

stars by prs for s/w 2b

6077/

/stars 1 • 3/13/62, prs.

decimal

define

mark X, Y

repeat 8, Y=Y+Y

8192-X Y

terminate

1j,	mark 1537, 371	/87 Taur, Aldebaran
	mark 1762, -189	/19 Orio, Rigel
	mark 1990, 168	/58 Orio, Betelgeuze
	mark 2280, -377	/9 CMaj, Sirius
	mark 2583, 125	/10 CMin, Procyon
	mark 3431, 283	/32 Leon, Regulus
	mark 4551, -242	/67 Virg, Spica
	mark 4842, 448	/16 Boot, Arcturus
1q,	mark 6747, 196	/53 Aquil, Altair
2j,	mark 1819, 143	/24 Orio, Bellatrix
	mark 1884, -29	/46 Orio
	mark 1910, -46	/50 Orio
	mark 1951, -221	/53 Orio
	mark 2152, -407	/ 2 CMaj
	mark 2230, 375	/24 Gemi
	mark 3201, -187	/30 Hyda, Alphard
	mark 4005, 344	/94 Leon, Denebola
2q,	mark 5975, 288	/55 Ophi
3j,	mark 46, 333	/88 Pegs, Algenib
	mark 362, -244	/31 Ceti
	mark 490, 338	/99 Pisc
	mark 566, -375	/52 Ceti
	mark 621, 462	/ 6 Arie
	mark 764, -78	/68 Ceti, Mira
	mark 900, 64	/86 Ceti
	mark 1007, 84	/92 Ceti
	mark 1243, -230	/23 Erid
	mark 1328, -314	/34 Erid
	mark 1495, 432	/74 Taur
	mark 1496, 356	/78 Taur
	mark 1618, 154	/ 1 Orio
	mark 1644, 52	/ 8 Orio
	mark 1723, -119	/67 Erid
	mark 1755, -371	/ 5 Leps
	mark 1779, -158	/20 Orio
	mark 1817, -57	/28 Orio
	mark 1843, -474	/ 9 Leps
	mark 1860, -8	/34 Orio
	mark 1868, -407	/11 Leps
	mark 1875, 225	/39 Orio
	mark 1880, -136	/44 Orio
	mark 1887, 480	/123 Taur
	mark 1948, -338	/14 Leps
	mark 2274, 296	/31 Gemi

mark 2460, 380

/54 Gemi

mark	2470, 504	/55	Gemi	
mark	2513, 193	/ 3	CMin	
mark	2967, 154	/11	Hyda	
mark	3016, 144	/16	Hyda	
mark	3424, 393	/30	Leon	
mark	3496, 463	/41	Leon, Algieba	
mark	3668, -357	/nu	Hyda	
mark	3805, 479	/68	Leon	
mark	3806, 364	/10	Leon	
mark	4124, -502	/ 2	Corv	
mark	4157, -387	/ 4	Corv	
mark	4236, -363	/ 7	Corv	
mark	4304, -21	/29	Virg	
mark	4384, 90	/43	Virg	
mark	4421, 262	/47	Virg	
mark	4606, -2	/79	Virg	
mark	4721, 430	/ 8	Boot	
mark	5037, -356	/ 9	Libr	
mark	5186, -205	/27	Libr	
mark	5344, 153	/24	Serp	
mark	5357, 358	/28	Serp	
mark	5373, -71	/32	Serp	
mark	5430, -508	/ 7	Scor	
mark	5459, -445	/ 8	Scor	
mark	5513, -78	/ 1	Ophi	
mark	5536, -101	/ 2	Ophi	
mark	5609, 494	/27	Herc	
mark	5641, -236	/13	Ophi	
mark	5828, -355	/35	Ophi	
mark	5860, 330	/64	Herc	
mark	5984, -349	/55	Serp	
mark	6047, 63	/62	Ophi	
mark	6107, -222	/64	Ophi	
mark	6159, 217	/72	Ophi	
mark	6236, -66	/58	Serp	
mark	6439, -483	/37	Sgtr	
mark	6490, 312	/17	Aqil	
mark	6491, -115	/16	Aqil	
mark	6507, -482	/41	Sgtr	
mark	6602, 66	/30	Aqil	
mark	6721, 236	/50	Aqil	
mark	6794, 437	/12	Sgte	
mark	6862, -25	/65	Aqil	
mark	6914, -344	/ 9	Capr	
mark	7014, 324	/ 6	Dlph	
mark	7318, -137	/22	Aqar	
mark	7391, 214	/ 8	Pegs	
mark	7404, -377	/49	Capr	
mark	7513, -18	/34	Aqar	
mark	7539, 130	/26	Pegs	
mark	7644, -12	/55	Aqar	
mark	7717, 235	/42	Pegs	
mark	7790, -372	/76	Aqar	
mark	7849, 334	/54	Pegs, Markab	

4j,

mark	1, -143	/33	Pisc
mark	54, 447	/89	Pegs
mark	54, -443	/ 7	Ceti
mark	82, -214	/ 8	Ceti
mark	223, -254	/17	Ceti
mark	248, 160	/63	Pisc
mark	273, -38	/20	Ceti
mark	329, 167	/71	Pisc
mark	376, 467	/84	Pisc
mark	450, -198	/45	Ceti
mark	548, 113	/106	Pisc
mark	570, 197	/110	Pisc
mark	595, -255	/53	Ceti
mark	606, -247	/55	Ceti
mark	615, 428	/ 5	Arie
mark	617, 61	/14	Pisc
mark	656, -491	/59	Ceti
mark	665, 52	/113	Pisc
mark	727, 191	/65	Ceti
mark	803, -290	/72	Ceti
mark	813, 182	/73	Ceti
mark	838, -357	/76	Ceti
mark	878, -2	/82	Ceti
mark	907, -340	/89	Ceti
mark	908, 221	/87	Ceti
mark	913, -432	/ 1	Erid
mark	947, -487	/ 2	Erid
mark	976, -212	/ 3	Erid
mark	992, 194	/91	Ceti
mark	1058, 440	/57	Arie
mark	1076, 470	/58	Arie
mark	1087, -209	/13	Erid
mark	1104, 68	/96	Ceti
mark	1110, -503	/16	Erid
mark	1135, 198	/ 1	Taur
mark	1148, 214	/ 2	Taur
mark	1168, 287	/ 5	Taur
mark	1170, -123	/17	Erid
mark	1185, -223	/18	Erid
mark	1191, -500	/19	Erid
mark	1205, 2	/10	Taur
mark	1260, -283	/26	Erid
mark	1304, -74	/32	Erid
mark	1338, 278	/35	Taur
mark	1353, 130	/38	Taur
mark	1358, 497	/37	Taur
mark	1405, -162	/38	Erid
mark	1414, 205	/47	Taur
mark	1423, 197	/49	Taur
mark	1426, -178	/40	Erid
mark	1430, 463	/50	Taur
mark	1446, 350	/54	Taur
mark	1463, 394	/61	Taur
mark	1470, 392	/64	Taur
mark	1476, 502	/65	Taur
mark	1477, 403	/68	Taur

mark 1483, 350	/71 Taur
mark 1485, 330	/73 Taur
mark 1495, 358	/77 Taur
mark 1507, 364	/
mark 1518, -6	/45 Erid
mark 1526, 333	/86 Taur
mark 1537, 226	/88 Taur
mark 1544, -81	/48 Erid
mark 1551, 280	/90 Taur
mark 1556, 358	/92 Taur
mark 1557, -330	/53 Erid
mark 1571, -452	/54 Erid
mark 1596, -78	/57 Erid
mark 1622, 199	/ 2 Orio
mark 1626, 124	/ 3 Orio
mark 1638, -128	/61 Erid
mark 1646, 228	/ 7 Orio
mark 1654, 304	/ 9 Orio
mark 1669, 36	/10 Orio
mark 1680, -289	/64 Erid
mark 1687, -167	/65 Erid
mark 1690, -460	/
mark 1690, 488	/102 Taur
mark 1700, 347	/11 Orio
mark 1729, 352	/15 Orio
mark 1732, -202	/69 Erid
mark 1750, -273	/ 3 Leps
mark 1753, 63	/17 Orio
mark 1756, -297	/ 4 Leps
mark 1792, -302	/ 6 Leps
mark 1799, -486	/
mark 1801, -11	/22 Orio
mark 1807, 79	/23 Orio
mark 1816, -180	/29 Orio
mark 1818, 40	/25 Orio
mark 1830, 497	/114 Taur
mark 1830, 69	/30 Orio
mark 1851, 134	/32 Orio
mark 1857, 421	/119 Taur
mark 1861, -168	/36 Orio
mark 1874, 214	/37 Orio
mark 1878, -138	/
mark 1880, -112	/42 Orio
mark 1885, 210	/40 Orio
mark 1899, -60	/48 Orio
mark 1900, 93	/47 Orio
mark 1900, -165	/49 Orio
mark 1909, 375	/126 Taur
mark 1936, -511	/13 Leps
mark 1957, 287	/134 Taur
mark 1974, -475	/15 Leps
mark 1982, 461	/54 Orio
mark 2002, -323	/16 Leps
mark 2020, -70	/
mark 2030, 220	/61 Orio

mark 2032, -241	/ 3	Mono
mark 2037, 458	/62	Orio
mark 2057, -340	/18	Leps
mark 2059, 336	/67	Orio
mark 2084, 368	/69	Orio
mark 2084, 324	/70	Orio
mark 2105, -142	/ 5	Mono
mark 2112, -311	/	
mark 2153, 106	/ 8	Mono
mark 2179, 462	/18	Gemi
mark 2179, -107	/10	Mono
mark 2184, -159	/11	Mono
mark 2204, 168	/13	Mono
mark 2232, -436	/ 7	CMaj
mark 2239, -413	/ 8	CMaj
mark 2245, -320	/	
mark 2250, 227	/15	Mono
mark 2266, 303	/30	Gemi
mark 2291, 57	/18	Mono
mark 2327, 303	/38	Gemi
mark 2328, -457	/15	CMaj
mark 2330, -271	/14	CMaj
mark 2340, -456	/19	CMaj
mark 2342, -385	/20	CMaj
mark 2378, -93	/19	Mono
mark 2379, 471	/43	Gemi
mark 2385, -352	/23	CMaj
mark 2428, -8	/22	Mono
mark 2491, -429	/	
mark 2519, 208	/ 4	CMin
mark 2527, 278	/ 6	CMin
mark 2559, -503	/	
mark 2597, -212	/26	Mono
mark 2704, -412	/	
mark 2709, -25	/28	Mono
mark 2714, 60	/	
mark 2751, -61	/29	Mono
mark 2757, -431	/16	Pupp
mark 2768, -288	/19	Pupp
mark 2794, 216	/17	Canc
mark 2848, -82	/	
mark 2915, 138	/ 4	Hyda
mark 2921, 84	/ 5	Hyda
mark 2942, -355	/ 9	Hyda
mark 2944, 497	/43	Canc
mark 2947, 85	/ 7	Hyda
mark 2951, -156	/	
mark 2953, 421	/47	Canc
mark 2968, -300	/12	Hyda
mark 2976, 141	/13	Hyda
mark 3032, 279	/65	Canc
mark 3124, 62	/22	Hyda
mark 3157, -263	/26	Hyda
mark 3161, -208	/27	Hyda
mark 3209, -53	/31	Hyda
mark 3225, -17	/32	Hyda
mark 3261, 116	/	

mark 3270, -16	/35 Hyda
mark 3274, -316	/38 Hyda
mark 3276, 236	/14 Leon
mark 3338, -327	/39 Hyda
mark 3385, 194	/29 Leon
mark 3415, -286	/40 Hyda
mark 3428, 239	/31 Leon
mark 3429, 3	/15 Sext
mark 3446, -270	/41 Hyda
mark 3495, 455	/40 Leon
mark 3534, -372	/42 Hyda
mark 3557, -3	/30 Sext
mark 3570, 223	/47 Leon
mark 3726, -404	/a1 Crat
mark 3736, -44	/61 Leon
mark 3738, 471	/60 Leon
mark 3754, 179	/63 Leon
mark 3793, -507	/11 Crat
mark 3821, -71	/74 Leon
mark 3836, -324	/12 Crat
mark 3846, 150	/77 Leon
mark 3861, 252	/78 Leon
mark 3868, -390	/15 Crat
mark 3935, -211	/21 Crat
mark 3936, -6	/91 Leon
mark 3981, -405	/27 Crat
mark 3986, 161	/ 3 Virg
mark 3998, 473	/93 Leon
mark 4013, 53	/ 5 Virg
mark 4072, 163	/ 8 Virg
mark 4097, 211	/ 9 Virg
mark 4180, -3	/15 Virg
mark 4185, 418	/11 Coma
mark 4249, -356	/ 8 Corv
mark 4290, -170	/26 Virg
mark 4305, 245	/30 Virg
mark 4376, -205	/40 Virg
mark 4403, 409	/36 Coma
mark 4465, -114	/51 Virg
mark 4466, 411	/42 Coma
mark 4512, -404	/61 Virg
mark 4563, -352	/69 Virg
mark 4590, -131	/74 Virg
mark 4603, 95	/78 Virg
mark 4679, 409	/ 4 Boot
mark 4691, 371	/ 5 Boot
mark 4759, 46	/93 Virg
mark 4820, 66	/
mark 4822, -223	/98 Virg
mark 4840, -126	/99 Virg
mark 4857, -294	/100 Virg
mark 4864, 382	/20 Boot
mark 4910, -41	/105 Virg
mark 4984, 383	/29 Boot
mark 4986, 322	/30 Boot
mark 4994, -119	/107 Virg

mark 5009, 396	/35 Boot
mark 5013, 53	/109 Virg
mark 5045, 444	/37 Boot
mark 5074, -90	/16 Libr
mark 5108, 57	/110 Virg
mark 5157, -442	/24 Libr
mark 5283, -221	/37 Libr
mark 5290, -329	/38 Libr
mark 5291, 247	/13 Serp
mark 5326, -440	/43 Libr
mark 5331, 455	/21 Serp
mark 5357, 175	/27 Serp
mark 5372, 420	/35 Serp
mark 5381, 109	/37 Serp
mark 5387, 484	/38 Serp
mark 5394, -374	/46 Libr
mark 5415, 364	/41 Serp
mark 5419, -318	/48 Libr
mark 5455, -253	/xi Scor
mark 5467, -464	/ 9 Scor
mark 5470, -469	/10 Scor
mark 5497, -437	/14 Scor
mark 5499, -223	/15 Scor
mark 5558, 29	/50 Serp
mark 5561, 441	/20 Herc
mark 5565, -451	/ 4 Ophi
mark 5580, 325	/24 Herc
mark 5582, -415	/ 7 Ophi
mark 5589, -186	/ 3 Ophi
mark 5606, -373	/ 8 Ophi
mark 5609, 50	/10 Ophi
mark 5610, -484	/ 9 Ophi
mark 5620, 266	/29 Herc
mark 5713, -241	/20 Ophi
mark 5742, 235	/25 Ophi
mark 5763, 217	/27 Ophi
mark 5807, 293	/60 Herc
mark 5868, -8	/41 Ophi
mark 5888, -478	/40 Ophi
mark 5889, -290	/53 Serp
mark 5924, -114	/
mark 5925, 96	/49 Ophi
mark 5987, -183	/57 Ophi
mark 6006, -292	/56 Serp
mark 6016, -492	/58 Ophi
mark 6117, -84	/57 Serp
mark 6117, 99	/66 Ophi
mark 6119, 381	/93 Herc
mark 6119, 67	/67 Ophi
mark 6125, 30	/68 Ophi
mark 6146, 57	/70 Ophi
mark 6158, 198	/71 Ophi
mark 6170, 473	/102 Herc
mark 6188, -480	/13 Sgtr
mark 6234, 76	/74 Ophi
mark 6235, 499	/106 Herc



mark 6247, -204	/xi Scut
mark 6254, -469	/ε1 Sgtr
mark 6255, 494	/109 Herc
mark 6278, -333	/ga Scut
mark 6313, -189	/a1 Scut
mark 6379, 465	/110 Herc
mark 6382, -110	/be Scut
mark 6386, 411	/111 Herc
mark 6436, 93	/63 Serp
mark 6457, 340	/13 Aqil
mark 6465, -134	/12 Aqil
mark 6478, -498	/39 Sgtr
mark 6553, 483	/ 1 Vulp
mark 6576, -410	/44 Sgtr
mark 6576, -368	/46 Sgtr
mark 6607, 3	/32 Aqil
mark 6651, 163	/38 Aqil
mark 6657, 445	/ 9 Vulp
mark 6665, -35	/41 Aqil
mark 6688, 405	/ 5 Sgte
mark 6693, 393	/ 6 Sgte
mark 6730, 416	/ 7 Sgte
mark 6739, 430	/ 8 Sgte
mark 6755, 17	/55 Aqil
mark 6766, 187	/59 Aqil
mark 6772, 140	/60 Aqil
mark 6882, 339	/67 Aqil
mark 6896, -292	/ 5 Capr
mark 6898, -292	/ 6 Capr
mark 6913, -297	/ 8 Capr
mark 6958, -413	/11 Capr
mark 6988, 250	/ 2 Dlph
mark 7001, 326	/ 4 Dlph
mark 7015, -33	/71 Aqil
mark 7020, 475	/29 Vulp
mark 7026, 354	/ 9 Dlph
mark 7047, 335	/11 Dlph
mark 7066, 359	/12 Dlph
mark 7067, -225	/ 2 Aqar
mark 7068, -123	/ 3 Aqar
mark 7096, -213	/ 6 Aqar
mark 7161, -461	/22 Capr
mark 7170, -401	/23 Capr
mark 7192, -268	/13 Aqar
mark 7199, 222	/ 5 Equl
mark 7223, 219	/ 7 Equl
mark 7230, 110	/ 8 Equl
mark 7263, -393	/32 Capr
mark 7267, 441	/ 1 Pegs
mark 7299, -506	/36 Capr
mark 7347, -453	/39 Capr
mark 7353, -189	/23 Aqar
mark 7365, -390	/40 Capr
mark 7379, -440	/43 Capr
mark 7394, 384	/ 9 Pegs
mark 7499, -60	/31 Aqar
mark 7513, 104	/22 Pegs

mark 7515, -327	/33	Aqar
mark 7575, -189	/43	Aqar
mark 7603, -43	/48	Aqar
mark 7604, 266	/31	Pegs
mark 7624, 20	/52	Aqar
mark 7639, 96	/35	Pegs
mark 7654, -255	/57	A ar
mark 7681, -14	/62	Aqar
mark 7727, -440	/66	Aqar
mark 7747, 266	/46	Pegs
mark 7761, -321	/71	Aqar
mark 7779, -185	/73	Aqar
mark 7795, 189	/50	Pegs
mark 7844, 75	/ 4	Pisc
mark 7862, 202	/55	Pegs
mark 7874, -494	/88	Aqar
mark 7903, -150	/90	Aqar
mark 7911, -219	/91	Aqar
mark 7919, 62	/ 6	Pisc
mark 7923, -222	/93	Aqar
mark 7952, -470	/98	Aqar
mark 7969, -482	/99	Aqar
mark 7975, 16	/ 8	Pisc
mark 7981, 133	/10	Pisc
mark 7988, 278	/70	Pegs
mark 8010, -489	/101	Aqar
mark 8049, 116	/17	Pisc
mark 8059, -418	/104	Aqar
mark 8061, 28	/18	Pisc
mark 8064, -344	/105	Aqar
mark 8159, 144	/28	Pisc
mark 8174, -149	/30	Pisc
mark 8188, -407	/ 2	Ceti

4q,

start 4

/title punch table

ftp/	0	0	004277	400000
	625151	514600	224145	453200
	141211	771000	274545	453100
	364545	453000	010171	050300
	324545	453200	065151	513600
	0	0	0	0
	0	0	0	0
	0	0	0	0
	364141	413600	402014	020100
	224545	453000	010177	010100
	374040	403700	073060	300700
	376010	603700	412214	224100
	010274	020100	615141	454300
	0	0	141414	141400
	0	0	0	0
	0	0	0	0
	0	0	204040	403700
	771014	224100	774040	404000
	770214	027700	770214	207700
	364141	413600	771111	110600
	364151	215600	771111	314600
	0	0	0	0
	101010	101000	000041	221400
	101074	101000	001422	410000
	0	0	761111	117600
	774545	453200	364141	412200
	774141	413600	774545	414100
	770505	010100	364151	513000
	771010	107700	004177	410000
	010300	010300	000060	600000
	030200	030200		

start ps5

spacewar 4.4 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
```

/ 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+E
          lio Y
          jda knob
          xct db1
          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
  sma
  cma
  dac C
  terminate
```

define

```
varsft
dzm  $\overline{xy}$ s
dac  $\overline{t1}$ 
idx  $\overline{xy}$ s
v2, idx  $\overline{xy}$ s
lac  $\overline{t1}$ 
scr 2s
dac  $\overline{t1}$ 
sza
jmp v2+R
scr 2s
swap
terminate
```

define

```
undosft
dac t1
dio  $\overline{t2}$ 
lac  $\overline{xy}$ s
add sft
dap .+
lac .
dac .+6
dac .+6
xor (10000) / changescr to scl or scl to scr.
dac  $\overline{xyt}$ 
lac  $\overline{t1}$ 
dio  $\overline{t2}$ 
scr .
scr .
terminate
```

define

```
integrate A,B
cli
lac i A
scr 9s
scr 1s
div  $\overline{t1}$ 
hlt
cma+cli-opr
xct  $\overline{xyt}$ 
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
si1,      add (311040
          sub (62210
          sma
          jmp si2
          add (62210

si3,      ral 2s
          mul (242763
          dac sin
          mul sin
          dac cos
          mul (756103
          add (121312
          mul cos
          add (532511
          mul cos
          add (144417
          mul sin
          sel 3s
          dac cos
          xor sin
          sma
          jmp csx-1
          lac (377777
          lio sin
          spi
          cma
          jmp csx

csx,      lac cos
          jmp .

si2,      cma
          add (62210
          sma
          jmp si3
          add (62210
          spa
          jmp .+3
          sub (62210
          jmp si3

          sub (62210
          jmp si1
  
```



/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

```
sq1,      0
          dap sqx
          law 1 23
          dac sq1
          dzn sq2
          lio sqt
          dzn sqt

sq3,      isp sq1
          jmp .+3
          lac sq2
sqx,      jmp .

          lac sq2
          sal 1s
          dac sq2
          lac sqt
          rcl 2s
          sza 1
          jmp sq3
          dac sqt
          lac sq2
          sal 1s
          add (1
          sub sqt
          sna+sza-skp
          jmp sq3
          spe
          onz
          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
```

```
ocz,
```

```
jmp .
```

```
oc,
```

```
0 /outline compiler proper  
dap ocz  
lac i ocz  
dap ocg  
plinst (stf 5  
dap ocm  
idx ocz
```

```
ock,
```

```
plinst (lac  $\overline{sx1}$   
plinst (lio  $\overline{sy1}$   
clf 6
```

```
ocj,
```

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

```
ocg,
```

```
lio .
```

```
och,
```

```
cla /outline table
```

```
rcl  $\overline{3s}$ 
```

```
dio  $\overline{oci}$ 
```

```
lio (swp swp=opr 60
```

```
dispatch
```

```
opr
```

```
jmp oc1
```

```
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 sc6
plinst (clf 5
plinst (lac  $\overline{scm}$ 
plinst (cma
plinst (dac  $\overline{scm}$ 
plinst (lac  $\overline{ssm}$ 
plinst (cma
plinst (dac  $\overline{ssm}$ 
plinst (lac  $\overline{csm}$ 
plinst (lio  $\overline{ssd}$ 
plinst (dac  $\overline{ssd}$ 
plinst (dio  $\overline{csm}$ 

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

oem,    jmp .
ocn,    jmp .

oc1,    plinst (add  $\overline{sen}$ 
        jsp ocs
        lac (sub  $\overline{sen}$ 
oce,    dac i oc
        idx oc
        jsp ocs
        plinst (ioh
ocd,    lac (xet db2
        dac i oc
        idx oc
        lio  $\overline{oci}$ 
        count  $\overline{oc}$ , oeh
        idx oeg
        jmp ocj

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

```

/display gravitational star

define

starp  
add bx  
swap  
add by  
swap  
loh  
ret db2  
terminate

blp,           dap blx                           /star

szs 60  
jmp blx  
random  
rar 9s  
and (400700  
spa  
xor (377777  
dac bx  
lac r0n  
ral 4s  
and (400700  
spa  
xor (377777  
dac by  
jsp bpt  
loh  
jmp .

blx,

bpt,

dap bpx  
random  
sar 9s  
sar 6s  
spa  
cma  
sal 3s  
add (bds  
dap bjm  
clf 6

bjl,

cla pli-opr  
szf 4  
jmp bjc  
sub ny1 1  
swp  
sub nx1 1  
jmp bjm-1

bjc,

sub ny1  
swp  
sub nx1  
ret db2

bjm,

bds,

jmp .  
repeat 10, starp  
szf 6

bpx,

jmp .  
stf 6  
lac bx  
cma  
dac bx

/background display \* 3/13/68, prs.

```

define
dislis J, C
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,
fou,
fie,
fyn,
spc
jmp fuu+R
sub (1000
sal 8s
lio /lio Y
jda kecb
xct db1
stf 5
fid,
idx fyn+R
sad (lio 0+2
jmp flp+R
sad fpo+R
jmp fx+R
dap fin+R
idx fyn+R
jmp fin+R

fuu,
fx,
szf 5
jmp /return
idx flo+R
idx flo+R
sas (0+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+
dap fyn+R
jmp fin+R

fpo,
flo,
lio
J
terminate

```

```

define
background      jsp bck
                 termin

bck,             dap bck
                 szs 40
                 jmp bck
                 jsp 1m
                 jsp 2m
                 idx bec
                 and (1
                 sza
                 jsp 3m
bc1,             law 3
                 and bec
                 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 1m
bex,             jmp

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

bec,            0
bkc,            0
fpr,            10000

```

mul=mus  
div=dis

start

/main control routine for spaceships

```

nob=30 /total number of colliding objects

m10, setup mtc, 5000 /delay for loop
      init m11, mtb /loc of calc routines
      add (nob
      dap mx1 / x
nx1=mtb nob
      add (nob
      dap my1 / 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 mdy / dy
ndy=ndx nob
      add (nob
      dap mom /angular velocity
nom=ndy nob
      add (2
      dap mth / angle
nth=nom 2
      add (2
      dac mfu /fuel
nfu=nth 2
      add (2
      dac mtr / no torps remaining
ntr=nfu 2
      add (2
      dap mot / outline of spaceship
not=ntr 2
      add (2
      dap mco / old control word
nco=not 2
      add (2
      dac mh1
nh1=nco 2
      add (2
      dac mh2
nh2=nh1 2
      add (2
      dac mh3
nh3=nh2 2
      add (2
      dac mh4
nh4=nh3 2
nnn=nh4 2

```



4

```
szf 4  
jmp . 4  
law dj6  
stf 4  
jmp . 3  
law dj5  
clf 4  
dap . 1  
lac .  
dac db1  
idx .-2  
xct .-3  
dac db2
```

```
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
```

md1,

```
jmp mdn  
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
```

db1,

0

db2,

0

dj5,

dpy-1

dpy-4000

dj6,

dpy-1 400

dpy-4000 400

```

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

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

1sc,     0               /scores
2sc,     0

a,       lac gct
         sma
         jmp a5
         count gct, a5
         lac 1sc
         sas 2sc
         jmp a4
         lac fiu
         sad ( jmp 4
         jmp a4
         law i1
         dac gct

a5,      lat
         and (40
         sza i
         jmp a2

a4,      jmp fi1
         lat
         and (40
         sza
         jmp a2
         dzm 1sc
         dzm 2sc

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 rx1
         dac ny1
         cma
         dec rx1 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
dac not 1
dap fi1
jda oc
ot2
dap fi2
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
jmp m10
```

a3,

/ 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

m11,    lac .
        sza i
        jmp mc1
        dap . 1
        jmp .
mb1,    lac .
        add mtc
        dac mtc
mc1,    idx m11
        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
```

```

col,      dap cox
          law 1
          add ml1
          sad (lac mtb 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 .           / 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 mt1
          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
mq2,      jmp cox
          idx mx2       / end of comparison loop
          idx my2
          idx ma2
          idx mb2
          index ml2, (lac mtb 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 m $\bar{d}$ x
    sar 3s
    add i mx1
    dac i mx1
    lac i m $\bar{d}$ y
    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
    for (scl
    dac mi1
    random
    scr 9s
    sir 9s
msh,   xct .
mi1,   hlt
    add i my1
    swap
    add i mx1
    jda ke $\bar{b}$ 
    xct db $\bar{1}$ 
    count mxc, mz1
    count i ma $\bar{1}$ , mb1
    dzn i ml1
    jnp mb1

mst,   ser 1s
    ser 3s
```

/ torpedo calc routine

```
tor,      jsp col
          jmp sex-3
          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
          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 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,    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

kcb,    0
        dap kc1
        lac kcb
        szf 4
        jmp . 6
        sub nx1 1
        swap
        sub ny1 1
        swap
        jmp kc1
        sub nx1
        swap
        sub ny1
        swap
kc1,    jmp .
```

/relocate for center display



/ spaceship calc

ss1,        jsp i  $\overline{cwg}$   
             dio  $\overline{scw}$   
             jmp sr0

ss2,        jsp i  $cwg$   
             rir  $\overline{4s}$   
             dio  $\overline{scw}$

sr0,  
sc1,

      clf 6  
      jsp col  
      jmp  $\overline{sex}$   
      lio  $\overline{scw}$   
      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  $\overline{mfu}$   
             spi i  
             clf 6

mth,

      add .  
      sma  
      sub (311040  
      spa  
      add (311040  
      dac i mth  
      jda  $\overline{sin}$   
      dac  $\overline{sn}$   
      dzm  $\overline{bx}$   
      dzm  $\overline{by}$   
      szs 60  
      jmp bsg  
      lac i mx1  
      dac  $\overline{t1}$   
      mul  $\overline{t1}$   
      scr 1s  
      dac  $\overline{acx}$   
      cla  
      scr 2s  
      dio  $\overline{iox}$   
      lac i my1  
      dac  $\overline{t1}$   
      mul  $\overline{t1}$   
      scr 1s  
      dac  $\overline{acy}$

```
cla
scr 2s
swap
add  $\bar{i}ox$ 
swap
scl  $\underline{2s}$ 
add  $\bar{a}cx$ 
add  $\bar{a}cy$ 
sub str
sma i sza-skip
jmp pof
add str
varsft
dac  $\bar{t}1$ 
jda  $\underline{sq}t$ 
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,  $\bar{b}y$ 
lac i mth
jda cos
dac  $\bar{c}s$ 
sar 9s
xct sac
szf i 6
cla
add  $\bar{b}y$ 
diff  $\bar{m}dy$ , my1, (sar 3s
lac  $\bar{s}n$ 
sar 9s
xct sac
cma
szf i 6
cla
add  $\bar{b}x$ 
diff  $\bar{m}dx$ , mx1, (sar 3s
scale  $\bar{s}n$ , 5s,  $\bar{s}sn$ 
scale  $\bar{c}s$ , 5s,  $\bar{s}cn$ 
lac i mx1
szf 4
sub nx1
szf i 4
sub nx1 1
sub  $\bar{s}sn$ 
dac  $\bar{s}x1$ 
sub  $\bar{s}sn$ 
dac  $\bar{s}tx$ 
```

bsg,

```

lac i my1
szf 4
sub ny1
szf i 4
sub ny1 1
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
jda kcb
xct db2
mot, sp5, jmp i .
sc6,   ioh _
      lio scw
      ril 2s
      spl i           / not blasting
      jmp sq9        / no tail
      rancf sar 9s, sar 4s, src
      scale sn, 8s, ssn
      scale cs, 8s, scn
      count i mfu, st2
      dzm i mfu
      jmp sq9
sq7,

```

```

st2,      yincr sk1, sy1, sub
          lio sy1
          xet db1
          count src, sc7
sc9,      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
          jmp sr5                  / no space for new objects

sr2,      lac (ter                / set up torpedo calc
          dac i sr1
          law nob
          add sr1
          dap ss3
          lio str
          swp
          szf 4
          add nx1
          szf i 4
          add nx1 1
          swp
ss3,      dio .
          add (nob
          dap ss4
          lio sty
          swp
          szf 4
          add ny1
          szf i 4
          add ny1 1
          swp
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
sr4, add i mdy
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, dac / 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
jmp mb1

```

/ here to handle spaceships dragged into star

/ spaceship in star

```
pof,      dzm 1 mdx
          dzm 1 mdy
          szs 50
          jmp po1
          lac (377777
          dac i mx1
          dac i my1
          lac i mb1
          dac ssn
          count ssn, .
          jmp mb1
```

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

/ outlines of spaceships

```
ot1,      111131
          111 11
          111111
          11163
          311111
          146111
          111114
          700000
```

. 5/

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

. 5/

variables  
constants

.-60/

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 scn
          dac csm
          cma
          dac csn
          dzm ssn
          dzm ssm
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 i 3
          lio 1sc
          scl 1s
          cla
          div c12
          hlt
          sza
          jmp fx1
fkr,      dio t3
          law 400
          jda fss
          law fys
          dap frt
flt,      idx t3
          cma
          dac t3
          law fus
          dap i t6
          dap i 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 i 3
          law not 1
          dap fug
          idx t5
          ral 9s
          cli
          xct db2
          isp t3
fug,      jmp i .
frt,      jmp .
fys,      law 4000
          jda fss
          law fub
          dap i t6
          dap i t7
          lac c26
          dac sx1
          lac c20
          szf i 3
          cma
          add c30
          dac sy1
          law i 2
          dac t3
          jmp fds
fub,      szf 3
          jmp . 3
          stf 3
          jmp . 2
          clf 3
          cli
          iot 11
          dio t1
          law 21
          xor t1
          sza i
          jmp fik
          law 42
          xor t1
          sza
          jmp fkg
          law a4+
          jmp fwt-1
fik,      law 4
          dap fiu
fwt,      add .
          dac t1
          isp t1
          jmp .-1
fiu,      jmp .

```



```
fx1,    dio ̄1
        dac t3
        law fx2
        dap frt
        law 1100
        jda fss
        jmp flt
fx2,    lio ̄1
        jmp fkr
```

```
fkq,    szf 3
        jmp fis
        szf 4
        jmp . 4
        law dj6
        stf 4
        jmp . 3
        law dj5
        clf 4
        dap . 1
        lac .
        dac db1
        idx .-2
        xct .-3
        dac db2
        jmp fis
```

```
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 4.4 5/17/63 ddp : pt 1 - pass 1  
spacewar 4.4 5/21/63 ddp : pt 2 - pass 1  
stars by prs for s/w 2b - pass 1  
foo  
nx1=mtb nob  
ny1=nx1 nob  
start  
foo - pass 1  
spacewar 4.4 5/17/63 ddp : pt 1 - pass 2  
spacewar 4.4 5/21/63 ddp : pt 2 - pass 2  
usw 1362 a+13 count i1  
stars by prs for s/w 2b - pass 2  
spacewar 4.4 syms 5/23/63 jcm

Constants area, inclusive  
from to  
2736 .3063

1j	6077
1m	706
1q	6117
1sc	1345
2j	6121
2m	764
2q	6141
2sc	1346
3j	6143
3m	1042
3q	6403
4j	6405
4m	1120
4q	7747
a	1347
a1	1337
a2	1405
a3	1436
a4	1370
a40	1342
a5	1364
a6	1377
acx	2727
acy	2731
bc1	667
bc3	673
bcc	1176
bck	656
bex	705
bey	704
bds	544
bjl	530
bjm	543
bjq	537
bkc	1177
blp	464
blx	511
bpt	512
bpx	645
bsg	2275
bx	2704
by	2705
c12	3407
c21	3411
c23	3412
c26	3413
c20	3410
c30	3414
col	1534
cos	100
cox	1614
cs	2732
esm	2676
csn	2701

csx	146
cwg	2720
cwr	40
db1	1331
db2	1332
ddd	20
dj5	1333
dj6	1335
fds	3272
f11	3225
f12	3230
f1d	24
f1e	16
f1k	3347
f1n	7
f1s	3233
f1u	3355
fkq	3367
fkr	3247
flo	55
flp	45
flt	3254
foo	12
fou	15
fpo	54
fpr	1200
frr	14
frr	3305
fs	4
fs1	3224
fss	3210
fub	3325
fug	3304
fus	3262
fuu	34
fwf	3351
fx	35
fx1	3356
fx2	3365
fyn	20
fys	3306
gct	2721
grv	32
hd1	23
hd2	24
hd3	25
hp1	1744
hp3	1775
hp4	2036
hp6	2053
hp7	2051
hpt	2724
hr1	26
hr2	27

hur	30
idl	1472
ids	1504
iox	2730
ke1	2102
keb	2060
ma1	1630
ma2	1631
maa	13
mb1	1517
mb2	1624
meo	2451
md1	1303
mdn	1307
mdx	2707
mdy	2710
me1	16
me2	17
mex	1632
mfu	2711
mg1	1460
mg2	1465
mg3	1464
mg4	1471
mh1	2713
mh2	2714
mh3	2715
mh4	2716
mhs	2
m11	1667
m11	1512
m12	1556
m10	1201
mom	2123
mot	2402
mq1	1522
mq2	1604
msh	1666
mst	1704
mt1	2722
mtb	3422
mtc	2706
mtb	2137
mtr	2712
mx1	1561
mx2	1562
mxo	2723
my1	1571
my2	1572
mz1	1653
na1	3532
nb1	3562
nco	3704
ndx	3612

ndy	3642
nfu	3676
nh1	3706
nh2	3710
nh3	3712
nh4	3714
nnn	3716
nob	30
nom	3672
not	3702
ntd	2717
nth	3674
ntr	3700
nx1	3452
ny1	3502
oc	226
oc1	373
oc2	416
oc3	424
oc4	432
oc5	440
oc6	446
oc9	456
occ	2672
ocd	407
oce	400
ocg	250
och	251
oci	2673
ocj	246
ock	237
ocm	371
ocn	372
oco	262
ocp	264
ocq	263
ocr	265
ocs	222
ocx	370
ocz	225
ot1	2630
ot2	2645
p	3110
po1	2623
pef	2607
ran	31
rlt	10
sac	14
sbf	61
sc1	2111
scm	2674
scn	2663
scw	2725
sex	1620

sft	66
si1	112
si2	147
si3	117
sin	106
sn	2726
sp5	2402
sq1	220
sq2	221
sq3	171
sq6	2403
sq7	2430
sq9	2446
sqt	162
sqx	174
sr1	2467
sr2	2500
sr3	2543
sr4	2547
sr5	2561
sr6	2553
sr7	2560
src	2735
sr0	2111
ss1	2103
ss2	2106
ss3	2514
ss4	2526
ssa	2702
ssc	2700
ssd	2677
ssi	2703
ssm	2675
ssn	2662
st1	2465
st2	2434
str	15
stx	2733
sty	2734
sx1	2670
sy1	2671
t1	2665
t2	2666
t3	3415
t4	3416
t5	3417
t6	3420
t7	3421
tc1	1721
tcr	1706
the	21
tlf	11
tno	6
tv1	7

v2  
xys  
xyt

3  
2664  
2667



spacewar 4.3 5/17/63 ddp . pt 1

```
3/      jmp sbf      / ignore sec. break
        jmp a40
        jmp a1      / use test word for control, not lot 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 lot 11 control
. 20/     / space
```

/ 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
          lic 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  $\overline{3s}$   
dio  $\overline{oci}$   
lio (rcl  $\overline{9s}$   
dispatch

opr  
jmp oc1

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 sq6
plinst (clf 5
plinst (lac  $\overline{scn}$ 
plinst (cma
plinst (dac  $\overline{scn}$ 
plinst (lac  $\overline{ssm}$ 
plinst (cma
plinst (dac  $\overline{ssm}$ 
plinst (lac  $\overline{csm}$ 
plinst (lio  $\overline{ssd}$ 
plinst (dac  $\overline{ssd}$ 
plinst (dio  $\overline{csm}$ 

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

ocm,   jmp .
ocn,   jmp .

oc1,   plinst (add  $\overline{ssn}$ 
      jsp ocs
      lac (sub  $\overline{scn}$ 
oc2,   dac i oc
      idx oc
      jsp ocs
      plinst (ioh  $\overline{scn}$ 
      lac (dpy-4000
oc3,   dac i oc
      idx oc
      lio  $\overline{oci}$ 
      count  $\overline{oc2}$ , ocn
      idx  $\overline{oc3}$ 
      jmp  $\overline{oc3}$ 

oc4,   comtab (add  $\overline{scm}$ , (add  $\overline{ssm}$ 
oc5,   comtab (add  $\overline{ssc}$ , (sub  $\overline{csm}$ 
oc6,   comtab (sub  $\overline{scm}$ , (sub  $\overline{ssm}$ 
oc7,   comtab (add  $\overline{csn}$ , (sub  $\overline{ssd}$ 
      szf 6
      jmp  $\overline{oc9}$ 
      stf 6
      plinst (dac  $\overline{ssa}$ 
      lac (dio  $\overline{ssi}$ 
      jmp  $\overline{ocd}$ 
oc8,   clf 6
      plinst (lac  $\overline{ssa}$ 
      lac (lio  $\overline{ssi}$ 
      jmp  $\overline{ocd}$ 

```



/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 px1  
dpy-4000

bjm,

bds,

bpx,

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

/background display • 2/13/62, 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
fio, J
terminate
```

```

define
background      jsp bck
                 termin

bck,            dap bck
                szs 40
                jmp bck
                jsp 1m
                jsp 2m
                idx bcc
                and (1
                sza
                jsp 3m
bc1,            law 3
                and bcc
                sza 1
                jsp 4m
bc3,            isp bkc
                jmp bey
                law i 20
                dac bkc
                law i 1
                add fpr
                spa
                add (20000
                dac fpr
bey,            jsp 4m
bcx,            jmp

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

bcc,            0
bkc,            0
fpr,            10000

```

mul=aus  
div=dis

start

spacewar 4.3 5/17/68 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 mx1 / 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 mdy / dy
ndy=ndx nob
          add (nob
          dap mon /angular velocity
nom=ndy nob
          add (2
          dap mth / 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
nnc=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 a2

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 fi1
        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 ml0
```



/ control word get routines

```

mg1,      dap mg3
          lot 1'
          jsp 8a
8j1,      0
          dac  $\bar{t}1$ 
          lot 111
          jsp 8a
8j2,      0
          lac  $\bar{t}1$ 
          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 .           / 1st control word
          sza i           / zero if not active
          jmp mq1        / not active
          dap . 1
          jmp .
mb1,      lac .           / alter count of number of instructions
          add  $\bar{m}tc$ 
          dac  $\bar{m}tc$ 
mq1,      idx ml1
          sad (lac mtb 1
          jmp idl
          sas (lac mtb nob
          jmp ids
background
          jsp blp        / display massive star
          count  $\bar{m}tc$ , . / use up rest of time of main loop
          jmp ml0        / repeat whole works

```

8a,       dap 8ay  
          dap 8ax  
          idx 8ax  
          clf 7  
          spi  
          stf 5  
          szf 1 5  
          stf 6  
          ril 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 mtb 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 . / no
mx2,      lac . / calc if collision
          sub . / delta x
          spa . / take abs val
          cma
          dac mt1
          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
mq2,      jmp cox
          idx mx2 / end of comparison loop
          idx my2
          idx ma2
          idx mb2
          index ml2, (lac mtb 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 m̄dx
sar 3s
add i mx1
dac i mx1
lac i m̄dy
sar 3s
add i my1
dac i my1
law mst
dap msh
lac i mb1 / time involved
cma cli-opr
sar 3s
dac m̄xc
sub (140
sma
idx msh
mz1, lac ran
and (777
ior (scl
dac mi1
random
scr 9s
sir 9s
msh, xct .
mi1, hlt
add i my1
swap
add i mx1
szs 20
jda kcb
dpy-i 300
count m̄xc, 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 (max 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
          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
          dap kc1
          swap
          sub ny1
          swap
          lac keb
          sub nx1
kc1,      jmp .

          /relocate for center display
```

```

/ 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-skp
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  $\bar{m}dy$ , my1, (sar 3s
lac  $\bar{s}n$ 
sar 9s
xct sac
cmā
szf i 6
cla
add  $\bar{b}x$ 
diff  $\bar{m}dx$ , 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  $\bar{s}tz$ 
```



```

lac i my1
szs 20
sub ny1
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
szs 20
jda kcb
dpy-4000
mot,sp5, jmp i .
sq6,    loh
        lio scw
        ril 2s
        spi i          / not blasting
        jmp sq9        / no tail
        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, sc7
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 tvl
        cma
        add i mdx
sr3,    dac .
        lac cs
        xct tvl
        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 mdx
          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 m11
          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 i 3
          lio 1sc
          scl 1s
          cla
          div c12
          hlt
          sza
          jmp fx1
fkr,      dio t3
          law 400
          jda fss
          law fys
          dap frt
flt,      idx t3
          cma
          dac t3
          law fus
          dap i t6
          dap i 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 i 3
          law not 1
          dap fug
          idx t5
          ral 9s
          cli
          dpy-4000+700
          isp t3
fug,      jmp i .
frr,      jmp .

fys,      law 4000
          jda fss
          law fub
          dap i t6
          dap i t7
          lac c26
          dac sx1
          lac c20
          szf i 3
          cma
          add c30
          dac sy1
          law i 2
          dac t3
          jmp fds
fub,      szf 3
          jmp . 3
          stf 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 51
        dac t3
        law fx2
        dap frt
        law 1100
        jde 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
bex	703
bey	702
bds	542
bjm	541
bke	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	1202
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

lco	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
sbf	61
sc1	2152
scm	2721
scn	2710

scw	2752
sex	1667
sft	66
si1	112
si2	147
si3	117
sin	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
tcr	1756
the	21
tlf	11

tno	6
tv1	7
v2	3
xys	2711
xyt	2714

spacewar 4.2 5/11/63 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
```

/ 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  $\overline{ssn}$ 
          dac Y
          lac X
          INS  $\overline{scn}$ 
          dac X
          term
```

```
          define
yincr X,Y,INS
          lac Y
          INS  $\overline{scn}$ 
          dac Y
          lac X
          -INS+add+sub  $\overline{ssn}$ 
          dac X
          terminate
```

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

```
          define
dispt A,Y,B
          repeat 6 B=B+B
          lio Y
          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
  sma
  cma
  dac C
  terminate
```



define

```
varsft
dzm  $\bar{x}$ ys
dac  $\bar{t}$ 1
idx  $\bar{x}$ ys
v2, idx  $\bar{x}$ ys
lac  $\bar{t}$ 1
scr 2s
dac  $\bar{t}$ 1
sza
jmp v2+R
scr 2s
swap
terminate
```

define

```
undosft
dac t1
dio  $\bar{t}$ 2
lac  $\bar{x}$ ys
add sft
dap .+1
lac .
dac .+6
dac .+6
xor (10000
dac  $\bar{x}$ yt
lac  $\bar{t}$ 1
dio  $\bar{t}$ 2
scr .
scr .
terminate
```

/ change scr to scl or scl to scr.

define

```
integrate A,B
cli
lac i A
scr 9s
scr 1s
div  $\bar{t}$ 1
hlt
cma+cli-opr
xct  $\bar{x}$ yt
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
si1,      add (311040
          sub (62210
          sma
          jmp si2
          add (62210

si3,      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 .

si2,      cma
          add (62210
          sma
          jmp si3
          add (62210
          spa
          jmp .+3
          sub (62210
          jmp si3

          sub (62210
          jmp si1
```

/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
          lio 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
          spa
          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  $\bar{s}x1$ 
        plinst (lio  $\bar{s}y1$ 
        clf 6
```

```
ocj,    setup  $\bar{oc}c,6$ 
ocg,    lio .                /outline table
och,    cla
```

```
        rcl  $\bar{3}s$ 
        dio  $\bar{oc}i$ 
        lio (rcl  $9s$ 
        dispatch
        opr
```

```
        jmp oc1
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 sq6
    plinst (clf 5
    plinst (lac  $\overline{scm}$ 
    plinst (cma
    plinst (dac  $\overline{scm}$ 
    plinst (lac  $\overline{ssm}$ 
    plinst (cma
    plinst (dac  $\overline{ssm}$ 
    plinst (lac  $\overline{csm}$ 
    plinst (lio  $\overline{ssd}$ 
    plinst (dac  $\overline{ssd}$ 
    plinst (dio  $\overline{csm}$ 

    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
ocd,    lac (dpy-4000
        dac i oc
        idx oc
        lio  $\overline{oci}$ 
        count  $\overline{occ}$ , och
        idx ocg
        jmp ocj

oc2,    comtab (add  $\overline{scm}$ , (add  $\overline{ssm}$ 
oc3,    comtab (add  $\overline{ssc}$ , (sub  $\overline{csm}$ 
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 a 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

blx,           jmp .

bpt,           dap bpx  
random  
sar 9s  
sar 6s  
spa  
cma  
sal 3s  
add (bds  
dap bjm  
cla cli clf 6-opr-opr  
dpy-4000

bjm,           jmp .

bds,           repeat 10, starp  
szf 6

bpx,           jmp .  
stf 6  
cma  
swap  
cma  
swap  
jmp bjm

/background display • 3/13/62, prs.

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

fin,
lac /lac X
sub fpr /right margin
sma
sub (20000
add (2000
frr,
spq
fou,
jmp fuu+R
fie,
sub (1000
sal 8s
fyn,
lio /lio Y
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 bcx
              szs 40
              jmp bcx
              jsp 1m
              jsp 2m
              idx bcc
              and (1
              sza
              jsp 3m
bc1,          law 3
              and bcc
              sza i
              jsp 4m
bc3,          isp bkc
              jmp bcy
              law i 10
              dac bkc
              law i 1
              add fpr
              spa
              add (20000
              dac fpr
bcy,          jsp 1m
bcx,          jmp

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

bcc,         0
bkc,         0
fpr,         10000

```



mul=mus  
div=dis

start

/main control routine for spaceships

```

nob=30                /total number of colliding objects

m10,    setup mtc, 5000                /delay for loop
        init m11, mtb /loc of calc routines
        add (nob
        dap mx1      / x
nx1=mtb nob
        add (nob
        dap my1      / 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 mdy      / dy
ndy=ndx nob
        add (nob
        dap mom      /angular velocity
nom=ndy nob
        add (2
        dap mth      / angle
nth=nom 2
        add (2
        dac mfu      /fuel
nfu=nth 2
        add (2
        dac mtr      / no torps remaining
ntr=nfu 2
        add (2
        dap mot      / outline of spaceship
not=ntr 2
        add (2
        dap mco      / old control word
nco=not 2
        add (2
        dac mh1
nh1=nco 2
        add (2
        dac mh2
nh2=nh1 2
        add (2
        dac mh3
nh3=nh2 2
        add (2
        dac mh4
nh4=nh3 2
nnp=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 ml1

mdn, count ntd,ml1
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

1sc,   0           /scores
2sc,   0

a,     dzn 8j1
      dzn 8j2
      lac gct
      sna
      jmp a5
      count gct, a5
      lac 1sc
      sas 2sc
      jmp a4
      law i 1
      dac gct
a5,    lat
      and (40
      sza i
      jmp a2
a4,    jmp fi1
      lat
      and (40
      sza
      jmp a2
      dzn 1sc
      dzn 2sc
a6,    lat
      rar 6s
      and (37
      sza
      cma
      dac gct
a2,    clear ntb, nnn-1      / clear out all tables
      law ss1
      dac ntb
      law ss2
      dac ntb 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 oe          / compile outline
ot1
a3, dac not 1
dap fi1
jda oe
ot2
dap fi2
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
jmp ml0
```

/ control word get routines

```
mg1,    dap mg3
        iot 11
        jsp 8a
8j1,    0
        dac 11
        iot 111
        jsp 8a
8j2,    0
        lio 11
        ril 4s
        rer 4s
mg3,    jmp .
mg2,    dap mg4
        lat
        swap
mg4,    jmp .

idl,    idx mth
        idx mfu
        idx intr
        idx oco
        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
all,    lac .           / 1st control word
        sza i           / zero if not active
        jmp mq1        / not active
        dap . 1
        jmp .
mb1,    lac .           / alter count of number of instructions
        add mtc
        dac mtc
mq1,    idx mli
        sad (lac atb 1
        jmp idl
        sas (lac atb nob
        jmp ids
        background
        jsp blp        / display massive star
        count mtc, .   / use up rest of time of main loop
        jmp mlc        / repeat whole works
```

8a,  
dap 8ay  
dap 8ax  
idx 8ax  
clf 7  
spi  
stf 5  
szf 1 5  
stf 6  
ril 1  
spi  
jmp 8ao

8ay,  
lac  
rir 8a  
spi  
cla  
rir 1s  
spi  
law 2  
dac 1 8ay

8ad,  
rir 8a  
spi  
jmp 8aq  
rir 5s  
spi  
lor (14  
rir 1s  
spi  
lor (1  
jmp 8aa

8ao,  
cla  
rir 9s  
spi  
law 2  
jmp 8ad

8aq,  
rir 5s  
spi  
lor (1  
rir 1s  
spi  
lor (14

8ar,  
rir 1s  
spi 5  
lor (4  
spi 6  
lor (10  
rir 1s  
spi 7  
lor (10  
spi 8  
lor (4

8ax,  
jmp

```

col,      dap cox
          law 1
          add ml1
          sad (lac mtb 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 . / no
mx2,      sub . / calc if collision
          spa . / delta x
          cma . / take abs val
          dac mt1
          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
mq2,      jmp cox
          idx mx2 / end of comparison loop
          idx my2
          idx ma2
          idx mb2
          index ml2, (lac mtb 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
mb2, lac i mb1 / duration of explosion
      add .
      cma
      sar 8s
      add (1
ma1, dac .
ma2, dac .
```

/ misc calculation routines

/ explosion

```
mex, lac i m̄dx
      sar 3s
      add i mx1
      dac i mx1
      lac i m̄dy
      sar 3s
      add i my1
      dac i my1
      law mst
      dap msh
      lac i mb1 / time involved
      cma cli-opr
      sar 3s
      dac m̄xc
      sub (140
      sma
      idx msh
mz1, lac ran
      and (777
      lor (sel
      dac m̄i1
      random
      ser 9s
      sir 9s
msh, xct .
m̄i1, hlt
      add i my1
      swap
      add i mx1
      dpy-i 300
      count m̄xc, mz1
      count i ma1, mb1
      dzm i ml1
      jmp mb1

mst, ser 1s
      ser 3s
```

/ torpedo calc routine

```
ter,    jsp col
        jmp sex-3
        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
        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 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,    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
hp4,    dac i mth
        lac i mth
        sma
        sub (311040
        spa
        add (311040
        dac i mth
        count hpt, hp4
        count i mh2, hp7
        dzx i mh2
hp7,    xct hd3
        dac i mh3
hp6,    lac i mx1
        dispt i, i my1, 2
        jmp mb1
```

```

/ 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
        ser 1s
        dac acx
        cla
        ser 2s
        dio iox
        lac i my1
        dac t1
        mul t1
        ser 1s
        dac acy

```

```

cla
scr 2s
swap
add  $\bar{i}ox$ 
swap
scl 2s
add  $\bar{a}cx$ 
add  $\bar{a}cy$ 
sub str
sma i sza-skp
jmp pof
add str
varsft
dac  $\bar{t}1$ 
jda set
mul  $\bar{t}1$ 
undosft
scr 9s
scr 6s
szs i 20      / switch 2 for light star
scr 2s
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  $\bar{m}dy$ , my1, (sar 3s
lac  $\bar{s}n$ 
sar 9s
xct sac
cma
szf i 6
cla
add  $\bar{b}x$ 
diff  $\bar{m}dx$ , mx1, (sar 3s
scale  $\bar{s}n$ , 5s,  $\bar{s}sn$ 
scale  $\bar{c}s$ , 5s,  $\bar{s}cn$ 
lac i mx1
sub  $\bar{s}sn$ 
dac  $\bar{s}x1$ 
sub  $\bar{s}sn$ 
dac str

```

```

lac i my1
add scn
dac sy1
add scn
dac sty
scale gn, 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,             ioh
                lio scw
                ril 2s
                spi i          / not blasting
                jmp sq9        / no tail
                ranc sar 9s, sar 4s, src
                scale gn, 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
                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 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 i mh2
       sza i
       jmp mb1
       lac scw
       and (600000
       xor (600000
       sza
       jmp mb1
       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
          lac i mb1
          dac ssr
          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
          311111
          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 scn
          dac csm
          cma
          dac csn
          dzm ssn
          dzm ssm
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
          sza
          jmp fx1
fkr,      dio t3
          law 400
          jda fss
          law fys
          dap frt
flt,      idx t3
          cma
          dac t3
          law fus
          dap i t6
          dap i 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 .
fvt,      jmp .
fys,      law 4000
          jda fss
          law fub
          dap i t6
          dap i t7
          lac c26
          dac sx1
          lac c20
          szf i 3
          cma
          add c30
          dac sy1
          law i 2
          dac t3
          jmp fds
fub,      szf 3
          jmp . 3
          stf 3
          jmp . 2
          clf 3
          iot 11
          dio  $\bar{t}1$ 
          iot 111
          dio t2
          law 1
          and  $\bar{t}1$ 
          and  $\bar{t}2$ 
          sza
          jmp fik
          law 2
          and  $\bar{t}1$ 
          and  $\bar{t}2$ 
          sza i
          jmp fis
          law a4+1
          dap fiu
          jmp fwt
fik,      law 4
          dap fiu
fwt,      add .
          dac  $\bar{t}1$ 
          isp  $\bar{t}1$ 
          jmp .-1
fiu,      jmp .

```

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

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 4.2 5/11/63 ddp : pt 1 - pass 1  
spacewar 4.2 5/16/63 ddp : pt 2 - pass 1  
stars by prs for s/w 2b - pass 1  
spacewar 4.2 5/11/63 ddp : pt 1 - pass 2  
spacewar 4.2 5/16/63 ddp : pt 2 - pass 2  
stars by prs for s/w 2b - pass 2  
spacewar 4.2 syms 5/2363 jem

Constants area, inclusive  
from to  
2714 3042

j	6077
1m	676
1q	6117
1sc	1305
2j	6121
2m	753
2q	6141
2sc	1306
3j	6143
3m	1030
3q	6403
4j	6405
4m	1105
4q	7747
8a	1502
8aa	1552
8ad	1525
8ao	1537
8aq	1544
8ax	1564
8ay	1515
8j1	1422
8j2	1426
a	1307
a1	1277
a2	1344
a3	1375
a4	1327
a40	1302
a5	1323
a6	1336
acx	2705
acy	2707
bc1	657
bc3	663
bcc	1162
bck	646
bex	675
bey	674
bds	534
bjm	533
bkc	1163
blp	466
blx	513
bpt	514
bpx	635
bsg	2277
bx	2662
by	2663
c12	3354
c21	3356
c23	3357
c26	3360
c20	3355

c30	3361
col	1565
cos	100
cox	1645
cs	2710
csm	2654
csn	2657
csx	146
cwg	2676
cwr	40
ddd	20
fds	3253
f11	3206
f12	3211
fid	23
fie	16
fik	3334
fin	7
fis	3214
fiu	3342
fkr	3230
flo	54
flp	44
flt	3235
foo	12
fou	15
fpo	53
fpr	1164
frr	14
frr	3266
fs	4
fs1	3205
fss	3171
fub	3306
fug	3265
fus	3243
fuu	33
fwf	3336
fx	34
fx1	3343
fx2	3352
fyn	20
fys	3267
gct	2677
grv	32
hd1	23
hd2	24
hd3	25
hp1	1773
hp3	2024
hp4	2062
hp6	2077
hp7	2075
hpt	2702

hr1	26
hr2	27
hur	30
idl	1440
ids	1452
iox	2706
ma1	1661
ma2	1662
maa	13
mb1	1465
mb2	1655
mco	2442
md1	1251
mdn	1255
mdx	2665
mdy	2666
me1	16
me2	17
mex	1663
mfu	2667
mg1	1417
mg2	1433
mg3	1432
mg4	1437
mh1	2671
mh2	2672
mh3	2673
mh4	2674
mhs	22
mi1	1720
m11	1460
m12	1607
m10	1165
mom	2123
mot	2373
mq1	1470
mq2	1635
msh	1717
mst	1734
mt1	2700
mtb	3367
mtc	2664
mth	2137
mtr	2670
mx1	1612
mx2	1613
mxo	2701
my1	1622
my2	1623
mz1	1704
na1	3477
nb1	3527
nco	3651
ndx	3557

ndy	3607
nfu	3643
nh1	3653
nh2	3655
nh3	3657
nh4	3661
nnn	3663
nob	30
nom	3637
not	3647
ntd	2675
nth	3641
ntr	3645
nx1	3417
ny1	3447
oc	230
oc1	375
oc2	420
oc3	426
oc4	434
oc5	442
oc6	450
oc9	460
occ	2650
ocd	411
oce	402
ocg	252
och	253
oci	2651
ocj	250
ock	241
ocm	373
ocn	374
oco	264
ocp	266
ocq	265
ocr	267
ocs	222
ocx	372
ocz	227
ot1	2606
ot2	2623
p	3071
po1	2601
pof	2565
ran	31
rlt	10
sac	14
sbf	61
sc1	2111
scm	2652
scn	2641
scw	2703
sex	1651



sft	66
si1	112
si2	147
si3	117
sin	106
sn	2704
sp5	2373
sq1	220
sq2	221
sq3	171
sq6	2374
sq7	2421
sq9	2437
sqt	162
sqx	174
sr1	2460
sr2	2472
sr3	2521
sr4	2525
sr5	2537
sr6	2531
sr7	2536
src	2713
sr0	2111
ss1	2103
ss2	2106
ss3	2500
ss4	2504
ssa	2660
ssc	2656
ssd	2655
ssi	2661
ssm	2653
ssn	2640
st1	2456
st2	2425
str	15
stx	2711
sty	2712
sx1	2646
sy1	2647
t1	2643
t2	2644
t3	3362
t4	3363
t5	3364
t6	3365
t7	3366
tc1	1751
ter	1736
the	21
tlf	11
tno	6
tv1	7

v2  
xys  
xyt

3  
2642  
2645

spacewar 4.0 2/2/63 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
```

/ 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
        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
  sma
  cma
  dac C
  terminate
```

```

define
    varsft
    dzm  $\overline{xyt}$ 
    dac  $\overline{t1}$ 
    idx  $\overline{xyt}$ 
v2,   idx  $\overline{xyt}$ 
    lac  $\overline{t1}$ 
    scr 2s
    dac  $\overline{t1}$ 
    sza
    jmp v2+R
    scr 2s
    swap
    terminate

define
    undosft
    dac t1
    dio  $\overline{t2}$ 
    lac  $\overline{xyt}$ 
    add sft
    dap .+1
    lac .
    dac .+6
    dac .+6
    xor (10000 / change scr to scl or scl to scr.
    dac  $\overline{xyt}$ 
    lac  $\overline{t1}$ 
    dio  $\overline{t2}$ 
    scr .
    scr .
    terminate

define
    integrate A,B
    cli
    lac i A
    scr 9s
    scr 1s
    div  $\overline{t1}$ 
    hlt
    cma+cli-opr
    xct  $\overline{xyt}$ 
    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
si1,      add (311040
          sub (62210
          sma
          jmp si2
          add (62210

si3,      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

csx,      lac cos
          jmp .

si2,      cma
          add (62210
          sma
          jmp si3
          add (62210
          spa
          jmp .+3
          sub (62210
          jmp si3

          sub (62210
          jmp si1
```

/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
          lio 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
          spa
          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  $\bar{s}x1$ 
        plinst (lio  $\bar{s}y1$ 
        clf 6
```

```
ocj,    setup  $\bar{o}cc,6$ 
ocg,    lio .             /outline table
och,    cla
```

```
        rcl  $\bar{3}s$ 
        dio  $\bar{o}ci$ 
        lio (rcl  $\bar{9}s$ 
        dispatch
```

```
        opr
        jmp oc1
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 sq6
    plinst (clf 5
    plinst (lac  $\overline{scm}$ 
    plinst (cma
    plinst (dac  $\overline{scm}$ 
    plinst (lac  $\overline{ssm}$ 
    plinst (cma
    plinst (dac  $\overline{ssm}$ 
    plinst (lac  $\overline{csm}$ 
    plinst (lio  $\overline{ssd}$ 
    plinst (dac  $\overline{ssd}$ 
    plinst (dio  $\overline{csm}$ 

    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
ocd,   lac (dpy-4000
        dac i oc
        idx oc
        lio  $\overline{oci}$ 
        count  $\overline{occ}$ , och
        idx ocg
        mp ocj

oc2,   comtab (add  $\overline{scm}$ , (add  $\overline{ssm}$ 
oc3,   comtab (add  $\overline{ssc}$ , (sub  $\overline{csm}$ 
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 a 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 (add 340  
spa  
xor (377777  
dac bx  
lac ran  
ral 4s  
and (add 340  
spa  
xor (377777  
dac by  
jsp bpt  
ioh  
jmp .

blx,

bpt,

dap bpx  
random  
sar 9s  
sar 5s  
spa  
cma  
sal 3s  
add (bds  
dap bjm  
cla cli clf 6-opr-opr  
dpy-4000

bjm,

bds,

jmp .  
repeat 20, starp  
szf 6

bpx,

jmp .  
stf 6  
cma  
swap  
cma  
swap  
jmp bjm

/background display • 3/13/62, prs.

```
define
dislis J, Q, B
  repeat 6, B=B+B
  clf 5
  lac flo+R
  dap fpo+R
fs,
  dap fin+R
  dap fyn+R
  idx fyn+R

fin,
  lac          /lac X
  sub fpr      /right margin
  sma
  jmp fgr+R
  add (2000

frr,
  spq
fou,
  jmp fuu+R
fie,
  sub (1000
  sal 8s
fyn,
  lio          /lio Y
  dpy-i+B
  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

fgr,
  add (-20000+2000
  jmp frr+R

fuu,
  szf 5
fx,
  jmp flo+R+1 /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 bcx
            szs 40
            jmp bcx
            isp bcc
bcx,         jmp .
            law i 2
            dac bcc
            dislis 1j,1q,3
            dislis 2j,2q,2
            dislis 3j,3q,1
            dislis 4j,4q,0
            isp bkc
            jmp bcx
            law i 20
            dac bkc
            law i 1
            add fpr
            spa
            add (20000
            dac fpr
            jmp bcx

bcc,         0
bkc,         0
fpr,         10000

```

```

mul=mus
div=dis

```

```

start

```

spacewar 4.0 ddp 2/2/63 pt.2

/main control routine for spaceships

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

m10, load mtc, -4000 /delay for loop
      init m11, mtb /loc of calc routines
      add (nob
      dap mx1 / x
nx1=mtb nob
      add (nob
      dap my1 / 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 mdy / dy
ndy=ndx nob
      add (nob
      dap mom /angular velocity
nom=ndy nob
      add (2
      dap mth / angle
nth=nom 2
      add (2
      dac mfu /fuel
nfu=nth 2
      add (2
      dac mtr / no torps remaining
ntr=nfu 2
      add (2
      dap mot / outline of spaceship
not=ntr 2
      add (2
      dap mco / old control word
nco=not 2
      add (2
      dac mh1
nh1=nco 2
      add (2
      dac mh2
nh2=nh1 2
      add (2
      dac mh3
nh3=nh2 2
      add (2
      dac mh4
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 i
jmp mdn
md1,          xct tlf          / restart delay is 2X torpedo life
sal 1s
dac  $\bar{n}td$ 
jmp m11

mdn,          count  $\bar{n}td, m11$ 
stf 1
stf 2
law ss1
xor mtb
sza
clf 1
sza i
idx  $\bar{1}sc$ 
law ss2
xor mtb 1
sza
clf 2
sza i
idx  $\bar{2}sc$ 
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 zsc
         jmp a4
         law i 1
         dac gct

a5,     lat
         and (40
         sza i
         jmp a2

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

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 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
jmp ml0
```

/ control word get routines

```
mg1,    dap mg3
        cli
        iot 11
mg3,    jmp .

mg2,    dap mg4
        lat
        swap
mg4,    jmp .

m11,    lac .                / 1st control word
        sza i                / zero if not active
        jmp mq1              / not active
        swap
        idx mt1
        spi
        jmp mq4
        law 1
        add m11
        dap m12
        law 1
        add mx1
        dap mx2
        law 1
        add my1
        dap my2
        law 1
        add ma1
        dap ma2
        law 1
        add mb1
        dap mb2

mot,    lac .
        dap sp5

m12,    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 mt1
        sub me1              / < EPSILON ?
        sma
        jmp mq2              / no

my1,    lac .
my2,    sub .
        spa
        cma
        sub me1              / < epsilon ?
        sma
        jmp mq2              / no
        add mt1
        sub me2
        sma
```

```
      jmp mq2
      lac (mex 400000 / yes, EXPLODE
      dac i m11 / replace calc routine with explosion
      dac i m12
      lac i mb1 / duration of explosion
mb2,  add .
      cma
      sar 8s
      add (1
ma1,  dac .
ma2,  dac .
mq2,  idx mx2 / end of comparison loop
      idx my2
      idx ma2
      idx mb2
      index m12, (lac mtb nob, m12
```

```

mq4,   lac i ml1      / routine for calculating spaceship
      dap . 1        / or other object and displaying it
      jsp .
mb1,   lac .          / alter count of number of instructions
      add mtc
      dac mtc
mq1,   idx mx1        / end of comparison and display loop
      idx my1
      idx ma1
      idx mb1
      idx mdx
      idx mdy
      idx mom
      idx mth
      idx mas
      idx mfu
      idx mtr
      idx mot
      idx mco
      idx mh1
      idx mh2
      idx mh3
      idx mh4
      index ml1, (lac mtb nob-1, ml1
      lac i ml1      / display and compute last point
      sza i          / if active
      jmp mq3
      dap . 1
      jsp .
      lac i mb1
      add mtc
      dac mtc
mq3,   background    / display stars of the heavens
      jsp blp        / display massive star
      count mtc, .  / use up rest of time of main loop
      jmp ml0        / repeat whole works

```

/ misc calculation routines

/ explosion

```
mex,    dap mxr
        lac i m̄dx
        sar 3s
        add i mx1
        dac i m̄x1
        lac i m̄dy
        sar 3s
        add i my1
        dac i my1
        law mst
        dap msh
        lac i mb1    / time involved
        cma cli-opr
        sar 3s
        dac m̄xc
ms1,    sub (140
        sma
        idx msh
mz1,    random
        and (777
        ior (scl
        dac mi1
        random
        scr 9s
        sir 9s
msh,    xct .
mi1,    hlt
        add i my1
        swap
        add i mx1
        dpy-i 300
        count m̄xc, mz1
        count i ma1, mxr
        dzm i ml1
mxr,    jmp .
mst,    scr 1s
        scr 3s
```

/ torpedo calc routine

```
tcr,    dap trc
        count i ma1, tc1
        lac (mex 400000
        dac i ml1
        law i 2
        dac i ma1
        jmp trc
tc1,    lac i mx1
        sar 9s
        xct the
        add i m̄dy
        dac i m̄dy
```

```
    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 i, i my1, 1
trc, jmp .
```

/ hyperspace routines

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

```
hp1,    dap hp2
        count i ma1, hp2
        law hp3          / next step
        dac i m11
        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
        random
        scr 9s
        sir 9s
        xct hr2
        dac i mdy
        dio i mdx
        setup hpt,3
        lac ran
hp4,    dac i mth
        lac i mth
        sma
        sub (311040
        spa
        add (311040
        dac i mth
        count hpt, hp4
        xct hd2
        dac i ma1
hp2,    jmp .
```

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

```
hp3,    dap hp5
        count i ma1, hp6
        lac i mh1
        dac i m11
        law 2000
        dac i mb1
        count i mh2, hp7
        dzm i mh2
```

```
hp7,    xct hd3
        dac i mh3
        lac i mh4
        add hur
        dac i mh4
        random
        ior (400000
        add i mh4
        spa
        jmp hp5
        lac (mex 400000
        dac i ml1
        law i 10
        dac i ma1
        law 2000
        dac i mb1
hp6,    lac i mx1
        dispt i, i my1, 2
hp5,    jmp .
```



/ spaceship calc

ss1,     dap srt                     / first spaceship  
        jsp i cwg  
        dio scw  
        jmp sr0

ss2,     dap srt                     / second spaceship  
        jsp i cwg  
        rir 4s  
        dio scw

sr0,

sc1,     lio scw                   /control word  
        clf 6 cla-opr               /update angle

        spi  
        add maa  
        ril 1s  
        spi  
        sub maa

mom,     add .  
        dac i mom  
        szs 10  
        jmp sr8  
        dzm i mom

sr8,     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 iox  
        lac i my1  
        dac t1  
        mul t1  
        scr 1s  
        dac acy

```

cla
scr 2s
swap
add  $\bar{t}_{ox}$ 
swap
scl 2s
add  $\bar{a}_{cx}$ 
add  $\bar{a}_{cy}$ 
sub str
sma i sza-skp
jmp pof
add str
varsft
dac  $\bar{t}_1$ 
jda sqt
mul  $\bar{t}_1$ 
undosft
scr 9s
scr 6s
szs i 20          / switch 2 for light star
scr 2s
sza
jmp bsg
scr 1s
dio  $\bar{t}_1$ 
integrate mx1,  $\bar{b}_x$ 
integrate my1,  $\bar{b}_y$ 
bsg,
cla
sad i mfu
clf 6
lac i mth
jda  $\bar{c}_{os}$ 
dac  $\bar{c}_s$ 
sar 9s
xct sac
szf i 6
cla
add  $\bar{b}_y$ 
diff  $\bar{m}_{dy}$ , my1, (sar 3s
lac  $\bar{s}_n$ 
sar 9s
xct sac
cma
szf i 6
cla
add  $\bar{b}_x$ 
diff  $\bar{m}_{dx}$ , mx1, (sar 3s
sp1,
scale  $\bar{s}_n$ , 5s,  $\bar{s}_{sn}$ 
sp2,
scale  $\bar{c}_s$ , 5s,  $\bar{s}_{cn}$ 
lac i mx1

```

```

sub  $\bar{s}sn$ 
dac  $\bar{s}x1$ 
sub  $\bar{s}sn$ 
dac  $\bar{s}tx$ 
lac i my1
add  $\bar{s}cn$ 
dac  $\bar{s}y1$ 
add  $\bar{s}cn$ 
dac  $\bar{s}ty$ 
scale  $\bar{s}n$ , 9s,  $\bar{s}sn$ 
scale  $\bar{c}s$ , 9s,  $\bar{s}cn$ 
lac  $\bar{s}sn$ 
dac  $\bar{s}sm$ 
add  $\bar{s}cn$ 
dac  $\bar{s}sc$ 
dac  $\bar{s}sd$ 
lac  $\bar{s}sn$ 
sub  $\bar{s}cn$ 
dac  $\bar{c}sn$ 
cma
dac  $\bar{c}sm$ 
lac  $\bar{s}cn$ 
dac  $\bar{s}cm$ 
cla cli-opr
dpy-4000
sp5, jmp .
sq6, ioh
ranch sar 9s, sar 4s,  $\bar{s}rc$ 
lio  $\bar{s}cw$ 
ril 2s
spi i / not blasting
jmp sq9 / no tail
sq7, scale  $\bar{s}n$ , 8s,  $\bar{s}sn$ 
scale  $\bar{c}s$ , 8s,  $\bar{s}cn$ 
count i mfu, st2
dzm i mfu
jmp sq9

st2, yincr  $\bar{s}x1$ ,  $\bar{s}y1$ , sub
dispt i,  $\bar{s}y1$ 
count  $\bar{s}rc$ , sq7
sq9, count i ma1, sr5 / check if torp tube reloaded
dzm i ma1 / prevent count around
mco, lac . / previous contro word
cma
szs i 30
clc
and  $\bar{s}cw$  / present control word
ral 3s / torpedo bit to bit 0
sma
jmp sr5 / no launch
count i  $\bar{m}tr$ , st1 / check if torpedos exhausted
dzm i  $\bar{m}tr$  / prevent count around
jmp sr5
st1, init sr1, mtb / search for unused object
sr1, lac .
sza i / 0 if unused
jmp sr2
index sr1, (lac mtb nob, sr1
hlt / no space for new objects
jmp .-1

```

```

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
trf,    xct tlf      / life of torpedo
sr6,    dac .
        law 20
sr7,    dap .
sr5,    count i mh3, st3      / length of torp calc.
        / hyperbutton active?
        dzm i mh3
        lac i mh2
        sza i
        jmp st3
        lac scw
        cma
        ior i mco
        and (600000
        sza
        jmp st3
        lac i m11
        dac i mh1
        lac (hp1 400000
        dac i m11
        xct hd1
        dac i ma1
        law 3
        dac i mb1

st3,
st, jmp . -> LAC T MCO
          LAC 3 LW
          DAC T MCO
          -> JMP .

```

/ here to handle spaceships dragged into star

/ spaceship in star

```
pof,      dzm i  $\overline{m}dx$ 
          dzm i  $\overline{m}dy$ 
          szs 50
          jmp po1
          lac (377777
          dac i mx1
          dac i my1
          lac i mb1
          dac  $\overline{s}sn$ 
          count  $\overline{s}sn, .$ 
          jmp srt
```

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

/ outlines of spaceships

ot1, 111131  
111111  
111111  
111163  
311111  
146111  
111114  
700000

. 5/

ot2, 013113  
113111  
116313  
131111  
161151  
111633  
365114  
700000

. 5/

constants  
variables  
p, . 200/ / space for patches

mtb, / table of objects and their properties

spacewar game saver patch

mg1=1510

ran=31

/+ tw → punch

/- tw → read

40/

cwr,      dap cwx  
          lat

cw2,      cks  
          ril 1s  
          spi+spa i-skp  
          jmp cw1  
          rrb

cw3,      rpa-i  
          rir 4s  
cwx,      jmp .

cw1,      ril 3s  
          spi+sma i-skp  
          jmp cw2  
          jsp mg1  
          ril 4s  
          ppa-i  
          jmp cw3

6000/

/new starting address

go,      lat  
          sma  
          jmp pu  
          rpb  
          dio ran  
          rpa-i  
          jmp 4

pu,      law i 200  
          dac  $\overline{pc}$   
          cli  
          ppa  
          isp  $\overline{pc}$   
          jmp  $\overline{pc}$   
          lio ran  
          ppb           ril 6s  
          ppb           ril 6s  
          ppb-i  
          jmp 4

variables

start go

1j	6077
1q	6117
1sc	3151
2j	6121
2	6141
2sc	3152
3j	6143
3q	6403
4j	6405
4q	7747
a	1402
a1	1374
a2	1437
a3	1470
a4	1420
a40	1377
a5	1414
a6	1431
acx	3162
acy	3164
bcc	1257
bck	746
bcx	752
bds	534
bjm	533
bkc	1260
blp	466
blx	513
bpt	514
bpx	735
bsg	2317
bx	3135
by	3136
cos	100
cs	3165
csm	3127
csn	3132
csx	146
cwg	3153
cwr	40
ddd	20
fgr	32
fid	22
fie	15
fin	6
flo	55
flp	45
foo	12
fou	14
fpo	54
fpr	1261
frr	13
fs	3
fuu	34



fx	35
fyn	17
gct	3154
grv	32
hd1	23
hd2	24
hd3	25
hp1	2001
hp2	2061
hp3	2062
hp4	2051
hp5	2123
hp6	2120
hp7	2074
hr1	26
hr2	27
hur	30
iox	3163
ma1	1610
ma2	1611
maa	13
mb1	1624
mb2	1604
mco	2466
md1	1346
mdn	1352
mdx	3140
mdy	3141
me1	16
me2	17
mex	1667
mfu	3142
mg1	1510
mg2	1514
mg3	1513
mg4	1520
mh1	3144
mh2	3145
mh3	3146
mh4	3147
mhs	22
mi1	1731
ml1	1521
ml2	1552
ml0	1262
moc	3155
mom	2143
mot	1550
mq1	1627
mq2	1612
mq3	1662
mq4	1621
ms1	1706
msh	1730

mst	1745
mt1	3156
mtb	3371
mtc	3137
mth	2157
mtr	3143
mx1	1555
mx2	1556
mxo	3157
mxr	1744
my1	1565
my2	1566
mz1	1711
na1	3501
nb1	3531
nco	3653
ndx	3561
ndy	3611
nfu	3645
nh1	3655
nh2	3657
nh3	3661
nh4	3663
nnn	3665
nob	30
nom	3641
not	3651
ntd	3150
nth	3643
ntr	3647
nx1	3421
ny1	3451
oc	230
oc1	375
oc2	420
oc3	426
oc4	434
oc5	442
oc6	450
oc9	460
occ	3123
ocd	411
oce	402
ocg	252
och	253
oci	3124
ocj	250
ock	241
ocm	373
ocn	374
oco	264
ocp	266
ocq	265
ocr	267

ocs	222
ocx	372
ocz	227
ot1	2626
ot2	2643
p	3171
po1	2621
pof	2605
ran	31
rlt	10
sac	14
sbf	61
sc1	2134
scm	3125
scn	3114
scw	3160
sft	66
si1	112
si2	147
si3	117
sin	106
sn	3161
sp1	2353
sp2	2356
sp5	2417
sq1	220
sq2	221
sq3	171
sq6	2420
sq7	2437
sq9	2463
sqt	162
sqx	174
sr1	2504
sr2	2514
sr3	2543
sr4	2547
sr5	2556
sr6	2553
sr7	2555
sr8	2151
src	3170
sr0	2134
srt	2604
ss1	2124
ss2	2130
ss3	2522
ss4	2526
ssa	3133
ssc	3131
ssd	3130
ssi	3134
ssm	3126
ssn	3113

st1	2502
st2	2451
st3	2602
str	15
stx	3166
sty	3167
sx1	3121
sy1	3122
t1	3116
t2	3117
tc1	1757
ter	1747
the	21
tlf	11
tno	6
trc	2000
trf	2552
tv1	7
v2	3
xys	3115
xyt	3120