

1																	1				
2	MM MMM FFFFFFFF 11																2				
3	MMM MMMM FF 1111																3				
4	MM MM MM MM FF 11																4				
5	MM MMMM MM FF 11																5				
6	MM MM MM FFFFFFFF 11																6				
7	MM MM FFFFFFFF 11																7				
8	MM MM FF 11																8				
9	MM MM FF 11																9				
10	MM MM FF 11																10				
11	MM MM FF 1111111111																11				
12	MM MM FF 1111111111																12				
13																	13				
14																	14				
15	SSSSSSSSSS				6666666666				444				AAAAAAAAAA				15				
16	SSSSSSSSSSSS				666666666666				4444				AAAAAAAAAAAAA				16				
17	SS SS				66 66				44 44				AA AA				17				
18	SS				66				44 44				AA AA				18				
19	SSS				66				44 44				AA AA				19				
20	SSSSSSSSS				66666666666				444444444444				AAAAAAAAAAAAA				20				
21	SSSSSSSSS				666666666666				4444444444444				AAAAAAAAAAAAA				21				
22	SSS				66 66				44				AA AA				22				
23	SS				66 66				44				AA AA				23				
24	SS SS				66 66				44				AA AA				24				
25	SSSSSSSSSSSS				666666666666				44				AA AA				25				
26	SSSSSSSSSS				6666666666				44				AA AA				26				
27																	27				
28																	28				
29	PPPPPPPPPPP				RRRRRRRRRRR				IIIIIIIIIII				MM MM FFFFFFFF 00000000000				RRRRRRRRRRR HH HH				29
30	PPPPPPPPPPP				RRRRRRRRRRR				IIIIIIIIIII				MMM MMM FFFFFFFF 00000000000				RRRRRRRRRRR HH HH				30
31	PP PP				RR RR				II				MMMM MMMM FF 00 00				RR RR HH HH				31
32	PP PP				RR RR				II				MM MM MM MM FF 00 00				RR RR HH HH				32
33	PP PP				RR RR				II				MM MMMM MM FF 00 00				RR RR HH HH				33
34	PPPPPPPPPPP				RRRRRRRRRRR				II				MM MM MM FFFFFFFF 00 00				RRRRRRRRRRR HHHHHHHHHHHH				34
35	PPPPPPPPPPP				RRRRRRRRRRR				II				MM MM MM FFFFFFFF 00 00				RRRRRRRRRRR HHHHHHHHHHHH				35
36	PP				RR RR				II				MM MM FF 00 00				RR RR HH HH				36
37	PP				RR RR				II				MM MM FF 00 00				RR RR HH HH				37
38	PP				RR RR				II				MM MM FF 00 00				RR RR HH HH				38
39	PP				RR RR				IIIIIIIIIII				MM MM FF 00000000000				RR RR HH HH				39
40	PP				RR RR				IIIIIIIIIII				MM MM FF 00000000000				RR RR HH HH				40
41																	41				
42																	42				
43	JJJJJJJJJJ				11				3333333333				AAAAAAAAAA				43				
44	JJJJJJJJJJ				111				333333333333				AAAAAAAAAAAAA				44				
45	JJ				1111				33 33				AA AA				45				
46	JJ				11				33				AA AA				46				
47	JJ				11				33				AA AA				47				
48	JJ				11				3333				AAAAAAAAAAAAA				48				
49	JJ				11				3333				AAAAAAAAAAAAA				49				
50	JJ				11				33				AA AA				50				
51	JJ JJ				11				33				AA AA				51				
52	JJ JJ				11				33 33				AA AA				52				
53	JJJJJJJJ				1111111111				333333333333				AA AA				53				
54	JJJJJJ				1111111111				3333333333				AA AA				54				
55																	55				
56	****A	START	JOB	13	PRIMFORH	Eratosthenes	Sieve	ROOM	9.17.21	PM	26	NOV	21	PRINTER1	SYS TK4-	JOB	13	START	A****	56	
57	****A	START	JOB	13	PRIMFORH	Eratosthenes	Sieve	ROOM	9.17.21	PM	26	NOV	21	PRINTER1	SYS TK4-	JOB	13	START	A****	57	
58	****A	START	JOB	13	PRIMFORH	Eratosthenes	Sieve	ROOM	9.17.21	PM	26	NOV	21	PRINTER1	SYS TK4-	JOB	13	START	A****	58	
59	****A	START	JOB	13	PRIMFORH	Eratosthenes	Sieve	ROOM	9.17.21	PM	26	NOV	21	PRINTER1	SYS TK4-	JOB	13	START	A****	59	
60																	60				

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 IEFACRT - Stepname Procstep Program Retcode
21.17.20 JOB 13 PRIMFORH PRIMES FORT IEKAA00 RC= 0000
21.17.21 JOB 13 PRIMFORH PRIMES GO LOADER RC= 0000
21.17.21 JOB 13 IEF404I PRIMFORH - ENDED - TIME=21.17.21
21.17.21 JOB 13 \$HASP395 PRIMFORH ENDED

----- JES2 JOB STATISTICS -----

26 NOV 21 JOB EXECUTION DATE

204 CARDS READ

339 SYSOUT PRINT RECORDS

0 SYSOUT PUNCH RECORDS

0.00 MINUTES EXECUTION TIME

```
1 //PRIMFORH JOB (FORTRAN), JOB 13
2 // 'Eratosthenes Sieve',
3 // CLASS=A,
4 // MSGCLASS=A,
5 // REGION=9000K,TIME=1440,
6 // MSGLEVEL=(1,1),
7 // USER=HERC01,PASSWORD= GENERATED BY GDL
8 *****
9 ***
10 *** Name: SYS2.JCLLIB(PRIMFORH)
11 ***
12 *** Desc: Sieve of Eratosthenes programmed in FORTRAN,
13 *** compiled using the IBM OS/360 FORTRAN H Level 21.8 compiler.
14 *** All prime numbers up to the value entered via
15 *** //GO.SYSIN DD are computed.
16 ***
17 *****
18 2 //PRIMES EXEC FORTHCLD,PARM.GO='SIZE=9000000'
19 3 XXFORTHCLD PROC SOUT='*' 00000100
20 4 XXFORT EXEC PGM=IEKAA00,PARM=(SOURCE,MAP) 00000200
21 5 XXSYSPRINT DD SYSOUT=&SOUT 00000300
22 6 XXSYSPUNCH DD SYSOUT=B 00000400
23 7 XXSYSLIN DD DSN= &LOADSET,UNIT=SYSDA,DISP=(MOD,PASS), 00000500
24 XX SPACE=(400,(200,50),RLSE) 00000600
25 8 //FORT.SYSIN DD *
26 9 XXGO EXEC PGM=LOADER,PARM=(MAP),COND=(4,LT,FORT) 00000700
27 10 XXSYSLIB DD DSN=SYS1.FORTLIB,DISP=SHR 00000800
28 11 XXSYSLOUT DD SYSOUT=&SOUT 00000900
29 12 XXSYSLIN DD DSN=*.FORT.SYSLIN,DISP=(OLD,PASS) 00001000
30 13 XXFT05F001 DD DDNAME=SYSIN 00001100
31 14 //GO.FT06F001 DD SYSOUT=*,DCB=(RECFM=FBA,LRECL=166,BLKSIZE=16600)
32 X/FT06F001 DD SYSOUT=&SOUT 00001200
33 15 XXFT07F001 DD SYSOUT=B 00001300
34 16 //GO.SYSIN DD *
```

```
1  STMT NO. MESSAGE
2  -
3      5      IEF653I SUBSTITUTION JCL - SYSOUT=*
4      11     IEF653I SUBSTITUTION JCL - SYSOUT=*
5      14     IEF653I SUBSTITUTION JCL - SYSOUT=*
6
7  IEF236I ALLOC. FOR PRIMFORH FORT PRIMES
8  IEF237I JES2 ALLOCATED TO SYSPRINT
9  IEF237I JES2 ALLOCATED TO SYSPUNCH
10 IEF237I 180 ALLOCATED TO SYSLIN
11 IEF237I JES2 ALLOCATED TO SYSIN
12 IEF142I PRIMFORH FORT PRIMES - STEP WAS EXECUTED - COND CODE 0000
13 IEF285I JES2.JOB00013.S00103 SYSOUT
14 IEF285I JES2.JOB00013.S00104 SYSOUT
15 IEF285I SYS21330.T211720.RA000.PRIMFORH.LOADSET PASSED *-----39
16 IEF285I VOL SER NOS= WORK02.
17 IEF285I JES2.JOB00013.SI0101 SYSIN
18 IEF373I STEP /FORT / START 21330.2117
19 IEF374I STEP /FORT / STOP 21330.2117 CPU 0MIN 00.05SEC SRB 0MIN 00.01SEC VIRT 9064K SYS 232K
20
21 *****
22 * 1. Jobstep of job: PRIMFORH Stepname: FORT Program name: IEKAA00 Executed on 26.11.21 from 21.17.20 to 21.17.20 *
23 * elapsed time 00:00:00,12 CPU-Identifier: TK4- Page-in: 0 *
24 * CPU time 00:00:00,06 Virtual Storage used: 9064K Page-out: 0 *
25 * corr. CPU: 00:00:00,06 CPU time has been corrected by 1 / 1,0 multiplier *
26 * *
27 * I/O Operation *
28 * Number of records read via DD * or DD DATA: 180 *
29 * DMY.....0 DMY.....0 180.....39 DMY.....0 *
30 * *
31 * Charge for step (w/o SYSOUT): 0,10 *
32 *****
33 IEF236I ALLOC. FOR PRIMFORH GO PRIMES
34 IEF237I 148 ALLOCATED TO SYSLIB
35 IEF237I JES2 ALLOCATED TO SYSLOUT
36 IEF237I 180 ALLOCATED TO SYSLIN
37 IEF237I JES2 ALLOCATED TO FT05F001
38 IEF237I JES2 ALLOCATED TO FT06F001
39 IEF237I JES2 ALLOCATED TO FT07F001
40 IEF142I PRIMFORH GO PRIMES - STEP WAS EXECUTED - COND CODE 0000
41 IEF285I SYS1.FORTLIB KEPT *-----65
42 IEF285I VOL SER NOS= MVSRES.
43 IEF285I JES2.JOB00013.S00105 SYSOUT
44 IEF285I SYS21330.T211720.RA000.PRIMFORH.LOADSET PASSED *-----40
45 IEF285I VOL SER NOS= WORK02.
46 IEF285I JES2.JOB00013.SI0102 SYSIN
47 IEF285I JES2.JOB00013.S00106 SYSOUT
48 IEF285I JES2.JOB00013.S00107 SYSOUT
49 IEF373I STEP /GO / START 21330.2117
50 IEF374I STEP /GO / STOP 21330.2117 CPU 0MIN 00.04SEC SRB 0MIN 00.01SEC VIRT 8824K SYS 224K
51
52 *****
53 * 2. Jobstep of job: PRIMFORH Stepname: GO Program name: LOADER Executed on 26.11.21 from 21.17.20 to 21.17.21 *
54 * elapsed time 00:00:00,10 CPU-Identifier: TK4- Page-in: 0 *
55 * CPU time 00:00:00,05 Virtual Storage used: 8824K Page-out: 0 *
56 * corr. CPU: 00:00:00,05 CPU time has been corrected by 1 / 1,0 multiplier *
57 * *
58 * I/O Operation *
59 * Number of records read via DD * or DD DATA: 1 *
60 * 148.....65 DMY.....0 180.....40 DMY.....0 DMY.....0 DMY.....0 *
1412THE
```

1	*	*
2	*	*
3	Charge for step (w/o SYSOUT): 0,08	
4	*****	
5	IEF237I 180 ALLOCATED TO SYS00001	
6	IEF285I SYS21330.T211721.RA000.PRIMFORH.R0000001 KEPT *-----0	
7	IEF285I VOL SER NOS= WORK02.	
8	IEF285I SYS21330.T211720.RA000.PRIMFORH.LOADSET DELETED	
9	IEF285I VOL SER NOS= WORK02.	
10	IEF375I JOB /PRIMFORH/ START 21330.2117	
11	IEF376I JOB /PRIMFORH/ STOP 21330.2117 CPU 0MIN 00.09SEC SRB 0MIN 00.02SEC	
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		

```
1  LEVEL 21.8 ( JUN 74 )                                OS/360  FORTRAN H                                DATE  21.330/21.17.20
2
3  COMPILER OPTIONS - NAME=  MAIN,OPT=00,LINECNT=50,SIZE=0000K,
4                      SOURCE,EBCDIC,NOLIST,NODECK,LOAD,MAP,NOEDIT,NOID,NOXREF
5  C /*-----*/
6  C /*  Sieve of Eratosthenes.                                */
7  C /*-----*/
8  C
9  C /*-----*/
10 C /*  Formats for output.                                    */
11 C /*-----*/
12 ISN 0002  3  FORMAT ( ' ' )
13 ISN 0003  4  FORMAT ( 3X, I7, 1X, I7, 1X, I7, 1X, I7, 1X, I7, 1X, I7, 1X, I7,
14                11X, I7, 1X, I7, 1X, I7, 1X, I7, 1X, I7, 1X, I7, 1X, I7, 1X,
15                1, I7, 1X, I7, 1X, I7 )
16 ISN 0004  5  FORMAT ( ' Sieve of Eratosthenes generated using OS/360 FORTRAN H L
17                level 21.8' )
18 ISN 0005  6  FORMAT ( ' Upper limit of test range = ', I12 )
19 ISN 0006  7  FORMAT ( ' Number of primes in range = ', I12 )
20 ISN 0007  8  FORMAT ( I7 )
21 ISN 0008  993 FORMAT ( ' DEBUG: marking as non prime: ', I4 )
22 ISN 0009  994 FORMAT ( ' DEBUG: starting p=', I4, '      j=', I4 )
23 C
24 C /*-----*/
25 C /*  Define array of flags, one for each integer in the range */
26 C /*  we will test.  If the flag is on, the corresponding      */
27 C /*  number is prime.  If it's off, the number is not prime.  */
28 C /*  We will initialize all the flags to on (assuming every   */
29 C /*  number is prime) and turn them off as we determine the  */
30 C /*  corresponding number is not prime.                        */
31 C /*-----*/
32 ISN 0010  LOGICAL*1 FLAGS(5000002)
33 C
34 C /*-----*/
35 C /*  The PRIME array will hold all the prime numbers we have  */
36 C /*  identified, and CPRIME will contain the number of primes  */
37 C /*  we've found.                                             */
38 C /*-----*/
39 ISN 0011  INTEGER*4 PRIME(350000)
40 ISN 0012  INTEGER*4 CPRIME
41 C
42 C /*-----*/
43 C /*  J is a loop counter and work variable.                    */
44 C /*-----*/
45 ISN 0013  INTEGER*4 J
46 C
47 C /*-----*/
48 C /*  K is the step amount for crossing out prime multiples   */
49 C /*-----*/
50 ISN 0014  INTEGER*4 K
51
52
53
54
55
56
57
58
59
60
```

PAGE 002

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


PAGE 003

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 /*-----*/

PAGE 004

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

/ MAIN / SIZE OF PROGRAM 61AD06 HEXADECIMAL BYTES PAGE 005																			
NAME	TAG		TYPE	ADD.	NAME	TAG		TYPE	ADD.	NAME	TAG		TYPE	ADD.	NAME	TAG		TYPE	ADD.
J	SF		I*4	0001B4	K	SF		I*4	0001B8	P	SF		I*4	0001BC	DEBUG	S		L*1	0001B0
FLAGS	S		L*1	0001CC	LIMIT	SF		I*4	0001C0	PRIME	SF		I*4	4C4D14	CPRIME	SF		I*4	0001C4
IBCOM#	F	XF	R*4	000000	REPEAT	SF		I*4	0001C8										

[illegible]

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

TOTAL LENGTH 61FFD0
ENTRY ADDRESS ACBD0

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

Upper limit of test range = 2000

Number of primes in range = 303

2	3	5	7	11	13	17	19	23	29	31	37	41	43	47	53
67	71	73	79	83	89	97	101	103	107	109	113	127	131	137	139
157	163	167	173	179	181	191	193	197	199	211	223	227	229	233	239
257	263	269	271	277	281	283	293	307	311	313	317	331	337	347	349
367	373	379	383	389	397	401	409	419	421	431	433	439	443	449	457
467	479	487	491	499	503	509	521	523	541	547	557	563	569	571	577
599	601	607	613	617	619	631	641	643	647	653	659	661	673	677	683
709	719	727	733	739	743	751	757	761	769	773	787	797	809	811	821
829	839	853	857	859	863	877	881	883	887	907	911	919	929	937	941
967	971	977	983	991	997	1009	1013	1019	1021	1031	1033	1039	1049	1051	1061
1087	1091	1093	1097	1103	1109	1117	1123	1129	1151	1153	1163	1171	1181	1187	1193
1217	1223	1229	1231	1237	1249	1259	1277	1279	1283	1289	1291	1297	1301	1303	1307
1327	1361	1367	1373	1381	1399	1409	1423	1427	1429	1433	1439	1447	1451	1453	1459
1483	1487	1489	1493	1499	1511	1523	1531	1543	1549	1553	1559	1567	1571	1579	1583
1607	1609	1613	1619	1621	1627	1637	1657	1663	1667	1669	1693	1697	1699	1709	1721
1741	1747	1753	1759	1777	1783	1787	1789	1801	1811	1823	1831	1847	1861	1867	1871
1879	1889	1901	1907	1913	1931	1933	1949	1951	1973	1979	1987	1993	1997	1999	

PPPPPPPPPPPP RRRRRRRRRRRR IIIIIIIIIII MM MM FFFFFFFFFFFFFF 000000000000 RRRRRRRRRRRR HH HH HH
PPPPPPPPPPPP RRRRRRRRRRRR IIIIIIIIIII MMM MMM FFFFFFFFFFFFFF 000000000000 RRRRRRRRRRRR HH HH HH
PP PP PP RR RR II MMMM MMMM FF 00 00 RR RR HH HH
PP PP PP RR RR II MM MM MM MM FF 00 00 RR RR HH HH
PP PP PP RR RR II MM MMMM MM FF 00 00 RR RR HH HH
PPPPPPPPPPPP RRRRRRRRRRRR II MM MM MM FFFFFFFF 00 00 RRRRRRRRRRRR HHHHHHHHHHHH
PPPPPPPPPPPP RRRRRRRRRRRR II MM MM MM FFFFFFFF 00 00 RRRRRRRRRRRR HHHHHHHHHHHH
PP RR RR II MM MM FF 00 00 RR RR HH HH
PP RR RR II MM MM FF 00 00 RR RR HH HH
PP RR RR IIIIIIIIIII MM MM FF 000000000000 RR RR HH HH
PP RR RR IIIIIIIIIII MM MM FF 000000000000 RR RR HH HH

JJJJJJJJJJ 11 3333333333 AAAAAAAAAA
JJJJJJJJJJ 111 333333333333 AAAAAAAAAA
JJ 1111 33 33 AA AA
JJ 11 33 AA AA
JJ 11 33 AA AA
JJ 11 3333 AAAAAAAAAA
JJ 11 3333 AAAAAAAAAA
JJ 11 33 AA AA
JJ JJ 11 33 AA AA
JJ JJ 11 33 AA AA
JJJJJJJJ 1111111111 333333333333 AA AA
JJJJJJ 1111111111 3333333333 AA AA

****A END JOB 13 PRIMFORH Eratosthenes Sieve ROOM 9.17.21 PM 26 NOV 21 PRINTER1 SYS TK4- JOB 13 END A****
****A END JOB 13 PRIMFORH Eratosthenes Sieve ROOM 9.17.21 PM 26 NOV 21 PRINTER1 SYS TK4- JOB 13 END A****
****A END JOB 13 PRIMFORH Eratosthenes Sieve ROOM 9.17.21 PM 26 NOV 21 PRINTER1 SYS TK4- JOB 13 END A****
****A END JOB 13 PRIMFORH Eratosthenes Sieve ROOM 9.17.21 PM 26 NOV 21 PRINTER1 SYS TK4- JOB 13 END A****