

spacewar 4.8 7/24/63 pt. 2 dfw

```
nob=30          /total number of colliding objects

m10,          setup mtc, 5000    /delay for loop
              init m11, mtb    /loc of calc routines
              init mx1, nx1     /x
              init my1, ny1     /y
              init ma1, na1     /count for length of explosion or torp
              init mb1, nb1     /time taken by calc routine
              init mdx, ndx     /dx
              init mdy, ndy     /dy
              init mom, nom     /angular velocity
              init mth, nth     /angle
              init mfu, nfu     /fuel
              init mtr, ntr     /number torps remaining
              init mot, not     /outline of spaceship
              init mco, nco     /old control word
              law nh1
              dac mh1
              law nh2
              dac mh2
              law nh3
              dac mh3
              law nh4
              dac mh4
              lac mtb
              sza i
              jmp mdn
              lac mtb 1
              sza i
              jmp mdn          /ship all gone away
              lac ntr          /test if both ships out of torps
              ior ntr 1
              sza i
              jmp mdn
              jmp m11
```

/ control word get routines

```
mg1,      dap mg3
          cli
          iot 11
          rir 4s
mg3,      jmp .
mg2,      dap mg4
          lat
          swap
mg4,      jmp .
idl,      idx mth
          idx mfu
          idx mtr
          idx mco
          idx mot
          idx mom
          idx mh1
          idx mh2
          idx mh3
          idx mh4
ids,      idx mx1
          idx my1
          idx ma1
          idx mb1
          idx mdy
          idx mdx
ml1,      lac .
          sza i
          jmp mq1
          dap mjm
          spa
          jmp mjm

          law 1
          add ml1
          sad (lac mtb nob
          jmp mjm-1
          dap ml2
          law 1
          add mx1
          dap mx2
          law 1
          add my1
          dap my2
          law 1
          add ma1
          dap ma2
          law 1
          add mb1
          dap mb2

          / 1st control word
          / zero if not active
          / not active
          /jmp to calc routine or make explode
          /control word + if object collidible

          /proximity test
```

```

ml2,      lac .           / 2nd control word
          spq             / can it collide?
          jmp mq2         / no
mx1,      lac .           / calc if collision
mx2,      sub .           / delta x
          spa             / take abs val
          cma
          dac  $\bar{t}1$ 
          sub me1         / < epsilon ?
          sma
          jmp mq2         / no
my1,      lac .
my2,      sub .
          spa
          cma
          sub me1         / < epsilon ?
          sma
          jmp mq2         / no
          add  $\bar{t}1$ 
          sub me2
          spa
          jmp mjm
mq2,      idx mx2         / end of comparison loop
          idx my2
          idx ma2
          idx mb2
          index ml2, (lac mtb nob, ml2

          idx mjm
mjm,      jmp .           /to calc routine or make object explode
mb1,      lac .           / alter count of number of instructions
          add  $\underline{m}tc$ 
          dac  $\underline{m}tc$ 
mq1,      idx ml1
          sad (lac mtb 1
          jmp idl
          sas (lac mtb nob
          jmp ids
          background
          jsp blp         / display massive star
          count  $\underline{m}tc$ , . / use up rest of time of main loop
          jmp ml0         / repeat whole works

```

```

blt,      law 20          /routine to set explosion
          dac i mb1
          dac i mb2

sex,      lac (mex 400000 /alternate entry point
          dac i ml1      / replace calc routine with explosion
          dac i ml2
          lac i mb1      / duration of explosion
mb2,      add .
          cma
          sar 8s
          add (1
ma1,      dac .
ma2,      dac .
          jmp mb1

```

/ misc calculation routines

 / explosion

```

mex,      lac i mdx
          sar 3s
          add i mx1
          dac i mx1
          lac i mdy
          sar 3s
          add i my1
          dac i my1
          law mst
          dap msh
          lac i mb1      / time involved
          cma cli-opr
          sar 3s
          dac t1
          sub (140
          sma
          idx msh
mz1,      lac ran
          and (777
          ior (scl
          dac ml1
          random
          scr 9s
          sir 9s
msh,      xct .
mi1,      hlt
          add i my1
          swap
          add i mx1
          dpy-i 300
          count t1, mz1
          count i ma1, mb1
          dzm i ml1
          jmp mb1

mst,      scr 1s
          scr 3s

```

/ torpedo calc routine

```
tc1,      jmp blt
          count i ma1, tc1
          lac (mex 400000
          dac i ml1
          law i 2
          dac i ma1
          law 20
          dac i mb1
          jmp mb1

tc1,      lac i mx1
          sar 9s
          xct the
mdy,      add ndy
          dac i mdy
          sar 3s
          add i my1
          dac i my1
          sar 9s
          xct the
mdx,      add ndx
          dac i mdx
          sar 3s
          add i mx1
          dac i mx1
          dispt i, i my1, 1
          jmp mb1
```

/ hyperspace routines

/ this routine handles a non-colliding ship invisibly
/ in hyperspace

```
hp1,      count i ma1, mb1
          law hp3           / next step
          dac i ml1
          law 7
          dac i mb1
          random
          scr 9s
          sir 9s
          xct hr1
          add i mx1
          dac i mx1
          swap
          add i my1
          dac i my1
          dzm i mdx
          dzm i mdy
          xct hd2
          dac i ma1
          jmp mb1
```

/ this routine handles a ship breaking out of
/ hyperspace.

```
hp3,      jmp sex
          count i ma1, hp6
          law 2000
          dac i mb1
          lac i mh4
          add hur
          dac i mh4
          random
          ior (400000
          add i mh4
          sma
          jmp po1
          lac i mh1
          dac i ml1
          lac ran
          scr 9s
          sir 9s
          xct hr2
          dac i mdy
          dio i mdx
          setup t1, 3
          lac ran
          dac i mth
hp4,      lac i mth
          sma
          sub (311040
          spa
          add (311040
          dac i mth
          count t1, hp4
          count i mh2, hp7
          dzm i mh2
hp7       xct hd3
          dac i mh3
hp6,      lac i mx1
          dispt i, i my1 2
          jmp mb1
```

```

/ spaceship calc
ss1,      jmp sex                /something came too close
          jsp i cwg
          jmp sr0

ss2,      jmp sex
          jsp i cwg
          rir 4s

sr0,      dio scw

sc1,      clf 6 cla-opr          /update angle
          spi
          add maa
          ril 1s
          spi
          sub maa
mom,      add .
          dac i mom
          szs 10
          jmp . 3
          dzm i mom
          ral 5s
          ril 1s
          spi
mfu,      stf 6
          lio nfu
          spi i
          clf 6

mth,      add .
          sma
          sub (311040
          spa
          add (311040
          dac i mth
          jda sin
          dac sn
          dzm bx
          dzm by
          szs 60
          jmp bsg
          lac i mxi
          dac t1
          mul t1
          scr 1s
          dac acx
          cla
          scr 2s
          dio fox
          lac i myi
          dac t1
          mul t1
          scr is
          dac acy

          cla
          scr 2s
          swap
          add fox
          swap
          scl 2s
          add acx
          add acy

```

```

sub str
sma i sza-skp
jmp pof
add str
varsft
dac t1
jda sqt
mul t1
undosft
scr 9s
scr 6s
szs i 20
scr 2s
sza
jmp bsg
scr 1s
dio t1
integrate mx1, bx
integrate my1, by
lac i mth
jda cos
dac cs
sar 9s
xct sac
szf i 6
cla
add by
diff mdy, my1, (sar 3s
lac sn
sar 9s
xct sac
cma
szf i 6
cla
add bx
diff mdx, mx1, (sar 3s
scale sn, 5s, ssn
scale cs, 5s, scn
lac i mx1
sub ssn
dac sx1
sub ssn
dac stx

```

/ switch 2 for light star

bsg,


```

lac i my1
add scn
dac sy1
add scn
dac sty
scale sn, 9s, ssn
scale cs, 9s, scn
dac scm
lac ssn
dac ssm
add scn
dac ssc
dac ssd
lac ssn
sub scn
dac csn
cma
dac csm
cla cli-opr
dpy-4000
mot,sp5, jmp i .
sq6,     szf i 6
        jmp sq9 /not blasting or no fuel
        rancr sar 9s, sar 4s, src
        scale sn, 8s, ssn
        scale cs, 8s, scn
sq7,     count i mfu, st2
        dzm i mfu
        jmp sq9
st2      yincr sx1 sy1, sub
        dispt i, sy1
        count src,sq7
sq9,     count i ma1, sr5 / check if torp tube reloaded
mco,     dzm i ma1 / prevent count around
        lac . / previous control word
        cma
        szs i 30
        cle
        and scw / present control word
        ral 3s / torpedo bit to bit 0
        sma
        jmp sr5 / no launch
mtr,     count ntr, st1 / check if torpedos exhausted
        dzm i mtr / prevent count around
        jmp sr5
st1,     init sr1, mtb nob-1 /search for unused object
sr1,     lac .
        sza i / 0 if unused
        jmp sr2
        law i 1
        add sr1
        dap sr1
        sas (lac mtb-1
        jmp sr1
        hlt / no space for new objects
        jmp sr5 /go on anyway

```

```

sr2,      lac (tcr                               / set up torpedo calc
          dac i sr1
          law nob
          add sr1
          dap ss3
          lio stx
ss3,      dio .
          add (nob
          dap ss4
          lio sty
ss4       dio .
          add (nob
          dap sr6
          add (nob
          dap sr7
          add (nob
          dap sr3
          add (nob
          dap sr4
          lac sn
          xct tvl
          cma
          add i mdx
sr3,      dac .
          lac cs
          xct tvl
          add i mdy
sr4,      dac .
          xct rlt
          dac i ma1                               / permit torp tubes to cool
          xct tlf                                 / life of torpedo
sr6,      dac .
          lac (lac mtb nob-1
          sub sr1
          sal 3s
          add (30
sr7,      dap .                                   / length of torp calc.
sr5,      lac scw
          dac i mco
          count i mh3, mb1
          dzm i mh3
          lac mh2
          sza i
          jmp mb1
          lac scw
          spa
          ral 1s
          sma                                     /hyperspace button on?
          jmp mb1                                 /no
          lac i m11
          dac i mh1
          lac (hp1 400000
          dac i m11
          xct hd1
          dac i ma1
          law 2
          dac i mb1
          jmp mb1

```

/ here to handle spaceships dragged into star

/ spaceship in star

```
pof      dzm i mdx
         dzm i mdy
         szs 50
         jmp po1
         lac (377777
         dac i mx1
         dac i my1
         jmp mq1
```

```
po1,    lac (mex 400000 / now go bang
         dac i ml1
         law i 10
         dac i ma1
         jmp mq1
```

/ here if a ship exploded or both ships out of torps

```
mdn,    count ntd, ml1 /wait awhile
         stf 1
         stf 2
         law ss1
         xor mtb
         sza
         clf 1
         sza i
         idx 1sc
         law ss2
         xor mtb 1
         sza
         clf 2
         sza i
         idx 2sc
         clf 2
         jmp a
```

```

a1,      law mg2          / test word control
         dac cwg
         jmp a

a40,     law cwr / here from start at 4
         dac cwg
         jmp a6

a,       lac gct
         sma
         jmp a5
         count gct, a5
         lac isc
         sas sc
         jmp a4
         law i 1
         dac gct

a5,      lat
         and (40
         sza i
         jmp a2

a4,      lac isc
         lio sc
         hlt
         lat
         and (40
         sza
         jmp a2
         dzm isc
         dzm sc

a6,      lat
         rar 6s
         and (37
         sza
         cma

a2,      dac gct
         clear mtb, nnn-1 / clear out all tables
         law ss1
         dac mtb
         law ss2
         dac mtb 1
         lac (200000
         dac nx1
         dac ny1
         cma
         dac nx1 1
         dac ny1 1
         lac (144420
         dac nth

```

```

law nnn          / start of outline program
dac not
lio ddd
spi i
jmp a3
jda oc          / compile outline
ot1
a3, dac not 1
    jda oc
    ot2
    xct tno
    dac ntr
    dac ntr 1
    lac foo
    dac nfu
    dac nfu+1
    law 2000
    dac nb1
    dac nb1 1
    xct mhs
    dac nh2
    dac nh2 1
    xct tlf
    sal 1s
    dac ntd     / restart delay is 2xtorp life
    jmp ml0     /start new game

```

/ outlines of spaceships

```

ot1, 111131
      111111
      111111
      111163
      311111
      146111
      111114
      700000

```

5/

```

ot2, 013113
      113111
      116313
      131111
      161151
      111633
      365114
      700000

```

. 5/

```

variables
constants

```

mtb,

/ table of objects and their properties

nx1=mtb nob
ny1=nx1 nob
na1=ny1 nob
nb1=na1 nob
ndx=nb1 nob
ndy=ndx nob
nom=ndy nob
nth=nom 2
nfu=nth 2
ntr=nfu 2
not=ntr 2
nco=not 2
nh1=nco 2
nh2=nh1 2
nh3=nh2 2
nh4=nh3 2
nnn=nh4 2

start 4