

spacewar 4.3 5/17/63 ddp . pt 1

```
3/      jmp sbf      / ignore seq. break
      jmp a40
      jmp a1      / use test word for control, not iot 11 co
```

/ interesting and often changed constants

/symb loc usual value (all instructions are executed,
/ and may be replaced by jda or jsp)

```
tno, 6, law i 41      / number of torps + 1
tv1, 7, sar 4s      / torpedo velocity
rlt, 10, law i 20      / torpedo reload time
tlf, 11, law i 140      / torpedo life
foo, 12, -20000      / fuel supply
maa, 13, 10      / spaceship angular acceleration
sac, 14, sar 4s      / spaceship acceleration
str, 15, 100      / star capture radius
me1, 16, 6000      / collision "radius"
me2, 17, 3000      / above/2
ddd, 20, -0      / 0 to save space for ddt
the, 21, sar 9s      / amount of torpedo space warpage
mhs, 22, law i 10      / number of hyperspace shots
hd1, 23, law i 40      / time in hyperspace before breakout
hd2, 24, law i 100      / time in hyperspace breakout
hd3, 25, law i 200      / time to recharge hyperfield generators
hr1, 26, scl 9s      / scale on hyperspatial displacement
hr2, 27, scl 4s      / scale on hyperspatially induced velocity
hur, 30, 40000      / hyperspatial uncertainty
ran, 31, 0      / random number
grv, 32, sar 6s      / gravitational constant
```

/ place to build a private control word routine.
/ it should leave the control word in the io as follows.
/ high order 4 bits, rotate ccw, rotate cw, (both mean hyperspace)
/ fire rocket, and fire torpedo. Low order 4 bits, same for
/ other ship. Routine is entered by jsp cwg.

40/

```
cwr,      jmp mg1      / normally iot 11 control
. 20/      / space
```

2 / routine to flush sequence breakes, if they occur.

```
sbf,      tyi
          lio 2
          lac 0
          lsm
          jmp i 1
```

```
          define
xincr X,Y,INS
          lac Y
          INS ssn
          dac Y
          lac X
          INS scn
          dac X
          term
```

```
          define
yincr X,Y,INS
          lac Y
          INS scn
          dac Y
          lac X
          -INS+add+sub ssn
          dac X
          terminate
```

```
          define
dispatch
          add (. 3
          dap . 1
          jmp .
          term
```

```
          define
dispt A,Y,B
          repeat 6  B=B+B
          lio Y
          szs 20
          jda kcb
          dpy-A+B
          term
```

```
          define
scale A,B,C
          lac A
          sar B
          dac C
          term
```

```
define
diff V,S,SF
    add i V
    dac i V
    xct SF
    add i S
    dac i S
    term
```

```
define
random
    lac ran
    rar 1s
    xor (355670
    add (355670
    dac ran
    term
```

```
define
ranct S,SS,C
    random
    S
    SS
    sna
    cna
    dac C
    terminate
```

define

varsft
dzm xys
dac t1
idx xys
v2, idx xys
lac t1
scr 2s
dac t1
sza
jmp v2+R
scr 2s
swap
terminate

define

undosft
dac t1
dio t2
lac xys
add sft
dap .+1
lac .
dac .+6
dac .+6
xor (10000 / change scr to scl or scl to scr.
dac xyt
lac t1
dio t2
scr .
scr .
terminate

define

integrate A,B
cli
lac i A
scr 9s
scr 1s
div t1
hlt
cma+cli-opr
xct xyt
xct grv
dac B
terminate

sft,

lac .-1
scr 7s
scr 6s
scr 5s
scr 4s
scr 3s
scr 2s
scr 1s
scr
scl 1s

/sine-cosine subroutine•Adams associates
 /calling sequence= number in AC, jda sin or jdacos.
 /argument is between +2 pi, with binary point to right of bit 3.
 /answer has binary point to right of bit 0. Time = 2.35-? ms.
 /changed for auto-multiply , ddp 1/19/63

```

cos,      0
          dap csx
          lac (62210
          add cos
          dac sin
          jmp .+4

sin,      0
          dap csx
          lac sin
          spa
s11,      add (311040
          sub (62210
          sma
          jmp si2
          add (62210

s13,      ral 2s
          mul (242763
          dac sin
          mul sin
          dac cos
          mul (756103
          add (121312
          mul cos
          add (532511
          mul cos
          add (144417
          mul sin
          scl 3s
          dac cos
          xor sin
          sma
          jmp csx-1
          lac (377777
          lio sin
          spi
          cma
          jmp csx

          lac cos
csx,      jmp .

s12,      cma
          add (62210
          sma
          jmp si3
          add (62210
          spa
          jmp .+3
          sub (62210
          jmp si3

          sub (62210
  
```

/integer square root
/input in ac, binary point to right of bit 17, jda sqt
/answer in ac with binary point between bits 8 and 9
/largest input number = 177777

```
sqt,      0  
          dap sqx  
          law i 23  
          dac sq1  
          dzm sq2  
          lic sqt  
          dzm sqt  
  
sq3,      isp sq1  
          jmp .+3  
          lac sq2  
sqx,      jmp .  
  
          lac sq2  
          sal 1s  
          dac sq2  
          lac sqt  
          rcl 2s  
          sza i  
          jmp sq3  
          dac sqt  
          lac sq2  
          sal 1s  
          add (1  
          sub sqt  
          sma+sza-skp  
          jmp sq3  
          spe  
          cma  
          dac sqt  
          idx sq2  
          jmp sq3  
  
sq1,      0  
sq2,      0
```

/outline compiler
/ac=where to compile to, call jda oc
/ot=address of outline table

define

plinst A
lac A
dac i oc
idx oc
terminate

define

comtab A, B
plinst A
jsp ocs
lac B
jmp oce
terminate

ocs, dap ocz /puts in swap

dio i oc
idx oc
dio i oc
idx oc

ocz, jmp .

oc, 0 /outline compiler proper

dap ocx
lac i ocx
dap ocg
plinst (stf 5
dap ocm
idx ocx

ock, plinst (lac $\overline{sx}1$
plinst (lio $\overline{sy}1$
clf 6

ocj, setup $\overline{occ}, 6$

ocg, lio . . /outline table

och, cla
rcl $\underline{3s}$
dio oci
lio (rcl $\underline{9s}$
dispatch

opr
jmp oc1
jmp oc2
jmp oc3
jmp oc4
jmp oc5
jmp oc6

oco, jmp oc2

ocq, jmp oc3

ocp, jmp oc4

ocr, jmp oc5

jmp oc6

```

plinst (szf 5      /7 code
add (4
dap ocn
plinst ocn
plinst (dac  $\overline{sx1}$ 
plinst (dio  $\overline{sy1}$ 
plinst (jmp  $\overline{sq6}$ 
plinst (clf  $\overline{5}$ 
plinst (lac  $\overline{scn}$ 
plinst (cma
plinst (dac  $\overline{scn}$ 
plinst (lac  $\overline{ssm}$ 
plinst (cma
plinst (dac  $\overline{ssm}$ 
plinst (lac  $\overline{csn}$ 
plinst (lio  $\overline{ssd}$ 
plinst (dac  $\overline{ssd}$ 
plinst (dio  $\overline{csn}$ 

plinst (lac  $\overline{ssc}$ 
plinst (lio  $\overline{csn}$ 
plinst (dac  $\overline{csn}$ 
plinst (dio  $\overline{ssc}$ 
plinst ocm
ocx,    jmp .
ocm,    jmp .
ocn,    jmp .

oc1,    plinst (add  $\overline{ssn}$ 
        jsp ocs
        lac (sub  $\overline{scn}$ 
oce,    dac i oc
        idx oc
        jsp ocs
        plinst (ioh
        lac (dpy-4000
ocd,    dac i oc
        idx oc
        lio  $\overline{oci}$ 
        count  $\overline{occe}$ , ocn
        idx ocg
        jmp ocj

oc2,    comtab (add  $\overline{scm}$ , (add  $\overline{ssm}$ 
oc3,    comtab (add  $\overline{ssc}$ , (sub  $\overline{csn}$ 
oc4,    comtab (sub  $\overline{scm}$ , (sub  $\overline{ssm}$ 
oc5,    comtab (add  $\overline{csn}$ , (sub  $\overline{ssd}$ 
oc6,    szf 6
        jmp oc9
        stf 6
        plinst (dac  $\overline{ssa}$ 
        lac (dio  $\overline{ssi}$ 
        jmp ocd
oc9,    clf 6
        plinst (lac  $\overline{ssa}$ 
        lac (lio  $\overline{ssi}$ 
        jmp ocd

```

/display gravitational star

define

starp
add bx
swap
add by
swap
ioh
dpy-4000
terminate

blp, dap blx /star

szs 60
jmp blx
random
rar 9s
and (400700
spa
xor (377777
dac bx
lac ran
ral 4s
and (400700
spa
xor (377777
dac by
jsp bpt
ioh
jmp .

blx,

bpt,

dap bpx
random
sar 9s
sar 6s
spa
cma
sal 3s
add (bds
dap bjm
cla cli clf 6-opr-opr
szf i 20
jmp bjm-1
sub ny1
swap
sub nxi
dpy-4000

bjm,

bds,

bpx,

jmp .
repeat 10, starp
szf 6
jmp .
stf 6
cma
swap
cma
swap
jmp bjm

/background display • 3/13/62, pns.

```
define
dislis J, 0
    dap fx+R
    clf 5
    lac flo+R
    dap fpo+R
fs,    dap fin+R
    dap fyn+R
    idx fyn+R

fin,    lac      /lac Y
    sub fpr      /right margin
    sma
    sub (20000
    add (2000
frr,    spc
fou,    jmp fuu+R
fie,    sub (1000
    sal 8s
fyn,    lio      /lio Y
    szs 20
    jda kcb
    dpy-1
    stf 5
fid,    idx fyn+R
    sad (lio Q+2
    jmp flp+R
    sad fpo+R
    jmp fx+R
    dap fin+R
    idx fyn+R
    jmp fin+R

fuu,    szf 5
fx,     jmp      /return
    idx flo+R
    idx flo+R
    sas (Q+2
    jmp fid+R
    law J
    dac flo+R
    jmp fid+R

flp,    lac (lio J
    sad fpo+R
    jmp fx+R
    dap fin+R
    law J+1
    dap fyn+R
    jmp fin+R

fpo,    lio
flo,    J
    terminate
```

```

define
background
    jsp bck
    termin

bck,
    dap bck
    szs 40
    jmp bck
    jsp 1m
    jsp 2m
    idx bcc
    and (1
    szc
    jsp 3m
bc1,
    law 3
    and bcc
    sza 1
    jsp 4m
bc3,
    isp bkc
    jmp bey
    law i 10
    dac bkc
    law i 1
    add fpr
    spa
    add (20000
    dac fpr
bey,
    jsp 4m
bck,
    jmp

1m,
    dislis 1j, 1q
2m,
    dislis 2j, 2q
3m,
    dislis 3j, 3q
4m,
    dislis 4j, 4q

bcc,
    0
bkc,
    0
fpr,
    10000

```

mul=mul
div=div

start

spacewar 4.3 5/17/63 ddp . pt 2

/main control routine for spaceships

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

mlc,      setup mtc, 5000             /delay for loop
          init ml1, mtb               /loc of calc routines
          add (nob
          dap mxi                      / x
nx1=mtb nob
          add (nob
          dap ny1                      / y
ny1=nx1 nob
          add (nob
          dap ma1                      / count for length of explosion or torp
na1=ny1 nob
          add (nob
          dap mb1                      / count of instructions taken by calc routine
nb1=na1 nob
          add (nob
          dac mdx                      / dx
ndx=nb1 nob
          add (nob
          dac ndy                      / dy
ndy=ndx nob
          add (nob
          dap mom                      /angular velocity
nom=ndy nob
          add (2
          dap nth                      / angle
nth=nom 2
          add (2
          dac mfu                      /fuel
nfu=nth 2
          add (2
          dac ntr                      / no torps remaining
ntr=nfu 2
          add (2
          dap not                      / outline of spaceship
not=ntr 2
          add (2
          dap nco                      / old control word
nco=not 2
          add (2
          dac nh1
nh1=nco 2
          add (2
          dac nh2
nh2=nh1 2
          add (2
          dac nh3
nh3=nh2 2
          add (2
          dac nh4
nh4=nh3 2
nnn=nh4 2
```

```

law ss1
xor mtb
sza
jmp mdn
law ss2
xor mtb 1
sza
jmp mdn
law 1      / test if both ships out of torps
add ntr
spa
jmp md1
law 1
add ntr 1
spa 1
jmp mdn
md1,      xct tlf      / restart delay is 2X torpedo life
sal 1s
dac ntd
jmp m11

mdn,      count ntd,m11
stf 1
stf 2
law ss1
xor mtb
sza
clf 1
sza 1
idx 1sc
law ss2
xor mtb 1
sza
clf 2
sza 1
idx 2sc
clf 2
jmp a

```

```

a1,      law ng2          / test word control
        dac owg
        jmp a

a40,     law owr          / here from start at 4
        dac owg
        jmp a6

1sc,     0                /scores
2sc,     0

a,        dzm 8j1
        dzm 8j2
        lac gct
        sma
        jmp a5
        count gct, a5
        lac 1sc
        sas 2sc
        jmp a4
        law i 1
        dac gct
a5,      lat
        and (40
        sza 1
        jmp a2
a4,      jmp f11
        lat
        and (40
        sza
        jmp a2
        dzm 1sc
        dzm 2sc
a6,      lat
        rar 6s
        and (37
        sza
        cma
        dac gct
a2,      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 1
jmp a3
jda oc          / compile outline
ot1
a3, dac not 1
dap fi1
jda oc
ot2
dap fi2
xct tno
dac ntr
dac ntr 1
lac fco
dac nfu
dac nfu+1
law 2000
dac nb1
dac nb1 1
xct mhs
dac nh2
dac nh2 1
jmp m10
```

/ control word get routines

```
mg1,    dap mg3
        lot 1
        jsp 8a
8j1,    0
        dac T1
        lot 111
        jsp 8a
8j2,    0
        lac T1
        ril 4s
        ror 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 . 1
        jmp .
mb1,    lac .
        add mtc
        dac mtc
mq1,    idx ml1
        sad (lac mtb 1
        jmp idl
        sas (lac mtb nob
        jmp ids
        background
        jsp blp
        count mtc, .
        jmp m10
        / 1st control word
        / zero if not active
        / not active
        / alter count of number of instructions
        / display massive star
        / use up rest of time of main loop
        / repeat whole works
```

8a,	dap 8ay
	dap 8ax
	idx 8ax
	clf 7
	spi
	stf 5
	szf 1 5
	stf 6
	rll 1
	spi
	jmp 8ao
8ay,	lac
	rir 8s
	spi
	cla
	rir 1s
	spi
	law 2
	dac 1 8ay
8ad,	rir 8s
	spi
	jmp 8aq
	rir 5s
	spi
	ior (14
	rir 1s
	spi
	ior (1
	jmp 8aa
8ao	cla
	rir 9s
	spi
	law 2
	jmp 8ad
8aq,	rir 5s
	spi
	ior (1
	rir 1s
	spi
	ior (14
8aa,	rir 1s
	spi 5
	ior (4
	spi 6
	ior (10
	rir 1s
	spi 5
	ior (10
	spi 6
	ior (4
8ax,	jmp

```

col,      dap cox
          law 1
          add ml1
          sad (lac mth nob
          jmp cox-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
ml2,      lac .
          spq      / 2nd control word
          jmp mq2   / can it collide?
mx1,      lac .
mx2,      sub .     / no
          spa      / calc if collision
          cma      / delta x
          dac mt1   / take abs val
          sub me1   / < EPSILON ?
          sma
          jmp mq2   / no
my1,      lac .
my2,      sub .
          spa
          cma
          sub me1   / < epsilon ?
          sma
          jmp mq2   / no
          add mt1
          sub me2
          spa
          jmp cox
mq2,      idx mx2   / end of comparison loop
          idx my2
          idx ma2
          idx mb2
          index ml2, (lac mth nob, ml2
          idx cox
cox,      jmp .

```

/routine to set explosion

```

        law 20
        dac i mb1
        dac i mb2
sex,     lac (mex 400000 /EXPLODE
        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 .
```

/ misc calculation routines

/ explosion

```

mex,     lac i mdr
        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 mxc
        sub (140
        sma
        idx msh
mz1,     lac ran
        and (777
        lor (scl
        dac ml1
        random
        scr 9s
        sir 9s
msh,     xct .
ml1,     hlt
        add i my1
        swap
        add i mx1
        szs 20
        jda kcb
        dpy-i 300
        count mxc, mz1
        count i ma1, mb1
        dzm i ml1
        jmp mb1

mst,     scr 1s
        scr 3s
```

/ torpedo calc routine

```
ter,      jsp col
          jmp sex-3
          count i ma1, tc1
          lac (mex 400000
          dac i rl1
          law i 2
          dac i ma1
          law 20
          dac i mb1
          jmp mb1
```

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

/ hyperspace routines

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

```
hp1,      count i ma1, mb1
          law hp3
          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
```

/ next step

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

```
hp3,      jsp col
          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 hpt, 3
          lac ran
          sar 6s
          dac i mom
          lac ran
          dac i mth
hp4,      lac i mth
          sma
          sub (311040
          spa
          add (311040
          dac i mth
          count hpt, 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
```

```
keb,      0                      /relocate for center display
          dap kc1
          swap
          sub ny1
          swap
          lac keb
          sub nx1
kc1,      jmp .
```

```

/ spaceship calc
ss1,      jsp i cwg
          dio scw
          jmp sr0

ss2,      jsp i cwg
          rir 4s
          dio scw

sr0,
sc1,      clf 6
          jsp col
          jmp sex
          lio scw
          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 7s
          ril 1s
          spi
          stf 6
          lio i mfu
          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 mx1
          dac t1
          mul t1
          scr 1s
          dac acx
          cla
          scr 2s
          dio tox
          lac i my1
          dac t1
          mul t1
          scr 1s
          dac acy

```

```

cla
scr 2s
swap
add  $\bar{f}ox$ 
swap
scr 2s
add  $\bar{a}cx$ 
add  $\bar{a}cy$ 
sub str
sma i sza-skip
jmp pof
add str
varsft
dac  $\bar{t}1$ 
jda sct
mul  $\bar{t}1$ 
undosft
scr 9s
scr 8s
sza
jmp bsg
scr 1s
dio  $\bar{t}1$ 
integrate mx1,  $\bar{b}x$ 
integrate my1,  $\bar{b}y$ 
bsg, lac i mth
jda cos
dac  $\bar{c}s$ 
sar 9s
xct sac
szf i 6
cla
add  $\bar{b}y$ 
diff mdy, my1, (sar 3s
lac  $\bar{s}n$ 
sar 9s
xct sac
cmā
szf i 6
cla
add  $\bar{b}x$ 
diff mdx, mx1, (sar 3s
scale  $\bar{s}n$ , 5s,  $\bar{s}sn$ 
scale  $\bar{c}s$ , 5s,  $\bar{s}cn$ 
lac i mx1
szs 20
sub  $\bar{n}x1$ 
sub  $\bar{s}sn$ 
dac  $\bar{s}x1$ 
sub  $\bar{s}sn$ 
dac str

```

```

lac i my1
szs 20
sub ny1
add scn
dac sy1
add scn
dac sty
scale sn, 9s, ssn
scale cs, 9s, sen
dac scm
lac ssn
dac ssm
add scn
dac ssc
dac ssd
lac ssn
sub scn
dac csn
cma
dac csm
cla cli-opr
szs 20
jda kcb
dpy-4000
mot,sp5, jmp i .
sq6, ioh
lio scw
ril 2s
spi i / not blasting
jmp sq9 / no tail
ranct sar 9s, sar 4s, src
scale sn, 8s, ssn
scale cs, 8s, sen
sq7, count i mfu, st2
dzm i mfu
jmp sq9
st2, yinor sx1, sy1, sub
dispt i sy1
count src, sq7
sq9, count i ma1, sr5 / check if torp tube reloaded
dzm i ma1 / prevent count around
mco, lac . / previous control word
cma
szs i 30
clc
and scw / present control word
ral 3s / torpedo bit to bit 0
sma
jmp sr5 / no launch
count i mtr, 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

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 tv1
        cma
        add i mdx
sr3,    dac .
        lac cs
        xct tv1
        add i mdy
sr4,    dac .
        xct rlt
        dac i ma                               / 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 i mh2
        sza i
        jmp mb1
        lac scw
        and (600000
        xor (600000
        sza
        jmp mb1
        lac i ml1
        dac i mh1
        lac (hp1 400000
        dac i ml1
        xct hd1
        dac i ma1
        law 2
        dac i mb1

```

/ here to handle spaceships dragged into star

/ spaceship in star

```
pof,      dzm i m̄dx
           dzm i mdy
           szs 50
           jmp po1
           lac (377777
           dac i mx1
           dac i my1
           lac i mb1
           dac ssn
           count ssn, .
           jmp mb1
```

```
po1,      lac (mex 400000 / now go bang
           dac i ml1
           law i 10
           dac i ma1
           jmp mb1
```

/ outlines of spaceships

```
ot1,      111131
           111111
           111111
           111163
           311 11
           146111
           111114
           700000
```

. 5/

```
ot2,      013113
           113111
           116313
           131111
           161151
           111633
           365114
           700000
```

. 5/

variables
constants

.-64/

p,

. 100/

/ space for patches

/display score routine

```
fss,      0          /set size of spaceship
          dap fs1
          lac fss
          dac scm
          dac ssc
          dac ssd
          dac sen
          dac csm
          cma
          dac csn
          dzm ssn
          dzm ssn
fs1,      jmp .
fi1,      law .      /set return of compiled outline
          sub c21
          dac t6
fi2,      law .
          sub c21
          dac t7
fis,      lac c23
          dac t4
          szf 3
          lio 2sc    /get score
          szf 1 3
          lio 1sc
          scl 1s
          cla
          div c12
          hlt
          sz2
          jmp fx1
fkr,      dio t3
          law 400
          jda fss
          law fys
          dap frt
flt,      idx t3
          cma
          dac t3
          law fus
          dap 1 t6
          dap 1 t7
fus,      lac c20
          szf 3
          cma
          dac sy1
          lac t4
          add c30
          dac t4
          dac sx1
```

```

fds,      szf 3      /display spaceship
          law not
          szf 1 3
          law not 1
          dap fug
          idx t5
          ral 9s
          cli
          dpy -4000+700
          isp t3
fug,      jmp 1 .
frr,      jmp .

fys,      law 4000
          jda fss
          law fub
          dap 1 t6
          dap 1 t7
          lac c26
          dac sx1
          lac c20
          szf 1 3
          cma
          add c30
          dac sy1
          law 1 2
          dac t3
          jmp fds
fub,      szf 3
          jmp . 3
          str 3
          jmp . 2
          clf 3
          iot 11
          dio t1
          iot 111
          dio t2
          law 1
          and t1
          and t2
          sza
          jmp fik
          law 2
          and t1
          and t2
          sza 1
          jmp fis
          law a4+
          dap fiu
          jmp fwt
fik,      law 4
          dap fiu
fwt,      add .
          dec t1
          isp t1
          jmp .-1
fiu,      jmp .

```

```

fx1,      dio t1
          dac t3
          law fx2
          dap frt
          law 1100
          jda fss
          jmp flt
fx2,      lio t1
          jmp flr

c12,      12
c20,      200000
c21,      21
c23,      -200000-30000
c26,      -260000
c30,      30000
t3,       0
t4,       0
t5,       0
t6,       0
t7,       0

```

```

mtb,      / table of objects and their properties

```

```

start 4

```

spacewar 4.3 5/17/63 ddp . pt 1 - pass 1
spacewar 4.3 5/17/63 ddp . pt 2 - pass 1
stars by prs for s/w 2b - pass 1
f
nx1=mtb nob
f

nx1=mtb nob
ny1=nx1 nob
start
f - pass 1
spacewar 4.3 5/17/63 ddp . pt 1 - pass 2
spacewar 4.3 5/17/63 ddp . pt 2 - pass 2
stars by prs for s/w 2b - pass 2
spacewar 4.3 syms 5/23/63 jcm

Constants area, inclusive
from to
2763 3111

1j	6077
1m	704
1q	6117
1sc	1323
2j	6121
2m	763
2q	6141
2sc	1324
3j	6143
3m	1042
3q	6403
4j	6405
4m	1121
4q	7747
8a	1520
8aa	1570
8ad	1543
8ao	1555
8aq	1562
8ax	1602
8ay	1533
8j1	1440
8j2	1444
a	1325
a1	1315
a2	1362
a3	1413
a4	1345
a40	1320
a5	1341
a6	1354
acx	2754
acy	2756
bc1	665
bc3	671
bcc	1200
bck	654
bcx	703
bcy	702
bds	542
bjm	541
bkc	1201
blp	466
blx	513
bpt	514
bpx	643
bsg	2336
bx	2731
by	2732
c12	3423
c21	3425
c23	3426
c26	3427
c20	3424

c30	3430
col	1603
cos	100
cox	1663
cs	2757
esm	2723
csn	2726
csx	146
cwg	2745
cwr	40
ddd	20
fds	3322
fi1	3255
fi2	3260
fid	25
fie	16
fik	3403
fin	7
fis	3263
fiu	3411
fkr	3277
flo	56
flp	46
flt	3304
foo	12
fou	15
fpo	55
fpr	1208
frr	14
frr	3335
fs	4
fs1	3254
fss	3240
fub	3355
fug	3334
fus	3312
fuu	35
fwf	3405
fx	36
fx1	3412
fx2	3421
fyn	20
fys	3336
gct	2746
grv	32
hd1	23
hd2	24
hd3	25
hp1	2015
hp3	2046
hp4	2107
hp6	2124
hp7	2122
hpt	2751

hr1	26
hr2	27
hur	30
idl	1456
ids	1470
iox	2755
kc1	2143
kcb	2132
ma1	1677
ma2	1700
maa	13
mb1	1503
mb2	1673
mco	2511
md1	1267
mdn	1273
mdx	2734
mdy	2735
me1	16
me2	17
mex	1701
mfu	2736
mg1	1435
mg2	1451
mg3	1450
mg4	1455
mh1	2740
mh2	2741
mh3	2742
mh4	2743
mhs	22
mi1	1736
ml1	1476
ml2	1625
ml0	1203
mom	2164
mot	2440
mq1	1506
mq2	1653
msh	1735
mst	1754
mt1	2747
mtb	3436
mtc	2733
mtb	2200
mtr	2737
mx1	1630
mx2	1631
mxo	2750
my1	1640
my2	1641
mz1	1722
na1	3546
nb1	3576

hco	3720
ndx	3626
ndy	3656
nfu	3712
nh1	3722
nh2	3724
nh3	3726
nh4	3730
nnn	3732
nob	30
nom	3706
not	3716
ntd	2744
nth	3710
ntr	3714
nx1	3466
ny1	3516
oc	230
oc1	375
oc2	420
oc3	426
oc4	434
oc5	442
oc6	450
oc9	460
occ	2717
ocd	411
oce	402
ocg	252
och	253
oci	2720
ocj	250
ock	241
ocm	373
ocn	374
oco	264
ocp	266
ocq	265
ocr	267
ocs	222
ocx	372
ocz	227
ot1	2655
ot2	2672
p	3140
po1	2650
pof	2634
ran	31
rlt	10
sac	14
sb1	61
sc1	2152
scm	2721
scn	2710

scw	2752
sex	1667
sft	66
sl1	112
sl2	147
sl3	117
sln	106
sn	2753
sp5	2440
sq1	220
sq2	221
sq3	171
sq6	2441
sq7	2466
sq9	2506
sqt	162
sqx	174
sr1	2527
sr2	2541
sr3	2570
sr4	2574
sr5	2606
sr6	2600
sr7	2605
src	2762
sr0	2152
ss1	2144
ss2	2147
ss3	2547
ss4	2553
ssa	2727
ssc	2725
ssd	2724
ssi	2730
ssm	2722
ssn	2707
st1	2525
st2	2472
str	15
stx	2760
sty	2761
sx1	2715
sy1	2716
t1	2712
t2	2713
t3	3431
t4	3432
t5	3433
t6	3434
t7	3435
tc1	1771
ter	1756
the	21
tlf	11

tno	6
tv1	7
v2	3
xys	2711
xyt	2714