

J E S 2 J O B L O G

21.17.20 JOB 13 \$HASP373 PRIMFORH STARTED - INIT 1 - CLASS A - SYS TK4-
21.17.20 JOB 13 IEF403I PRIMFORH - STARTED - TIME=21.17.20
21.17.20 JOB 13 IEFACTRT - Stepname Procstep Program Retcode
1 21.17.20 JOB 13 PRIMFORH PRIMES FORT IEKAA00 RC= 0000
2 21.17.21 JOB 13 PRIMFORH PRIMES GO LOADER RC= 0000
3 21.17.21 JOB 13 IEF404I PRIMFORH - ENDED - TIME=21.17.21
4 21.17.21 JOB 13 \$HASP395 PRIMFORH ENDED

5
6
7 ----- JES2 JOB STATISTICS -----
8
9
10 26 NOV 21 JOB EXECUTION DATE
11
12
13 204 CARDS READ
14
15
16 339 SYSOUT PRINT RECORDS
17
18
19 0 SYSOUT PUNCH RECORDS
20
21
22 0.00 MINUTES EXECUTION TIME
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80

1412THE

1

1 //PRIMFORH JOB (FORTRAN),
// 'Eratosthenes Sieve',
// CLASS=A,
// MSGCLASS=A,
// REGION=9000K, TIME=1440,
// MSGLEVEL=(1,1),

JOB 13

1 // USER=HERC01,PASSWORD= GENERATED BY GDL

2 ***
3 *** Name: SYS2.JCLLIB(PRIMFORH)
4 ***
5 *** Desc: Sieve of Eratosthenes programmed in FORTRAN,
6 *** compiled using the IBM OS/360 FORTRAN H Level 21.8 compiler.
7 *** All prime numbers up to the value entered via
8 *** //GO.SYSIN DD are computed.
9 ***

10 ***
11 *****

12 2 //PRIMES EXEC FORTHCLD,PARM.GO='SIZE=9000000'
13 3 XXFORTHCLD PROC SOUT='*' 00000100
14 4 XXFORT EXEC PGM=IEKAA00,PARM=(SOURCE,MAP) 00000200
15 5 XXSYSPRINT DD SYSOUT=&SOUT 00000300
16 6 XXSYSPUNCH DD SYSOUT=B 00000400
17 7 XXSYSLIN DD DSNAME=&LOADSET,UNIT=SYSDA,DISP=(MOD,PASS), 00000500
18 XX SPACE=(400,(200,50),RLSE) 00000600
19 8 //FORT.SYSIN DD *
20 9 XXGO EXEC PGM=LOADER,PARM=(MAP),COND=(4,LT,FORT) 00000700
21 10 XXSYSLIB DD DSNAME=SYS1.FORTLIB,DISP=SHR 00000800
22 11 XXSYSLOUT DD SYSOUT=&SOUT 00000900
23 12 XXSYSLIN DD DSNAME=*.FORT.SYSLIN,DISP=(OLD,PASS) 00001000
24 13 XXFT05F001 DD DDNAME=SYSIN 00001100
25 14 //GO.FT06F001 DD SYSOUT=*,DCB=(RECFM=FBA,LRECL=166,BLKSIZE=16600)
26 X/FT06F001 DD SYSOUT=&SOUT 00001200
27 15 XXFT07F001 DD SYSOUT=B 00001300
28 16 //GO.SYSIN DD *

1412THE

1

STMT NO. MESSAGE

5 IEF653I SUBSTITUTION JCL - SYSOUT=*

11 IEF653I SUBSTITUTION JCL - SYSOUT=*

14 IEF653I SUBSTITUTION JCL - SYSOUT=*

1 IEF236I ALLOC. FOR PRIMFORH FORT PRIMES
 2 IEF237I JES2 ALLOCATED TO SYSPRINT
 3 IEF237I JES2 ALLOCATED TO SYSPUNCH
 4 IEF237I 180 ALLOCATED TO SYSLIN
 5 IEF237I JES2 ALLOCATED TO SYSIN
 6 IEF142I PRIMFORH FORT PRIMES - STEP WAS EXECUTED - COND CODE 0000
 7 IEF285I JES2.JOB00013.S00103 SYSOUT
 8 IEF285I JES2.JOB00013.S00104 SYSOUT
 9 IEF285I SYS21330.T211720.RA000.PRIMFORH.LOADSET PASSED *-----39
 10 IEF285I VOL SER NOS= WORK02.
 11 IEF285I JES2.JOB00013.SI0101 SYSIN
 12 IEF373I STEP /FORT / START 21330.2117
 13 IEF374I STEP /FORT / STOP 21330.2117 CPU 0MIN 00.05SEC SRB 0MIN 00.01SEC VIRT 9064K SYS 232K
 14
 15 ****
 16 * 1. Jobstep of job: PRIMFORH Stepname: FORT Program name: IEKAA00 Executed on 26.11.21 from 21.17.20 to 21.17.20 *
 17 * elapsed time 00:00:00,12 CPU-Identifier: TK4- Page-in: 0
 18 * CPU time 00:00:00,06 Virtual Storage used: 9064K Page-out: 0
 19 * corr. CPU: 00:00:00,06 CPU time has been corrected by 1 / 1,0 multiplier
 20 *
 21 * I/O Operation
 22 * Number of records read via DD * or DD DATA: 180
 23 * DMY.....0 DMY.....0 180.....39 DMY.....0
 24 *
 25 * Charge for step (w/o SYSOUT): 0,10
 26 ****
 27 IEF236I ALLOC. FOR PRIMFORH GO PRIMES
 28 IEF237I 148 ALLOCATED TO SYSLIB
 29 IEF237I JES2 ALLOCATED TO SYSLOUT
 30 IEF237I 180 ALLOCATED TO SYSLIN
 31 IEF237I JES2 ALLOCATED TO FT05F001
 32 IEF237I JES2 ALLOCATED TO FT06F001
 33 IEF237I JES2 ALLOCATED TO FT07F001
 34 IEF142I PRIMFORH GO PRIMES - STEP WAS EXECUTED - COND CODE 0000
 35 IEF285I SYS1.FORTLIB KEPT *-----65
 36 IEF285I VOL SER NOS= MVSRES.
 37 IEF285I JES2.JOB00013.S00105 SYSOUT
 38 IEF285I SYS21330.T211720.RA000.PRIMFORH.LOADSET PASSED *-----40
 39 IEF285I VOL SER NOS= WORK02.
 40 IEF285I JES2.JOB00013.SI0102 SYSIN
 41 IEF285I JES2.JOB00013.S00106 SYSOUT
 42 IEF285I JES2.JOB00013.S00107 SYSOUT
 43 IEF373I STEP /GO / START 21330.2117
 44 IEF374I STEP /GO / STOP 21330.2117 CPU 0MIN 00.04SEC SRB 0MIN 00.01SEC VIRT 8824K SYS 224K
 45
 46 ****
 47 * 2. Jobstep of job: PRIMFORH Stepname: GO Program name: LOADER Executed on 26.11.21 from 21.17.20 to 21.17.21 *
 48 * elapsed time 00:00:00,10 CPU-Identifier: TK4- Page-in: 0
 49 * CPU time 00:00:00,05 Virtual Storage used: 8824K Page-out: 0
 50 * corr. CPU: 00:00:00,05 CPU time has been corrected by 1 / 1,0 multiplier
 51 *
 52 * I/O Operation
 53 * Number of records read via DD * or DD DATA: 1
 54 * 148.....65 DMY.....0 180.....40 DMY.....0 DMY.....0
 55 *
 56 * Charge for step (w/o SYSOUT): 0,08
 57 ****
 58 IEF237I 180 ALLOCATED TO SYS00001
 59 IEF285I SYS21330.T211721.RA000.PRIMFORH.R0000001 KEPT *-----0
 60 IEF285I VOL SER NOS= WORK02.

IEF285I SYS21330.T211720.RA000.PRIMFORH.LOADSET

DELETED

IEF285I VOL SER NOS= WORK02.

IEF375I JOB /PRIMFORH/ START 21330.2117

IEF376I JOB /PRIMFORH/ STOP 21330.2117 CPU 0MIN 00.09SEC SRB 0MIN 00.02SEC

1412THE

1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
10	10
11	11
12	12
13	13
14	14
15	15
16	16
17	17
18	18
19	19
20	20
21	21
22	22
23	23
24	24
25	25
26	26
27	27
28	28
29	29
30	30
31	31
32	32
33	33
34	34
35	35
36	36
37	37
38	38
39	39
40	40
41	41
42	42
43	43
44	44
45	45
46	46
47	47
48	48
49	49
50	50
51	51
52	52
53	53
54	54
55	55
56	56
57	57
58	58
59	59
60	60

COMPILER OPTIONS - NAME= MAIN,OPT=00,LINECNT=50,SIZE=0000K,
 SOURCE,EBCDIC,NOLIST,NOECK,LOAD,MAP,NOEDIT,NOID,NOXREF

```

C /*-----*/
C /* Sieve of Eratosthenes. */
C /*-----*/
C /*-----*/
C /* Formats for output. */
C /*-----*/
ISN 0002 3 FORMAT (' ')
ISN 0003 4 FORMAT ( 3X, I7, 1X, I7, 1X, I7, 1X, I7, 1X, I7, 1X, I7, 1X, I7,
11X, I7, 1X, I7, 1X,
1, I7, 1X, I7, 1X, I7)
ISN 0004 5 FORMAT (' Sieve of Eratosthenes generated using OS/360 FORTRAN H L
1evel 21.8')
ISN 0005 6 FORMAT (' Upper limit of test range = ', I12)
ISN 0006 7 FORMAT (' Number of primes in range = ', I12)
ISN 0007 8 FORMAT (I7)
ISN 0008 993 FORMAT (' DEBUG: marking as non prime: ', I4)
ISN 0009 994 FORMAT (' DEBUG: starting p=', I4, ' j=', I4)
C
C /*-----*/
C /* Define array of flags, one for each integer in the range */
C /* we will test. If the flag is on, the corresponding */
C /* number is prime. If it's off, the number is not prime. */
C /* We will initialize all the flags to on (assuming every */
C /* number is prime) and turn them off as we determine the */
C /* corresponding number is not prime. */
C /*-----*/
ISN 0010 LOGICAL*1 FLAGS(5000002)
C
C /*-----*/
C /* The PRIME array will hold all the prime numbers we have */
C /* identified, and CPRIME will contain the number of primes */
C /* we've found. */
C /*-----*/
ISN 0011 INTEGER*4 PRIME(350000)
ISN 0012 INTEGER*4 CPRIME
C
C /*-----*/
C /* J is a loop counter and work variable. */
C /*-----*/
ISN 0013 INTEGER*4 J
C
C /*-----*/
C /* K is the step amount for crossing out prime multiples */
C /*-----*/
ISN 0014 INTEGER*4 K

```

```

C
C /*-----*/
C /* P is the number that we've most recently determined */
C /* definitely to be prime. */
C /*-----*/
1 ISN 0015 INTEGER*4 P
2
3 C
4 C /*-----*/
5 C /* The DEBUG flag is set to TRUE if debugging messages are to */
6 C /* be issued and FALSE otherwise. */
7 C /*-----*/
8 ISN 0016 LOGICAL*1 DEBUG
9 C
10 C /*-----*/
11 C /* LIMIT sets the upper bound of the range of numbers we */
12 C /* will test. */
13 C /*-----*/
14 ISN 0017 INTEGER*4 LIMIT
15 C
16 C /*-----*/
17 C /* REPEAT is the number of times that the entire prime */
18 C /* generation process is to be repeated, and is useful */
19 C /* for benchmarking (otherwise it should be 1). */
20 C /*-----*/
21 ISN 0018 INTEGER*4 REPEAT
22 C
23 C /*-----*/
24 C /* Initialize LIMIT, DEBUG and REPEAT. */
25 C /*-----*/
26 ISN 0019 READ (5, 8) LIMIT
27 C DEBUG = .TRUE.
28 ISN 0020 DEBUG = .FALSE.
29 ISN 0021 REPEAT = 1
30 C
31 C *-----*
32 C * THIS IS THE TOP OF THE LOOP FOR BENCHMARK TESTING.
33 C *-----*
34 ISN 0022 100 CONTINUE
35 ISN 0023 REPEAT = REPEAT - 1
36 ISN 0024 IF (REPEAT .LT. 0) GO TO 999
37 C
38 C /*-----*/
39 C /* Initialize all flags to on. We optimistically assume */
40 C /* all numbers are prime, and will subsequently turn flags */
41 C /* off as reality sets in. */
42 C /*-----*/
43 ISN 0026 DO 200 J = 3, LIMIT, 2
44 ISN 0027 FLAGS(J) = .TRUE.
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80

```

ISN 0028 200 CONTINUE

```

C
C /*-----*/
C /* The first prime number is 3, the 2 is handled manually */
C /*-----*/

```

ISN 0029 P = 3

```

C
C /*-----*/
C /* Start of the main loop. P is the prime number we're
C /* currently working on. If P*P is greater than the limit
C /* value, we're done (all the numbers between P and the limit
C /* inclusive have already been marked appropriately). Any
C /* non-prime less than P*P has also already been marked
C /* appropriately, so we will start this pass marking with
C /* P*P (which we will call J).
C /*-----*/

```

ISN 0030 300 CONTINUE

ISN 0031 J = P * P

ISN 0032 K = 2 * P

ISN 0033 IF (J .GE. LIMIT) GO TO 700

ISN 0035 IF (.NOT. DEBUG) GO TO 400

ISN 0037 WRITE (6, 994) P, J

```

C
C /*-----*/
C /* By definition, all multiples of prime number P are not
C /* prime. Turn off the flags for the multiples of P to
C /* mark them as non-prime. Note: Even numbers are skipped.
C /*-----*/

```

ISN 0038 400 CONTINUE

ISN 0039 IF (J .GT. LIMIT) GO TO 500

ISN 0041 IF (.NOT. DEBUG) GO TO 420

ISN 0043 WRITE (6, 993) J

ISN 0044 420 FLAGS(J) = .FALSE.

ISN 0045 J = J + K

ISN 0046 GO TO 400

```

C
C /*-----*/
C /* Done marking all multiples of J as not prime. Find the
C /* next prime number after J, set it to P and loop back to
C /* process it. Note: Even numbers are skipped.
C /*-----*/

```

ISN 0047 500 CONTINUE

ISN 0048 P = P + 2

ISN 0049 IF (FLAGS(P)) GO TO 600

ISN 0051 GO TO 500

```

C
C /*-----*/
C /* Bottom of the main loop.
C /*-----*/

```

```

ISN 0052      C /*-----*/
600    CONTINUE
ISN 0053      GO TO 300
C
1   C /*-----*/                                1
2   C /* Bottom of the benchmark loop. */      2
3   C /*-----*/                                3
4   ISN 0054    700    CONTINUE                 4
5   ISN 0055    GO TO 100                      5
6   ISN 0056    999    CONTINUE                 6
7   C
8   C /*-----*/                                7
9   C /* Set the prime numbers we have found in the PRIME array. */  8
10  C /*-----*/                               9
11  ISN 0057    CPRIME = 1                     10
12  ISN 0058    PRIME(CPRIME) = 2               11
13  ISN 0059    DO 800 J = 3, LIMIT, 2         12
14  ISN 0060    IF (.NOT. FLAGS(J)) GO TO 800  13
15  ISN 0062    CPRIME = CPRIME + 1             14
16  ISN 0063    PRIME(CPRIME) = J              15
17  ISN 0064    800    CONTINUE                 16
18  C
19  C /*-----*/                                17
20  C /* Display the results. */                18
21  C /*-----*/                               19
22  ISN 0065    WRITE (6, 3)                   20
23  ISN 0066    WRITE (6, 5)                   21
24  ISN 0067    WRITE (6, 6) LIMIT              22
25  ISN 0068    WRITE (6, 7) CPRIME             23
26  ISN 0069    WRITE (6, 3)                   24
27  ISN 0070    WRITE (6, 4) (PRIME(J), J = 1, CPRIME) 25
28  ISN 0071    WRITE (6, 3)                   26
29  C
30  C /*-----*/                                27
31  C /* End of program. */                    28
32  C /*-----*/                               29
33  ISN 0072    STOP                         30
34  ISN 0073    END                          31

```

14121HE

1

100 61AA6C
420 61AB48
999 61AB98

200 61AA92
500 61AB66
800 61ABEC

300 61AAB0
600 61AB8C

400 61AB0C
700 61AB92

1 *OPTIONS IN EFFECT* NAME= MAIN,OPT=00,LINECNT=50,SIZE=0000K,
2
3 *OPTIONS IN EFFECT* SOURCE,EBCDIC,NOLIST,NOECK,LOAD,MAP,NOEDIT,NOID,NOXREF
4
5 *STATISTICS* SOURCE STATEMENTS = 72 ,PROGRAM SIZE = 6401286
6
7 *STATISTICS* NO DIAGNOSTICS GENERATED
8
9
10 ***** END OF COMPIRATION ***** 8917K BYTES OF CORE NOT USED
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80

VS LOADER

OPTIONS USED - PRINT, NOMAP, NOLET, CALL, RES, NOTERM, SIZE=9000000, NAME=**GO

TOTAL LENGTH 61FFD0
ENTRY ADDRESS ACBD0

1412THE

1

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60

Sieve of Eratosthenes generated using OS/360 FORTRAN H Level 21.8

Upper limit of test range = 2000

Number of primes in range = 303

1412 THE

1	2	3	5	7	11	13	17	19	23	29	31	37	41	43	47	53	1				
2	67	71	73	79	83	89	97	101	103	107	109	113	127	131	137	139	2				
3	157	163	167	173	179	181	191	193	197	199	211	223	227	229	233	239	3				
4	257	263	269	271	277	281	283	293	307	311	313	317	331	337	347	349	4				
5	367	373	379	383	389	397	401	409	419	421	431	433	439	443	449	457	5				
6	467	479	487	491	499	503	509	521	523	541	547	557	563	569	571	577	6				
7	599	601	607	613	617	619	631	641	643	647	653	659	661	673	677	683	9				
8	709	719	727	733	739	743	751	757	761	769	773	787	797	809	811	821	10				
9	829	839	853	857	859	863	877	881	883	887	907	911	919	929	937	941	11				
10	967	971	977	983	991	997	1009	1013	1019	1021	1031	1033	1039	1049	1051	1061	13				
11	1087	1091	1093	1097	1103	1109	1117	1123	1129	1151	1153	1163	1171	1181	1187	1193	14				
12	1217	1223	1229	1231	1237	1249	1259	1277	1279	1283	1289	1291	1297	1301	1303	1307	15				
13	1327	1361	1367	1373	1381	1399	1409	1423	1427	1429	1433	1439	1447	1451	1453	1459	17				
14	1483	1487	1489	1493	1499	1511	1523	1531	1543	1549	1553	1559	1567	1571	1579	1583	18				
15	1607	1609	1613	1619	1621	1627	1637	1657	1663	1667	1669	1693	1697	1699	1709	1721	20				
16	1741	1747	1753	1759	1777	1783	1787	1789	1801	1811	1823	1831	1847	1861	1867	1871	21				
17	1879	1889	1901	1907	1913	1931	1949	1951	1973	1979	1987	1993	1997	1999			22				
18																	23				
19																	24				
20	PPPPPPPPPPPP	RRRRRRRRRRR	IIIIIIIII	MM	MM	FFFFFFFFFF	000000000000	RRRRRRRRRR	HH	HH							26				
21	PPPPPPPPPPPP	RRRRRRRRRRR	IIIIIIIII	MM	MM	FFFFFFFFFF	000000000000	RRRRRRRRRR	HH	HH							27				
22	PP	PP	RR	RR	II	MM	MM	FF	00	00	RR	RR	HH	HH			29				
23	PP	PP	RR	RR	II	MM	MM	MM	FF	00	00	RR	RR	HH	HH		30				
24	PP	PP	RR	RR	II	MM	MM	MM	FF	00	00	RR	RR	HH	HH		31				
25	PPPPPPPPPPPP	RRRRRRRRRRR	II	MM	MM	FFFFFFFFFF	00	00	RRRRRRRRRR	HHHHHHHHHHHH							33				
26	PPPPPPPPPPPP	RRRRRRRRRRR	II	MM	MM	FFFFFFFFFF	00	00	RRRRRRRRRR	HHHHHHHHHHHH							34				
27	PP	RR	RR	II	MM	MM	FF	00	00	RR	RR	HH	HH				35				
28	PP	RR	RR	II	MM	MM	FF	00	00	RR	RR	HH	HH				36				
29	PP	RR	RR	II	MM	MM	FF	00	00	RR	RR	HH	HH				37				
30	PP	RR	RR	IIIIIIIII	MM	MM	FF	000000000000	RR	RR	HH	HH				38					
31	PP	RR	RR	IIIIIIIII	MM	MM	FF	000000000000	RR	RR	HH	HH				39					
32																	40				
33																	41				
34																	42				
35	JJJJJJJJJJ	11	3333333333														43				
36	JJJJJJJJJJ	111	3333333333														44				
37	JJ	1111	33	33													45				
38	JJ	11	33	33													46				
39	JJ	11	33	33													47				
40	JJ	11	3333														48				
41	JJ	11	3333														49				
42	JJ	11	33														50				
43	JJ	JJ	11	33													51				
44	JJ	JJ	11	33													52				
45	JJJJJJJJ	1111111111	3333333333														53				
46	JJJJJJ	1111111111	3333333333														54				
47																	55				
48																	56				
49	****A	END	JOB	13	PRIMFORH	Eratosthenes	Sieve	ROOM	9.17.21	PM	26	NOV	21	PRINTER1	SYS	TK4-	JOB	13	END	A****	57
50	****A	END	JOB	13	PRIMFORH	Eratosthenes	Sieve	ROOM	9.17.21	PM	26	NOV	21	PRINTER1	SYS	TK4-	JOB	13	END	A****	58
51	****A	END	JOB	13	PRIMFORH	Eratosthenes	Sieve	ROOM	9.17.21	PM	26	NOV	21	PRINTER1	SYS	TK4-	JOB	13	END	A****	59
52	****A	END	JOB	13	PRIMFORH	Eratosthenes	Sieve	ROOM	9.17.21	PM	26	NOV	21	PRINTER1	SYS	TK4-	JOB	13	END	A****	60

1