND(20 TO 180),RND(10 TO 40)**

1484 **END FOR b**

1485 **GTime:CLS#16:DiceCLS:ws=1:FOR a=1 TO 4:StarMove:PAUSE 2**

1486 **END DEFINE**



1488 **DEfINE PROCEDURE GPlay**

1489 **REPeat act\_lp**

1490 IF APs=0 OR RPs=0:**EXIT act\_lp**

1491 IF ack=1 AND astro(n,7)=Gp:ack=0:Mes\$='↑ ↓ ↔Transfer →':**SSInfo**

1492 IF ack=1 AND astro(n,5)=0 :ack=0:Mes\$='↑ ↓ ↔Transfer →':**Victory**

1493 IF ack=1 AND side(Gp,5)>dw+3

1494 Mes\$=' Attack Mode ':**SSInfo:RollDice:SSInfo**

1495 **END IF**

1496 IF ack=1 AND side(Gp,5)<dw+3

1497 ack=0:**Write 'Abort low on WMDs':PAUSE 50:EXIT act\_lp**

1498 **END IF**

1499 IF ack=0:ch=11:BLOCK#ch,10,60,74,30,0:CURSOR#ch,74,20+↑\*10:PRINT#ch,↑

1500 IF ack=0:ch=12:BLOCK#ch,10,60,4,30,0:CURSOR#ch,4,20+↑\*10:PRINT#ch,↑

1501 **GTime:k=CODE(INKEY\$(25))**

1502 **SElect ON k**

1503 =192:**TranPS** :REMark Transfer Planet > Star

1504 =200:**TranSP** :REMark Transfer Star > Planet

1505 =208:t=t-1:IF t<1:t=6 :REMark Select Food < > Fuel

1506 =216:t=t+1:IF t>6:t=1

1507 = 32:**EXIT act\_lp**

1508 = 78,104:IF ack=0 AND vck=1:**Harvest** :REMark (H)arvest [**Increased Industrial Output - WMD's**]

1509 **END SElect**

1510 **END REPeat act\_lp**

1511 **END DEfINE**

```
ALLIANCE
SShip: 2
Moral: 2
Food: 4
Ores: 2
Tech: 6
Trade: 6
WMD: 20
Fuel: 9
Attack Mode
```

```
ALLIANCE
SShip: 1
Moral: 3
Food: 6 >
Ores: 3
Tech: 6
Trade: 6
WMD: 21
Fuel: 3
↑↓ ↔Transfer↔
```

1513 **DEfINE PROCEDURE TranSP**

1514 min(5)=16:min(6)=5:FOR i=1 TO 4:min(i)=1+side(Gp,5) DIV 20

1515 max(1)=20:max(2)=8:max(3)=12:max(4)=16:max(5)=16:max(6)=5

1516 IF side(Gp,t)<=min(t) OR astro(n,t)=max(t):**RETurn**

1517 side(Gp,t)=side(Gp,t)-1:astro(n,t)=astro(n,t)+1:SSInfo

1518 **END DEfINE**

Transfer Star Ship to Planet [**Population increase Advances Technology & Production**]

1520 **DEfINE PROCEDURE Harvest**

1521 ch=12:vck=0:PAUSE 20:BLOCK#ch,90,10,0,90,0:Popu=1+INT(astro(n,12))

1522 astro(n,1)=astro(n,1)+Popu DIV 2 :IF astro(n,1)>20:astro(n,1)=20

1523 astro(n,2)=astro(n,2)+Popu DIV 2 :IF astro(n,2)> 8:astro(n,2)= 8

1524 astro(n,3)=astro(n,3)+Popu DIV 3 :IF astro(n,3)>12:astro(n,3)=12

1525 astro(n,4)=astro(n,4)+Popu DIV 3 :IF astro(n,4)>16:astro(n,4)=16

1526 astro(n,5)=astro(n,5)+astro(n,3) :IF astro(n,5)>16:astro(n,5)=16

1527 astro(n,6)=astro(n,6)+astro(n,3) DIV 3:IF astro(n,6)> 5:astro(n,6)= 5

1528 astro(n,12)=astro(n,1)/2.1:SSInfo

1529 **Write 'To leave Orbit press '**:BLOCK#16,16,3,150,4,7:PAUSE 30

1530 **END DEfINE**

```
ALLIANCE
2.5b:Popu
2 :Moral
< 6 :Food
4 :Ores
3 :Tech
3 :Trade
0 :WMD
1 :Fuel
(H)arvest
```

```
ALLIANCE
SShip: 1
Moral: 3
Food: 6 >
Ores: 3
Tech: 6
Trade: 6
WMD: 21
Fuel: 3
↑↓ ↔Transfer↔
```

1532 **DEfINE PROCEDURE TranPS**

Transfer Planet to Star Ship

1533 min(6)=0:FOR i=1 TO 5:min(i)=1

1534 max(1)=20:max(2)=8:max(3)=12:max(4)=16:max(5)=99:max(6)=10

1535 IF astro(n,t)=min(t) OR side(Gp,t)=max(t):**RETurn**

1536 side(Gp,t)=side(Gp,t)+1:astro(n,t)=astro(n,t)-1

1537 IF astro(n,1)<astro(n,12):astro(n,12)=astro(n,1):**PTMoral**

1538 Ship=1+(side(Gp,5)DIV 20):**SSInfo**

1539 **END DEfINE**

```
ALLIANCE
7.1b:Popu
6 :Moral
< 15 :Food
8 :Ores
9 :Tech
8 :Trade
15 :WMD
5 :Fuel
```

# 1541 DEFINE PROCEDURE SStats

```

1542 AWMD=0:APs=0:ASSn=0:RWMD=0:RPs=0:RSSn=0:SFuel
1543 FOR i=1 TO 70:IF astro(i,7)=1:APs=APs+1:AWMD=AWMD+astro(i,5)
1544 FOR i=1 TO 70:IF astro(i,7)=2:RPs=RPs+1:RWMD=RWMD+astro(i,5)
1545 IF wck=0:SET_WMD:wck=1
1546 BLOCK#17,20,10,0,12,0:CURSOR#17,0,12:PRINT#17,AWMD :REMark All Land WMD
1547 BLOCK#17,20,10,0,22,0:CURSOR#17,0,22:PRINT#17,side(1,5) :REMark SS WMD
1548 ASSn=1+side(1,5) DIV 21:CURSOR#17,78,22:PRINT#17,' ',ASSn :REMark All SShips
1549 CURSOR#17,78,32:PRINT#17,FILL$(' ',2-LEN(Alnt))&Alnt :REMark All Fighters
1550 CURSOR#17,78,52:PRINT#17,FILL$(' ',2-LEN(APs))&APs :REMark All SSystems
1551 CURSOR#18,78,12:PRINT#18,FILL$(' ',3-LEN(RWMD))&RWMD :REMark Rep Land WMD
1552 CURSOR#18,84,22:PRINT#18,FILL$(' ',2-LEN(side(2,5)))&side(2,5) :REMark SS WMD
1553 RSSn=1+side(2,5) DIV 21:CURSOR#18,12,22:PRINT#18,RSSn :REMark Rep SShip
1554 CURSOR#18,6,32:PRINT#18,FILL$(' ',2-LEN(RInt))&RInt :REMark Rep Fighters
1555 CURSOR#18,6,52:PRINT#18,FILL$(' ',2-LEN(RPs))&RPs :REMark Rep SSystems
1556 END DEFINE

```

# 1558 DEFINE PROCEDURE Set\_WMD

```

1559 IF AWMD > RWMD:side(2,5)=20+AWMD -RWMD:side(1,5)=20
1560 IF RWMD > AWMD:side(1,5)=20+RWMD-AWMD:side(2,5)=20
1561 IF RWMD=AWMD:side(1,5)=20:side(2,5)=20
1562 side(1,6)=10:side(2,6)=10:SFuel
1563 END DEFINE

```



# 1565 DEFINE PROCEDURE RollDice

```

1566 ch=2:GTime:IF SPck=0 AND Gp=Gp1:Write 'Press any key to Fire':PAUSE
1567 xt=RND(80 TO 120):ys=46:IF Gp=1:xs=60:ELSE xs=132
1568 FOR a=1 TO RND(4 TO 6)
1569 dx= 13:DiceA:PAUSE 4:DiceB:Phaser 2,xs,ys,xt,12:BEEP 0,1,19,69,81,43,29,0
1570 dx=151:DiceA:PAUSE 4:DiceB:Phaser 0,xs,ys,xt,12:BEEP
1571 END FOR a
1572 OVER#ch,1:dt1=RND(1 TO 8):dx=13:DiceA:CURSOR#ch,dx,dy,-3,-4:PRINT#ch,dt1
1573 dt2=RND(1 TO 8):dx=151:DiceA:CURSOR#ch,dx,dy,-3,-4:PRINT#ch,dt2:OVER#ch,0
1574 IF dt1>dt2:astro(n,5)=astro(n,5)-dw:IF astro(n,5)<1:astro(n,5)=0
1575 IF dt1=dt2 OR dt1<dt2:side(Gp,5)=side(Gp,5)-dw
1576 END DEFINE

```

# 1578 DEFINE PROCEDURE DiceA

```

1579 dy=39:I=2.5:INK#ch,0:FILL#ch,1:CIRCLE#ch,dx,dy,3.5:FILL#ch,0:INK#ch,7
1580 LINE#ch,dx-l,dy+l TO dx+l,dy+l TO dx+l,dy-l TO dx-l,dy-l TO dx-l,dy+l
1581 END DEFINE

```



# 1583 DEFINE PROCEDURE DiceB

```

1584 dy=39:I=2.5:INK#ch,0:FILL#ch,1:CIRCLE#ch,dx,dy,3.5:FILL#ch,0:INK#ch,7
1585 LINE#ch,dx,dx+l TO dx+l,dy TO dx,dy-l TO dx-l,dy TO dx,dy+l
1586 END DEFINE

```



# 1588 DEFINE PROCEDURE DiceCLS

```

1589 ch=2:INK#ch,0:dy=39:dx=13:FILL#ch,1:CIRCLE#ch,dx,dy,4:FILL#ch,0
1590 dx=151:FILL#ch,1:CIRCLE#ch,dx,dy,4:FILL#ch,0:INK#ch,7:GPnt 2
1591 END DEFINE

```



# 1593 DEFINE PROCEDURE GPnt(pnt)

```

1594 IF pnt=0:x=8:x1=2:ELSE x=156:x1=162
1595 FILL#2,1:LINE#2,x,40 TO x1,44 TO x1,42 TO x,40:FILL#2,0
1596 IF pnt=2:BLOCK#2,30,10,2,126,0:BLOCK#2,30,10,470,126,0
1597 END DEFINE

```





```

GALAXY GRID
x:131 y:54

ALLIANCE
SShip: 2
Moral: 3
Food: 5
Ores: 3
Tech: 6
Trade: 6
WMD: 33
Fuel: 6

```

```

1599 DEFINE PROCEDURE SSInfo
1600 SStats:SSMoral:PTInfo:ch=11:IF side(Gp,6)<3:side(Gp,6)=3
1601 IF Gp=1:Gp$='ALLIANCE':ss=RSSn:INK#ch,5:INK#2,5
1602 IF Gp=2:Gp$='REPUBLIC':ss=FSsn:INK#ch,3:INK#2,3
1603 CLS#ch:QBold ch,7,0,0,Gp$:SFuel:CIRCLE#2,13,39,4:GPnt 0
1604 PRINT#ch,' SShip':FILL$(' ',4-LEN(ss))&ss
1605 PRINT#ch,' Moral':FILL$(' ',4-LEN(smv))&smv
1606 PRINT#ch,' Food':FILL$(' ',4-LEN(side(Gp,1)))&side(Gp,1)
1607 PRINT#ch,' Ores':FILL$(' ',4-LEN(side(Gp,2)))&side(Gp,2)
1608 PRINT#ch,' Tech':FILL$(' ',4-LEN(side(Gp,3)))&side(Gp,3)
1609 PRINT#ch,' Trade':FILL$(' ',4-LEN(side(Gp,4)))&side(Gp,4)
1610 PRINT#ch,' WMD':FILL$(' ',4-LEN(side(Gp,5)))&side(Gp,5)
1611 PRINT#ch,' Fuel':FILL$(' ',4-LEN(side(Gp,6)))&side(Gp,6)
1612 CURSOR#ch,4,90:PRINT#ch,Mes$
1613 IF ack=0 AND vck=1:ch=12:CURSOR#ch,0,90:PRINT#ch,' (H)arvest '
1614 END DEFINE

```

```

:
1616 DEFINE PROCEDURE SSMoral
1617 smv=1+INT((side(Gp,1)+side(Gp,2)+side(Gp,3))DIV 7) :REMark Ship Moral value
1618 END DEFINE

```

```

1620 DEFINE PROCEDURE Victory
1621 Write 'Transfer then - (H)arvest':BEEP
1622 side(Gp,1)=side(Gp,1)+astro(n,1)DIV 3 :IF side(Gp,1)>20:side(Gp,1)=20
1623 side(Gp,2)=side(Gp,2)+astro(n,2)DIV 4 :IF side(Gp,2)> 8:side(Gp,2)= 8
1624 side(Gp,3)=side(Gp,3)+astro(n,3)DIV 4 :IF side(Gp,3)>12:side(Gp,3)=12
1625 side(Gp,4)=side(Gp,4)+astro(n,4)DIV 4 :IF side(Gp,4)>16:side(Gp,4)=16
1626 side(Gp,5)=side(Gp,5)+1+INT(astro(n,12)):IF side(Gp,5)>99:side(Gp,5)=99
1627 astro(n,7)=Gp:LMap:vck=1:SSInfo
1628 END DEFINE

```

```

1630 DEFINE PROCEDURE PTMoral
1631 pmv=1+INT((astro(n,1)+astro(n,2)+astro(n,3)+astro(n,5)) DIV 9)
1632 END DEFINE

```

```

1634 DEFINE PROCEDURE PTInfo
1635 PMap:ch=12 :Sp$='INDEPENDANT':INK#ch,6:INK#2,6
1636 IF astro(n,7)=1:Sp$=' ALLIANCE ':INK#ch,5:INK#2,5
1637 IF astro(n,7)=2:Sp$=' REPUBLIC ':INK#ch,3:INK#2,3
1638 bx=astro(n,8):by=astro(n,9):Pop$=astro(n,12):PTMoral
1639 CLS#14:PRINT#14,'GALAXY GRID "x:"bx' y:"by' '
1640 CLS#15:PRINT#15,' STAR SYSTEM'FILL$(' ',12-LEN(astro$(n)))&astro$(n)
1641 CLS#ch:QBold ch,7,-2,0,Sp$:CIRCLE#2,151,39,4:GPnt 1
1642 IF ' ' INSTR Pop$>=0:Pop$=Pop$&' 0'
1643 IF ' ' INSTR Pop$+1>0:Pop$=Pop$(1 TO ' ' INSTR Pop$+1)
1644 PRINT#ch,FILL$(' ',6-LEN(Pop$))&Pop$&'b:Popu'
1645 PRINT#ch,FILL$(' ',6-LEN(pmv))&pmv&' :Moral'
1646 PRINT#ch,FILL$(' ',6-LEN(astro(n,1)))&astro(n,1)&' :Food'
1647 PRINT#ch,FILL$(' ',6-LEN(astro(n,2)))&astro(n,2)&' :Ores'
1648 PRINT#ch,FILL$(' ',6-LEN(astro(n,3)))&astro(n,3)&' :Tech'
1649 PRINT#ch,FILL$(' ',6-LEN(astro(n,4)))&astro(n,4)&' :Trade'
1650 PRINT#ch,FILL$(' ',6-LEN(astro(n,5)))&astro(n,5)&' :WMD'
1651 PRINT#ch,FILL$(' ',6-LEN(astro(n,6)))&astro(n,6)&' :Fuel':PMap
1652 END DEFINE

```

```

STAR SYSTEM
Enif
INDEPENDANT
3.8b:Popu
3 :Moral
7 :Food
5 :Ores
4 :Tech
4 :Trade
3 :WMD
1 :Fuel

```

## 1654 REMark Opponent & Simulation Mode

1656 **DEFine PROCEDURE GStarSel**

1657 **REPeat WMD\_Ip**

System

1658 sn=RND(1 TO 70):IF WMD>11 AND WMD<50:**EXIT WMD\_Ip**

1659 IF WMD>49 AND astro(sn,7)<>Gp:**EXIT WMD\_Ip**

Attack)

1660 IF WMD<12 AND astro(sn,7) = Gp:**EXIT WMD\_Ip**

1661 **END REPeat WMD\_Ip**

1662 n=sn:**SSInfo:PTInfo:HJump:StarView:EOrbit**

1663 IF astro(n,7)=Gp:**GSTrans:ELSE GSAtt**

1664 **END DEFine**

**Note:** Random selection of Star

Go with any Star System  
Independent or Enemy Stars (for

Annexed Stars (Restock WMDs

1666 **DEFine PROCEDURE GSAtt**

1667 Mes\$=' Attack Mode ':**SSInfo**

1668 **REPeat Att\_Ip**

1669 IF side(Gp,5)<astro(n,5)\*3 OR side(Gp,5)<6:**EXIT Att\_Ip**

1670 IF astro(n,5)=0

1671 **Victory:PAUSE 50:astro(n,7)=Gp:LMap:GSTrans:EXIT Att\_Ip**

1672 **END IF**

1673 **RollDice:SSInfo:PAUSE 50**

1674 **END REPeat Att\_Ip**

1675 **END DEFine**

Attack Mode

1677 **DEFine PROCEDURE GSTrans**

1678 Mes\$=' ↑ ↓ ← Transfer → ':**SSInfo**

1679 FOR i=1 TO 4:min(i)=1+side(Gp,5) DIV 20

1680 max(1)=20:max(2)=8:max(3)=12:max(4)=16

1681 FOR t=1 TO 4

1682 **REPeat SP\_Ip**

1683 **PAUSE 5:IF side(Gp,t)=min(t) OR astro(n,t)=max(t):EXIT SP\_Ip**

1684 side(Gp,t)=side(Gp,t)-1:astro(n,t)=astro(n,t)+1:**SSInfo**

1685 **END REPeat SP\_Ip**

1686 **END FOR t**

1687 ack=0:vck=1:**SSInfo:PAUSE 30:Harvest:vck=0:SSInfo:PAUSE 30**

1688 min(1)=1+INT(astro(n,12)):min(5)=3:min(6)=0:FOR i=2 TO 4:min(i)=1

1689 max(1)=20:max(2)=8:max(3)=12:max(4)=16:max(5)=99:max(6)=10

1690 FOR t=1 TO 6

1691 **REPeat PS\_Ip**

1692 **PAUSE 5:IF astro(n,t)=min(t) OR side(Gp,t)=max(t):EXIT PS\_Ip**

1693 side(Gp,t)=side(Gp,t)+1:astro(n,t)=astro(n,t)-1:**SSInfo**

1694 **END REPeat PS\_Ip**

1695 **END FOR t**

1696 IF side(Gp,5)>50 AND astro(n,5)<10

1697 astro(n,1)=astro(n,5)+5:side(Gp,5)=side(Gp,5)-5

1698 **END IF**

1699 **SSInfo:PAUSE 30**

1700 **END DEFine**

Transfer Mode

Star Ship to Planet Transfers

Planet to Star Ship Transfers

1702 **DEFine PROCEDURE GSwap**

1703 **SSInfo:LOrbit:ws=1:PMap:bx=-9:by=-9:n=RND(20 TO 50)**

1704 **BLOCK#2,60,10,348,6,0:DPS 11:DPS 12:DiceCLS:Mes\$="**

1705 **IF APs=0 OR RPs=0:GEnd**

1706 **END DEFine**

Move to next Turn

If Planet count zero Game End

## 1708 REMark Part 3 End Game

### 1710 DEFine PROCEDURE GEnd

```

1711 dx=60:dy=80:ws=1:DeathStar:wx=156:wy=28:CSIZE 2,1
1712 IF RPs=0
1713   BEEP 2000,20,30,0,0,0,0:INK 2:DSDefeat:PAUSE 10:ASS 1,7,50,20
1714   OVER 1:FOR i=1 TO 2:CUSOR 60+i,40:PRINT 'ALLIANCE':OVER 0
1715 END IF
1716 IF APs=0
1717   BEEP 2000,20,30,0,0,0,0:INK 2:x=56:y=80:DSAttack:PAUSE 10:RSS 1,7,50,20
1718   OVER 1:FOR i=1 TO 2:CUSOR 84+i,40:PRINT 'REPUBLIC':OVER 0
1719 END IF
1720 CSIZE 2,0:INK 7:OVER 1:FOR i=1 TO 2:CUSOR 102+i,60:PRINT 'Victory'
1721 OVER 0:CSIZE 0,0:INK 4:LName:nck=1:GMenu
1722 END DEFine

```

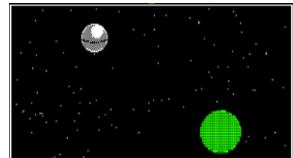
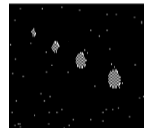


### 1724 DEFine PROCEDURE DeathStar

```

1725 FOR a=1 TO 16
1726   INK astro(n,10):wx=80+a*4:wy=-46+a*4
1727   FILL 1:CIRCLE wx,wy,50-a*2:FILL 0
1728   INK 248:FILL 1:CIRCLE 20+a*2,90-a,a/3:FILL 0:PAUSE 4
1729   INK 0:FILL 1:CIRCLE 20+a*2,90-a,a/3:FILL 0
1730   FILL 1:CIRCLE wx,wy,50-a*2:FILL 0
1731 END FOR a
1732 INK astro(n,10):FILL 1:CIRCLE wx,wy,14:FILL 0
1733 INK 248:FILL 1:CIRCLE 56,80,9:FILL 0
1734 INK 7:CIRCLE 56,80,9:INK 0:ARC 44,80 TO 68,80,PI/3
1735 INK 7:FILL 1:CIRCLE 58,84,4:FILL 0
1736 END DEFine

```



### 1738 DEFine PROCEDURE DSAttack

```

1739 FOR a=1 TO 6
1740   INK 2:FILL 1:LINE dx+2,dy+4 TO wx-16,wy-6 TO wx-15,wy-4 TO dx+2,dy+4
1741   FILL 0:PAUSE 3:INK 0:FILL 1
1742   LINE dx+2,dy+4 TO wx-16,wy-6 TO wx-15,wy-4 TO dx+2,dy+4:FILL 0:PAUSE 2
1743 END FOR a
1744 INK 7:FILL 1:CIRCLE dx-1,dy+4,4:FILL 0:BEEP 20000,20,10,2,3,4,5,0
1745 INK 0:FILL 1:CIRCLE wx-16,wy-6,16:FILL 0
1746 INK 7:FOR b=7 TO 10:CIRCLE wx-b,wy-12,b*4,.2,PI/2:PAUSE 5
1747 INK astro(n,10)
1748 FOR c=1 TO 12:FILL 1:CIRCLE wx+40+c*4,wy+RND(-12 TO 4),RND(2 TO 3):FILL 0
1749 END DEFine

```

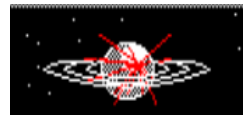


### 1751 DEFine PROCEDURE DSDefeat

```

1752 FOR i=1 TO 18
1753   x1=56+RND(-3 TO 3)*5:y1=80+RND(-3 TO 3)*4:LINE dx,dy+2 TO x1,y1:PAUSE 1
1754 END FOR i
1755 INK 7:dx=dx-4:BEEP 20000,12,4,8,0,0,0:PAUSE 10
1756 FOR i=2 TO 4
1757   x1=dx-i*4:x2=dx-i*7:x3=dx+i*7:x4=dx+i*4:y1=dy:y2=dy+3
1758   ARC x1,y2 TO x2,y1,PI/4 TO x3,y1,PI/4 TO x4,y2,PI/4:PAUSE
1759 END FOR i
1760 END DEFine

```



## 1762 REMark Part 4 League Table

### 1764 DEFine PROCEDURE LTDefault

1765 REMark Score League Table

```
1766 name$(1)='SPlay ' :score(1,1)=5072:score(1,2)=1024:score(1,3)=58
1767 name$(2)='CPlay ' :score(2,1)=3765:score(2,2)=1185:score(2,3)=62
1768 name$(3)='QBITS ' :score(3,1)=3452:score(3,2)=1296:score(3,3)=66
1769 END DEFINE
```

League Table				
Skill	Time h:m:s	WMD	Stars	Gamer
Marshal	01:24:32	1024	58	SPlay
Admiral	01:02:45	1185	62	CPlay
Captain	00:57:32	1296	66	QBITS

### 1771 DEFine PROCEDURE LTReset

1772 REMark Resets League Table

```
1773 name$(1)=' ' :score(1,1)=0:score(1,2)=0:score(1,3)=0
1774 name$(2)=' ' :score(2,1)=0:score(2,2)=0:score(2,3)=0
1775 name$(3)=' ' :score(3,1)=0:score(3,2)=0:score(3,3)=0
1776 END DEFINE
```

League Table				
Skill	Time h:m:s	WMD	Stars	Gamer
Marshal	00:00:00	0	0	
Admiral	00:00:00	0	0	
Captain	00:00:00	0	0	

### 1778 DEFine PROCEDURE LScore

1779 ch=13:GFile\$='QBAD2375DLT'

1780 IF LTck=0:FCHECK:IF pck=0:LTDefault:LSave:ELSE LLoad

1781 OVER#ch,1:CSIZE#ch,2,0:INK#ch,7:CLS#ch

1782 FOR i=1 TO 2:CUSOR#ch,60+i,2:PRINT#ch,'League Table'

1783 OVER#ch,0:CSIZE#ch,0,0:INK#ch,5

1784 CUSOR#ch,6,16:PRINT#ch,'Skill Time h:m:s WMD Stars Gamer'

1785 CUSOR#ch,6,28:PRINT#ch,'Marshal\' Admiral\' Captain'

1786 FOR a=1 TO 3

1787 HST\$=DATE\$(score(a,1)):HS2=score(a,2):HS3=score(a,3):INK#ch,3

1788 CUSOR#ch, 60,18+a\*10:PRINT#ch,HST\$(13 TO 20)

1789 CUSOR#ch,110,18+a\*10:PRINT#ch,FILL\$(' ',6-LEN(HS2))&HS2

1790 CUSOR#ch,166,18+a\*10:PRINT#ch,FILL\$(' ',3-LEN(HS3))&HS3:INK#ch,6

1791 CUSOR#ch,192,18+a\*10:PRINT#ch,' :name\$(a)

1792 END FOR a

1793 END DEFINE

### 1795 DEFine PROCEDURE LName

1796 GTS=DATE-Gclk+GTS:Gclk\$=DATE\$(GTS):IF LTch=0:LScore

1797 IF score(GSk,1)>0 AND score(GSk,1)<GTS:RETURN

1798 IF Gp1=1:score(GSk,2)=AWMD+side(1,5):score(GSk,3)=APs

1799 IF Gp1=2:score(GSk,2)=RWMD+side(2,5):score(GSk,3)=RPs

1800 score(GSk,1)=GTS:BLOCK#20,50,10,390,230,0:LTch=1:LScore

1801 OPEN\_NEW#99,con\_60x10a10x10\_10

1802 WINDOW#99,60,10,320+gx,177+gy+GSk\*10:PAPER#99,0:CLS#99

1803 INK#99,6:INPUT#99,name\$(GSk):CLOSE#99:LSave

1804 END DEFINE

League Table				
Skill	Time h:m:s	WMD	Stars	Gamer
Marshal	00:00:00	0	0	
Admiral	00:00:00	0	0	
Captain	01:13:22	68	10	QBITS

### 1806 DEFine PROCEDURE LSave

1807 DELETE Drv\$(dn%)&'QBAD2375DLT':OPEN\_NEW#99,Drv\$(dn%)&'QBAD2375DLT'

1808 FOR a=1 TO 3:PRINT#99,name\$(a)\score(a,1)\score(a,2)\score(a,3)

1809 CLOSE#99:LTck=1

1810 END DEFINE

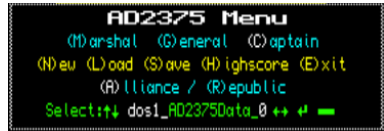
### 1812 DEFine PROCEDURE LLoad

1813 OPEN\_IN#99,Drv\$(dn%)&'QBAD2375DLT'

1814 FOR a=1 TO 3:INPUT#99,name\$(a)\score(a,1)\score(a,2)\score(a,3)

1815 CLOSE#99:LTck=1

1816 END DEFINE

1820 **DEFine PROCEDURE GLoad**1821 ch=13:SelPath:IF pck=0:BLOCK#ch,250,10,0,50,0:RETu**rn** :ELSE F**C**heck

1822 IF pck=0 OR eck=1

1823 CURSOR#ch,24,50:PRINT#ch,' File NOT Found':CLS#ch,4

File NOT Found

1824 PAUSE 30:CURSOR#ch,0,50:CLS#ch,4:eck=0:RETu**rn**

1825 END IF

1826 OPEN \_IN#99,Drv\$(dn%)&amp;Gf\$&amp;df%:pck=0

1827 CURSOR#ch,24,50:PRINT#ch,'Loading':CLS#ch,4:OVER#ch,1

Loading

1828 FOR a=1 TO 70

1829 CURSOR#ch,70+a\*2,50:PRINT#ch,'.:PAUSE 1

1830 INPUT#99,astro\$(a):FOR b=1 TO 11:INPUT#99,astro(a,b):END FOR b

1831 END FOR a

1832 FOR a=1 TO 2:FOR b=1 TO 5:INPUT#99,side(a,b):END FOR b:END FOR a

1833 INPUT#99,GTS\TD2\Gp1\Gp2\AInt\RIInt:CLOSE#99

1834 OVER#ch,0:AT#ch,5,0:CLS#ch,3:nck=0:G**C**hoice1835 END **DEF**ine1837 **DEFine PROCEDURE GSave**1838 ch=13:SelPath:IF pck=0:BLOCK#ch,250,10,0,50,0:RETu**rn** :ELSE F**C**heck

1839 IF eck=1

1840 CURSOR#ch,24,50:PRINT#ch,'DEVICE ERROR':CLS#ch,4

DEVICE ERROR

1841 PAUSE 30:CURSOR#ch,24,50:CLS#ch,4:eck=0:RETu**rn**

1842 END IF

1843 IF pck=1

1844 CURSOR#ch,24,50:PRINT#ch,'Overwrite Y/N':CLS#ch,4:PAUSE

Overwrite Y/N

1845 IF KEYROW(5)<>64:CURSOR#ch,24,50:CLS#ch,4:RETu**rn**

1846 END IF

1847 DELETE Drv\$(dn%)&amp;Gf\$&amp;df%:pck=0

1848 CURSOR#ch,24,50:PRINT#ch,'Saving':CLS#ch,4:OVER#ch,1

Saving

1849 OPEN \_NEW#99,Drv\$(dn%)&amp;Gf\$&amp;df%

1850 FOR a=1 TO 70

1851 CURSOR#ch,70+a\*2,50:PRINT#ch,'.:PAUSE 1

1852 PRINT#99,astro\$(a):FOR b=1 TO 11:PRINT#99,astro(a,b):END FOR b

1853 END FOR a

1854 FOR a=1 TO 2:FOR b=1 TO 5:PRINT#99,side(a,b):END FOR b:END FOR b

1855 PRINT#99,GTS\TD2\Gp1\Gp2\AInt\RIInt:CLOSE#99

1856 OVER#ch,0:AT#ch,5,0:CLS#ch,3

1857 END **DEF**ine1859 **DEFine PROCEDURE FCheck**

1860 ch=13:CURSOR#ch,24,50:PRINT#ch,'Searching...':CLS#ch,4

Searching...

1861 PAUSE 20:DELETE Drv\$(dn%)&amp;'FList'

1862 OPEN \_NEW#99,Drv\$(dn%)&amp;'FList':DIR#99,Drv\$(dn%):CLOSE#99

1863 OPEN \_IN#99,Drv\$(dn%)&amp;'FList'

1864 **REPEAT** Dir\_lp1865 IF EOF(#99):CLOSE#99:BLOCK#ch,250,10,0,50,0:pck=0:**EXIT** Dir\_lp1866 INPUT#99,Fchk\$:IF Fchk\$=GFile\$:CLOSE#99:pck=1:**EXIT** Dir\_lp1867 **UNTIL** **REPEAT** Dir\_lp1868 END **DEF**ine

# 1870 DEFINE PROCEDURE SelPath

```

1871 ch=13:eck=0:INK#ch,4:BLOCK#ch,12,3,224,54,4 :Gf$='AD2375Data_'
1872 CURSOR#ch, 24,50:PRINT#ch,'Select: ↑ ↓';Drv$(dn%)&Gf$&df%
1873 CURSOR#ch,190,50:PRINT#ch,' ← → ←':BLOCK#ch,2,4,214,52,4:INK#ch,7
1874 REPEAT Path_lp
1875 CURSOR#ch, 84,50:PRINT#ch,Drv$(dn%)
1876 CURSOR#ch,180,50:PRINT#ch,df% :k=CODE(INKEY$(5))
1877 SELECT ON k
1878 =192:df%=df%-1:IF df%<0:df%=9
1879 =200:df%=df%+1:IF df%>9:df%=0
1880 =208:dn%=dn%+1:IF dn%>dm%:dn%=0
1881 =216:dn%=dn%-1:IF dn%<1:dn%=dm%
1882 = 10:pck=1:GFile$=Gf$&df%:EXIT Path_lp
1883 = 32:pck=0:EXIT Path_lp
1884 END SELECT
1885 END REPEAT Path_lp
1886 END DEFINE

```

# 1888 REMARK Game Init Intro & Notes



# 1890 DEFINE PROCEDURE Init\_Win

```

1891 OPEN#20,scr_:WINDOW#20,512,256,gx,gy:BORDER#20,1,3:PAPER#20,0:CLS#20
1892 WINDOW#1,496,220,8+gx,10+gy:PAPER#1,0:CLS#1:SCALE#1,120,0,0
1893 WINDOW#2,500,226,6+gx,4+gy :PAPER#2,0:SCALE#2,100,0,0:INK#2,127:CLS#2
1894 GTitle 2,138,0,'QBITS GALAXY AD2375':GIntro:INK#2,248
1895 FILL#2,1:LINE#2,31,37 TO 133,2,37 TO 127,31 TO 37,31 TO 31,37:FILL#2,0
1896 WINDOW#1,312,122,100+gx,24+gy:BORDER#1,1,7
1897 WINDOW#1,304,118,104+gx,26+gy:SCALE#1,100,0,0:CLS#1
1898 GTitle 2,158,0,'GALAXY AD2375':Emblem:RESTORE 1900
1899 FOR ch=3 TO 18:OPEN#ch,scr_:READ a,b,c,d:WINDOW#ch,a,b,c+gx,d+gy
1900 DATA 132, 46,104,26, 90, 56,211,26, 132,46,276, 26 :REMark win 3-5
1901 DATA 132, 40,104,62, 132, 40,276,62 :REMark win 6-7
1902 DATA 132, 50,104,94, 90, 62,211,82, 132,50,276, 94 :REMark win 8-10
1903 DATA 90,100, 8,28, 90,100,414,28, 274,66,120,160 :REMark win 11-13
1904 DATA 78, 22, 14, 4, 78, 22,420, 4, 198,13,154,146 :REMark win 14-16
1905 DATA 100, 62,18,158, 100, 62, 396,158 :REMark win 17-18
1906 INK#14,7:INK#15,7:INK#16,7 :SCALE#17,70,0,0:SCALE#18,70,0,0
1907 WINDOW#0,500,24,6+gx,230+gy:Init_Stats:DPS 11:DPS 12
1908 sn=20:astro$(sn)='Sol':astro$(sn,10)=228:astro$(sn,11)=2
1909 BORDER#13,1,7:BORDER#16,1,2:StarView:StarFleet
1910 END DEFINE

```

# 1912 DEFINE PROCEDURE GTitle(ch,x,y,str\$)

```

1913 OVER#ch,1:CSIZE#ch,2,1
1914 INK#ch,2:FOR a=0 TO 1:CURSOR#ch,x+a,y :PRINT#ch,str$
1915 INK#ch,6:FOR a=2 TO 3:CURSOR#ch,x+a,y+1:PRINT#ch,str$
1916 OVER#ch,0:CSIZE#ch,0,0
1917 END DEFINE

```

# 1919 DEFINE PROCEDURE Emblem

```
1920 x=82:y=96:INK#2,6:CIRCLE#2,x,y,4:CIRCLE#2,x,y,3,2
1921 FILL#2,1:LINE#2,x-2.5,y TO x-1.2,y+2 TO x+1.2,y+2 TO x+2.5,y
1922 LINE#2 TO x+1.2,y-1.9 TO x-1.2,y-1.9 TO x-2.5,y:FILL#2,0
1923 INK#2,0:FILL#2,1:CIRCLE#2,x,y,1:FILL#2,0:LINE#2,x-2.5,y TO x+2.5,y
1924 LINE#2,x+1.8,y+2 TO x-2,y-2.2:LINE#2,x-1.8,y+2 TO x+2,y-2.4
1925 END DEFINE
```



# 1927 DEFINE PROCEDURE GIcon

```
1928 ch=20:x=74:y=7:IF SPck=0:INK#ch,6:ELSE INK#ch,2
1929 LINE#ch,x-35,y TO x,y+.4 TO x+35,y TO x,y-.4 TO x-35,y
1930 FILL#ch,1:CIRCLE#ch,x,y,3,.3,PI/2:FILL#ch,0
1931 END DEFINE
```



# 1933 DEFINE PROCEDURE Init\_Stats

```
1934 CLS#17:INK#17,5:CUSOR#17,0,2:CLS#18:INK#18,3:CUSOR#18,0,2
1935 PRINT#17,'WMD' :PRINT#18,' WMD'
1936 QBold 17,7,25,2,'ALLIANCE' :QBold 18,7,1,2,'REPUBLIC'
1937 PRINT#17,' :Land Based' :PRINT#18,' Land Based:'
1938 PRINT#17,' : ' :PRINT#18,' : '
1939 PRINT#17,' : ' :PRINT#18,' : '
1940 PRINT#17,'Fuel:' :PRINT#18,' :Fuel'
1941 PRINT#17,'Star Systems:' :PRINT#18,' :Star Systems'
1942 Alliance:Republic:SFuel:IF Gp1=1:ch=17:x=25:ELSE ch=18:x=5
1943 LINE#ch,x,69 TO x+50,69 TO x+50,57 TO x,57 TO x,69
1944 ASS 17,7,45,36:AF2 17,22,28:INK#18,7:RSS 18,7,40,34:RF2 18,62,28
1945 LINE#2,2,0 TO 2,29:ARC#2 TO 4,33,-PI/2:LINE#2 TO 18,33 TO 22,36
1946 LINE#2,162,0 TO 162,29:ARC#2 TO 160,33,PI/2:LINE#2 TO 146,33 TO 142,36
1947 END DEFINE
```



# 1949 DEFINE PROCEDURE Alliance

```
1950 ch=2:x=25.1:y=38:INK#ch,7
1951 FILL#ch,1:ARC#ch,x-2.8,y TO x+2.8,y,-PI/1.2:FILL#ch,0
1952 LINE#ch,x+2.8,y TO x+2.5,y-3:LINE#ch,x-2.8,y TO x-2.5,y-3
1953 INK#ch,0:LINE#ch,x-.4,y+.8 TO x-.4,y+2.8:LINE#ch,x+.4,y+.8 TO x+.4,y+2.8
1954 INK#ch,7:LINE#ch,x-.2,y+.8 TO x-.2,y+2.8:LINE#ch,x+.2,y+.8 TO x+.2,y+2.8
1955 ARC#ch,x-2,y-.5 TO x-2,y-.5,PI:LINE#ch TO x-2,y-.5
1956 ARC#ch,x+.2,y-.5 TO x+2,y-.5,PI:LINE#ch TO x,y-.5
1957 CIRCLE#ch,x,y-3.2,1.2,.6,PI:LINE#ch,x-2.5,y-3 TO x,y-4 TO x+2.5,y-3
1958 END DEFINE
```



# 1960 DEFINE PROCEDURE Republic

```
1961 ch=2:x=138.6:y=38:INK#ch,7
1962 FILL#ch,1:ARC#ch,x-2.5,y TO x+2.5,y,-PI/1.5:LINE#ch TO x+3,y-3.6
1963 LINE#ch TO x+1.4,y-5 TO x-1.4,y-5 TO x-3,y-3.6 TO x-2.5,y:FILL#ch,0
1964 INK#ch,0:LINE#ch, x-2,y-3.8 TO x,y-3 TO x+2,y-3.8
1965 FILL#ch,1:ARC#ch,x-2,y-.5 TO x-.2,y-.5,PI:LINE#ch TO x-2,y-.5:FILL#ch,0
1966 FILL#ch,1:ARC#ch,x+.2,y-.5 TO x+2,y-.5,PI:LINE#ch TO x-2,y-.5:FILL#ch,0
1967 FILL#ch,1:LINE#ch,x-2,y-5 TO x,y-4 TO x+2,y-5 TO x-2,y-5:FILL#ch,0:INK#2,7
1968 END DEFINE
```

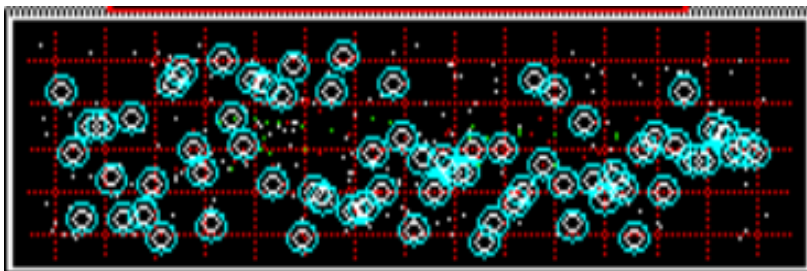


# 1970 DEFINE PROCEDURE StarFleet

```
1971 ASS 1,7, 44,38:AF1 1, 20,46:AF2 1, 44,58:AF3 1, 66,46
1972 RSS 1,7,148,36:RF1 1,124,46:RF2 1,148,56:RF3 1,170,46
1973 END DEFINE
```



1975 DATA	'Vega'	'Bellatrix'	'Zibal'	'Castor'	'Naos'	'Artos'	'Ilad'
1976 DATA	'Tabit'	'Diphda'	'Elnath'	'Acrux'	'Errai'	'Cellus'	'Detrus'
1977 DATA	'Polaris'	'Homan'	'Rigel'	'Sabik'	'Sarin'	'Westron'	'Zenda'
1978 DATA	'Furud'	'Gienah'	'Hadar'	'Propus'	'Izar'	'Centra'	'Nebtron'
1979 DATA	'Enif'	'Algo'	'Mirzarm'	'Libeta'	'Atria'	'Panus'	'Ulan'
1980 DATA	'Keid'	'Kochab'	'Lesath'	'Ankaa'	'Marsic'	'Nebus'	'Alean'
1981 DATA	'Meissa'	'Subra'	'Menkib'	'Altair'	'Muscida'	'Pusan'	'Curson'
1982 DATA	'Cursa'	'Nashira'	'Nunki'	'Ogma'	'Pollux'	'Xilon'	'Scarb'
1983 DATA	'Sirius'	'Merack'	'Serena'	'Denab'	'Tureis'	'Noran'	'Theron'
1984 DATA	'Beid'	'Wesen'	'Yildun'	'Caph'	'Zooma'	'Jarron'	'Cetrus'



1986 **DEFINE PROCEDURE Init\_New**

**Note:** Set Info for 70 Star Systems

1987 ch=13:Ac=0:Rc=0:INK#ch,7:**RESTORE 1975**

1988 **CURSOR#ch,18,50:CLS#ch,4:PRINT#ch,'Initilising':OVER#ch,1**

1989 **FOR n=1 TO 70**

1990 **READ astro\$(n):CURSOR#ch,92+n\*2,50:PRINT#ch,'':PAUSE 1**

1991 Popu=RND(4 TO 16)/RND(2 TO 8) :REMark Population 0.5<<8bn

1992 astro(n,1)=INT(Popu\*2.5) :REMark Food max 20

1993 astro(n,2)=INT(Popu+2) :REMark Ores max 10

1994 astro(n,3)=INT(Popu\*1.5) :REMark Tec max 12

1995 astro(n,4)=INT((Popu+astro(n,2))/2) :REMark Trade max 20

1996 astro(n,5)=INT((Popu+astro(n,3))/2) :REMark WMD max 20

1997 astro(n,6)=INT(astro(n,2)/3) :REMark Fuel Cells

1998 astro(n,7)=0 :REMark Side 0=Independent 1=Alliance 2=Republic

1999 astro(n,8)=n\*4+15 :REMark x Grid coordinate

2000 astro(n,9)=RND(10 TO 90) :REMark y Grid coordinate

2001 astro(n,10)=RND(24 TO 46) :REMark Planet Colour

2002 astro(n,11)=RND(1 TO 3) :REMark Solar System

2003 astro(n,12)=Popu :REMark Population max 10bn

2004 **END FOR n**

2005 n=RND( 5 TO 65):astro(n,7)=1:astro(n,5)=12:astro(n,6)=10

2006 n=RND(35 TO 45):astro(n,7)=2:astro(n,5)=12:astro(n,6)=10

2007 **REPEAT Count\_Ip**

2008 n=RND(20 TO 50):IF Rc<9 AND astro(n,7)=0:astro(n,7)=2:Rc=Rc+1

2009 n=RND( 5 TO 30):IF Ac<9 AND astro(n,7)=0:astro(n,7)=1:Ac=Ac+1

2010 n=RND(40 TO 65):IF Ac<9 AND astro(n,7)=0:astro(n,7)=1:Ac=Ac+1

2011 IF Ac=9 AND Rc=9:**EXIT Count\_Ip**

2012 **END REPEAT Count\_Ip**

2013 **FOR n=1 TO 70** :IF astro(n,7)>0 AND astro(n,5)<3:astro(n,5)=3

2014 **FOR s=1 TO 2** :side(s,1)=4:side(s,2)=2:side(s,3)=6:side(s,4)=6

2015 **OVER#ch,0:Alnt=12:RInt=12:sch=0:GMapInit:LMap=GTS=0:nck=0**

2016 **END DEFINE**



2018 **DEFine PROCEDURE GIntro**

```
2019 FOR Stars=1 TO 250:INK RND(2 TO 7):POINT RND(0 TO 280),RND(0 TO 140)
2020 CSIZE 0,0:RESTORE 2025:ch=1:FOR a=1 TO 12:READ i,c,r,str$:MPrn :PAUSE 5
2021 ASS 1,7,40,72:RSS 1,7,160,70:PAUSE
2022 FOR a=1 TO 20:BLOCK 12+a*20,200,240-a*10,20,0:PAUSE 2:END FOR a:CLS#2
2023 END DEFine
```



```
2025 DATA 5,110, 30,'There is Disquiet in the Galaxy as the Republic'
2026 DATA 5,126, 40,'First Order seeks to enforce new Taxes on'
2027 DATA 5,156, 50,'Interstellar Travel and Trade.'
2028 DATA 3,158, 74,'(M)arshal (G)eneral (C)aptain'
2029 DATA 6,166, 86,'Choose Skill Level and Side'
2030 DATA 3,173, 98,'(A)lliance or (R)epublic'
2031 DATA 5, 96,122,'Select Star System and <HyperSpace Jump> to Planet'
2032 DATA 5, 96,132,'Attack (<?) Annex for WMD/Trade/Tech/Ores/Food/Fuel'
2033 DATA 5,116,142,'Harvest Results and Reassign to Defend your'
2034 DATA 5,132,152,'Star Sytems or use to Attack the Enemy'
2035 DATA 3,186,170,'(G)ame Nates (M)enu',7,206,192,'Press Any key'
2036 DATA 2, 26, 70,'(F1) Simulation Mode (F2) Graphics Check'
2037 DATA 2, 26, 80,'(F3) Timeslip Check - Swap Sides/Assets'
2038 DATA 2, 26, 90,'(F4) Enemy Encounter (F5) Tech/WMD Cheat'
2039 DATA 5, 50,104,'(D)efault - Highscore - Reset(d)'
```

2041 **DEFine PROCEDURE GNotes**

```
2042 ch=1:INK 6:CURSOR 76,4:PRINT 'GALAXY AD2375 Game Notes':INK 4
2043 RESTORE 2031:FOR a=1 TO 5:READ i,c,r,str$:c=c-96:r=6+a*10:MPrn
2044 RESTORE 2036:FOR a=1 TO 4:READ i,c,r,str$:MPrn
2045 INK 7:OVER 1:CURSOR 96,56:PRINT 'G M':OVER 0
2046 PAUSE 500:CLS:FOR i=1 TO 120:POINT RND(10 TO 200),RND(5 TO 95)
2047 END DEFine
```

2049 **DEFine PROCEDURE MPrn**:INK i:CURSOR#ch,c,r:PRINT#ch,str\$:**END DEFine**

2051 **DEFine PROCEDURE GameTest**

```
2052 Pulsar:Binary:Nebula:Blackhole:TimeSlip:HJump:StarView
2053 SPck=1:EOrbit:RoIDice:SSInfo:LOrbit:Encounter
2054 FOR n=1 TO 70:astro(n,5)=RND(5 TO 12):astro(n,7)=Gp1
2055 LMap:SSStats:ch=1:nck=1:SPck=0:GEnd:PAUSE
2056 END DEFine
```