

JJJJJJJJJ	2222222222	AAAAAAAAAAAA
JJJJJJJJJJ	222222222222	AAAAAAAAAAAAA
JJ	22	AA
JJ	22	AA
JJ	22	AA
JJ	22	AAAAAAAAAAAA
JJ	22	AAAAAAAAAAAA
JJ	22	AA
JJJ	22	AA
JJJJJJJ	222222222222	AA
JJJJJJ	222222222222	AA

****A	START	JOB	2	FF#PAS	ROOM	5.58.01	PM	26	MAY	20	PRINTER1	SYS	TK4-	JOB	2	START	A****
****A	START	JOB	2	FF#PAS	ROOM	5.58.01	PM	26	MAY	20	PRINTER1	SYS	TK4-	JOB	2	START	A****
****A	START	JOB	2	FF#PAS	ROOM	5.58.01	PM	26	MAY	20	PRINTER1	SYS	TK4-	JOB	2	START	A****
****A	START	JOB	2	FF#PAS	ROOM	5.58.01	PM	26	MAY	20	PRINTER1	SYS	TK4-	JOB	2	START	A****

J E S 2 J O B L O G

17.57.55	JOB	2	\$HASP373	FF#PAS	STARTED - INIT 1 - CLASS A - SYS TK4-	
17.57.55	JOB	2	IEF403I	FF#PAS	- STARTED - TIME=17.57.55	
17.57.55	JOB	2	CC1001C	COMPILE	/PASCAL /00:00:00.11/00:00:00/00000/	/FF#PAS
17.57.55	JOB	2	CC1001C	XREF	/PASCREF /00:00:00.14/00:00:00/00000/	/FF#PAS
17.57.56	JOB	2	CC1001C	POSTPROC	/ASMPCODE/00:00:00.08/00:00:00/00000/	/FF#PAS
17.58.01	JOB	2	CC1001C	GO	/LOADER /00:00:05.50/00:00:06/00000/	/FF#PAS
17.58.01	JOB	2	IEF404I	FF#PAS	- ENDED - TIME=17.58.01	
17.58.01	JOB	2	\$HASP395	FF#PAS	ENDED	

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

26 MAY 20 JOB EXECUTION DATE

5 CARDS READ

1,106 SYSOUT PRINT RECORDS

0 SYSOUT PUNCH RECORDS

0.10 MINUTES EXECUTION TIME

1	//FF#PAS	JOB CLASS=A,MSGCLASS=A,MSGLEVEL=(1,1),NOTIFY=HERC01,	JOB	2
	//	USER=HERC01,PASSWORD=		
		GENERATED BY IKJEFF10		
2	// EXEC PAXCG,	PARM.COMPILE='D-,K+',PARM.GO=MAP		
3	XXPAXCG	PROC GOTIME=299,GOPARM=,GOREG=2048K,DUMP='DUMMY',	00010000	
	XX	SOUT='*',WORK=VIO,OPT='M+' 1..72 MARGINS	00020000	
	***		00030000	
	***		00040000	
4	XXCOMPILE EXEC	PGM=PASCAL,REGION=8192K,PARM='&OPT'	00050000	
5	XXSTEPLIB	DD DSN=PASCAL.PASLIB,DISP=SHR	00060000	
6	XXINPUT	DD DDNAME=SYSIN	00070000	
7	XXPRD	DD DSN=PASCAL.PASOBJ(PASMSG),DISP=SHR	00080000	
8	XXOUTPUT	DD UNIT=&WORK,DSN=&&XIN,SPACE=(TRK,(19,19)),DISP=(,PASS)	00090000	
9	XXPRR	DD DSN=&&PCODE,UNIT=&WORK,DCB=RECFM=VB,	00100000	
	XX	SPACE=(TRK,(20,5),RLSE),DISP=(,PASS)	00110000	
10	XXQRR	DD DSN=&&TABLES,UNIT=&WORK,DCB=RECFM=VB,	00120000	
	XX	SPACE=(TRK,(5,2),RLSE),DISP=(,PASS)	00130000	
	***		00140000	
	***		00150000	
11	//SYSIN DD	DSN=FATT.TEST.SOURCE(FF#PAS),DISP=SHR		
12	XXXREF EXEC	PGM=PASCREF,REGION=2048K,COND=EVEN	00160000	
13	XXSTEPLIB	DD DISP=SHR,DSN=PASCAL.PASLIB	00170000	
14	XXINPUT	DD DISP=(OLD,DELETE),DSN=*.COMPILE.OUTPUT	00180000	
15	XXOUTPUT	DD SYSOUT=&SOUT	00190000	
	***		00200000	
	***		00210000	
16	XXPOSTPROC EXEC	PGM=ASMPCODE,COND=(0,LT,COMPILE),REGION=2048K	00220000	
17	XXSTEPLIB	DD DSN=PASCAL.PASLIB,DISP=SHR	00230000	
18	XXINPUT	DD DSN=*.COMPILE.PRR,DISP=(OLD,DELETE)	00240000	
19	XXPRD	DD DSN=*.COMPILE.QRR,DISP=(OLD,PASS)	00250000	
20	XXOUTPUT	DD SYSOUT=&SOUT	00260000	
21	XXPRR	DD DSN=&&OBJECT,UNIT=&WORK,DCB=RECFM=FB,	00270000	
	XX	SPACE=(TRK,(10,5),RLSE),DISP=(,PASS)	00280000	
	***		00290000	
	***		00300000	
22	XXGO EXEC	PGM=LOADER,COND=((0,LT,COMPILE),(0,LT,POSTPROC)),	00310000	
	XX	PARM='//TIME=&GOTIME,&GOPARM',REGION=&GOREG	00320000	
23	XXSTEPLIB	DD DSN=PASCAL.PASLIB,DISP=SHR (NEEDED FOR K+ ONLY)	00330000	
24	XXSYSLIN	DD DSN=*.POSTPROC.PRR,DISP=(OLD,DELETE)	00340000	
25	XXSYSLOUT	DD SYSOUT=&SOUT	00350000	
26	XXSYSLIB	DD DISP=SHR,DSN=PASCAL.PASLIB	00360000	
27	XX	DD DISP=SHR,DSN=SYS1.FORTLIB	00370000	
28	XXINPUT	DD DDNAME=SYSIN	00380000	
29	XXPRD	DD DUMMY	00390000	
30	XXQRD	DD DSN=*.COMPILE.QRR,DISP=(OLD,DELETE)	00400000	
31	XXFT06F001	DD SYSOUT=&SOUT	00410000	
32	XXOUTPUT	DD SYSOUT=&SOUT	00420000	
33	XXQRR	DD UNIT=&WORK,SPACE=(TRK,(2,2))	00430000	
34	XXSYSUDUMP	DD &DUMP.SYSOUT=&SOUT	00440000	
	***		00450000	
35	//GO.SYSIN DD	DSN=FATT.TEST.DATA(FF),DISP=SHR		

STMT NO. MESSAGE

```

4      IEF653I SUBSTITUTION JCL - PGM=PASCAL,REGION=8192K,PARM='M+'
8      IEF653I SUBSTITUTION JCL - UNIT=VIO,DSN=XXIN,SPACE=(TRK,(19,19)),DISP=(,PASS)
9      IEF653I SUBSTITUTION JCL - DSN=XXPCODE,UNIT=VIO,DCB=RECFM=VB,
10     IEF653I SUBSTITUTION JCL - DSN=XXTABLES,UNIT=VIO,DCB=RECFM=VB,
15     IEF653I SUBSTITUTION JCL - SYSOUT=*
20     IEF653I SUBSTITUTION JCL - SYSOUT=*
21     IEF653I SUBSTITUTION JCL - DSN=XXOBJECT,UNIT=VIO,DCB=RECFM=FB,
22     IEF653I SUBSTITUTION JCL - PARM='//TIME=299,',REGION=2048K
25     IEF653I SUBSTITUTION JCL - SYSOUT=*
31     IEF653I SUBSTITUTION JCL - SYSOUT=*
32     IEF653I SUBSTITUTION JCL - SYSOUT=*
33     IEF653I SUBSTITUTION JCL - UNIT=VIO,SPACE=(TRK,(2,2))
34     IEF653I SUBSTITUTION JCL - DUMMY,SYSOUT=*
IEF236I ALLOC. FOR FF#PAS COMPILE
IEF237I 280 ALLOCATED TO STEPLIB
IEF237I 191 ALLOCATED TO SYS00010
IEF237I 242 ALLOCATED TO INPUT
IEF237I 290 ALLOCATED TO PRD
IEF237I VIO ALLOCATED TO OUTPUT
IEF237I VIO ALLOCATED TO PRR
IEF237I VIO ALLOCATED TO QRR
IEF142I FF#PAS COMPILE - STEP WAS EXECUTED - COND CODE 0000
IEF285I PASCAL.PASLIB KEPT *-----0
IEF285I VOL SER NOS= PUB002.
IEF285I SYS1.UCAT.MVS KEPT *-----0
IEF285I VOL SER NOS= MVSCAT.
IEF285I FATT.TEST.SOURCE KEPT *-----3
IEF285I VOL SER NOS= MV0001.
IEF285I PASCAL.PASOBJ KEPT *-----0
IEF285I VOL SER NOS= PUB003.
IEF285I SYS20147.T175755.RA000.FF#PAS.XIN PASSED *-----9
IEF285I SYS20147.T175755.RA000.FF#PAS.PCODE PASSED *-----8
IEF285I SYS20147.T175755.RA000.FF#PAS.TABLES PASSED *-----3
IEF373I STEP /COMPILE / START 20147.1757
IEF374I STEP /COMPILE / STOP 20147.1757 CPU OMIN 00.11SEC SRB OMIN 00.00SEC VIRT 4148K SYS 296K
**** JOBCARD READ 20147 17:57:55 ****
* PRC-CCI 370/033 VS2 R03.8 TK4- STEP STATISTICS *
* STEP NAME COMPILE USER CORE 4148K TAPES USED/IO 000/000000000 START TIME 17:57:55 TCB TIME 00:00:00.11 *
* PGM NAME PASCAL SYSTEM CORE 296K DISKS USED/IO 004/000000003 STOP TIME 17:57:55 SRB TIME 00:00:00.00 *
* COND CODE 0000 PRIVATE AREA SZ 8192K ALLOC TIME 17:57:55 ELAPSED TIME 00:00:00 PGM LOAD 17:57:55 *
** PGNO * NR SRV UNITS * ACTIVE TIME ** PAGES IN *** PAGES OUT ** # SWAPS * PGS SWAP IN * PGS SWAP OUT * VIO PGS IN * VIO PGS OUT **
* 004 2018 00:00:00.13 0 0 0 0 0 0 0 0 0 19 *
***** VIO IO 000000020 *****
* CPU $ ( 0.03) + EXCP $ ( 0.03) + MEMORY $ ( 1.30) = TOTAL $ ( 1.36) *
*****
IEF236I ALLOC. FOR FF#PAS XREF
IEF237I 280 ALLOCATED TO STEPLIB
IEF237I 191 ALLOCATED TO SYS00012
IEF237I VIO ALLOCATED TO INPUT
IEF237I JES2 ALLOCATED TO OUTPUT
IEF142I FF#PAS XREF - STEP WAS EXECUTED - COND CODE 0000
IEF285I PASCAL.PASLIB KEPT *-----0
IEF285I VOL SER NOS= PUB002.
IEF285I SYS1.UCAT.MVS KEPT *-----0
IEF285I VOL SER NOS= MVSCAT.
IEF285I SYS20147.T175755.RA000.FF#PAS.XIN DELETED *-----10
IEF285I JES2.JOB00002.S00101 SYSOUT
IEF373I STEP /XREF / START 20147.1757
IEF374I STEP /XREF / STOP 20147.1757 CPU OMIN 00.14SEC SRB OMIN 00.00SEC VIRT 2048K SYS 252K
*****
* PRC-CCI 370/033 VS2 R03.8 TK4- STEP STATISTICS *

```

```

* STEP NAME XREF 2048K TAPES USED/IO 000/000000000 START TIME 17:57:55 TCB TIME 00:00:00.14 *
* PGM NAME PASCREF SYSTEM CORE 252K DISKS USED/IO 002/000000000 STOP TIME 17:57:55 SRB TIME 00:00:00.00 *
* COND CODE 0000 PRIVATE AREA SZ 2048K ALLOC TIME 17:57:55 ELAPSED TIME 00:00:00 PGM LOAD 17:57:55 *
** PGNO * NR SRV UNITS * ACTIVE TIME ** PAGES IN *** PAGES OUT ** # SWAPS * PGS SWAP IN * PGS SWAP OUT * VIO PGS IN * VIO PGS OUT **
* 004 1582 00:00:00.14 0 0 0 0 0 0 0 0 0 0 *
***** VIO IO 000000010 *****
* CPU $ ( 0.05) + EXCP $ ( 0.01) + MEMORY $ ( 0.81) = TOTAL $ ( 0.87) *
*****
IEF236I ALLOC. FOR FF#PAS POSTPROC
IEF237I 280 ALLOCATED TO STEPLIB
IEF237I 191 ALLOCATED TO SYS00014
IEF237I VIO ALLOCATED TO INPUT
IEF237I VIO ALLOCATED TO PRD
IEF237I JES2 ALLOCATED TO OUTPUT
IEF237I VIO ALLOCATED TO PRR
IEF142I FF#PAS POSTPROC - STEP WAS EXECUTED - COND CODE 0000
IEF285I PASCAL.PASLIB KEPT *-----0
IEF285I VOL SER NOS= PUB002.
IEF285I SYS1.UCAT.MVS KEPT *-----0
IEF285I VOL SER NOS= MVSCAT.
IEF285I SYS20147.T175755.RA000.FF#PAS.PCODE DELETED *-----9
IEF285I SYS20147.T175755.RA000.FF#PAS.TABLES PASSED *-----4
IEF285I JES2.JOB00002.S00102 SYSOUT
IEF285I SYS20147.T175755.RA000.FF#PAS.OBJECT PASSED *-----7
IEF373I STEP /POSTPROC/ START 20147.1757
IEF374I STEP /POSTPROC/ STOP 20147.1757 CPU OMIN 00.08SEC SRB OMIN 00.00SEC VIRT 2064K SYS 300K
*****
* PRC-CCI 370/033 VS2 R03.8 TK4- STEP STATISTICS *
* STEP NAME POSTPROC USER CORE 2064K TAPES USED/IO 000/000000000 START TIME 17:57:55 TCB TIME 00:00:00.08 *
* PGM NAME ASMPCODE SYSTEM CORE 300K DISKS USED/IO 002/000000000 STOP TIME 17:57:56 SRB TIME 00:00:00.00 *
* COND CODE 0000 PRIVATE AREA SZ 2048K ALLOC TIME 17:57:55 ELAPSED TIME 00:00:00 PGM LOAD 17:57:55 *
** PGNO * NR SRV UNITS * ACTIVE TIME ** PAGES IN *** PAGES OUT ** # SWAPS * PGS SWAP IN * PGS SWAP OUT * VIO PGS IN * VIO PGS OUT **
* 004 912 00:00:00.08 0 0 0 0 0 0 0 0 7 *
***** VIO IO 000000020 *****
* CPU $ ( 0.02) + EXCP $ ( 0.02) + MEMORY $ ( 0.47) = TOTAL $ ( 0.51) *
*****
IEF236I ALLOC. FOR FF#PAS GO
IEF237I 280 ALLOCATED TO STEPLIB
IEF237I 191 ALLOCATED TO SYS00016
IEF237I VIO ALLOCATED TO SYSLIN
IEF237I JES2 ALLOCATED TO SYSLOUT
IEF237I 280 ALLOCATED TO SYSLIB
IEF237I 148 ALLOCATED TO
IEF237I 242 ALLOCATED TO INPUT
IEF237I DMY ALLOCATED TO PRD
IEF237I VIO ALLOCATED TO QRD
IEF237I JES2 ALLOCATED TO FT06F001
IEF237I JES2 ALLOCATED TO OUTPUT
IEF237I VIO ALLOCATED TO QRR
IEF237I DMY ALLOCATED TO SYSUDUMP
IEF142I FF#PAS GO - STEP WAS EXECUTED - COND CODE 0000
IEF285I PASCAL.PASLIB KEPT *-----0
IEF285I VOL SER NOS= PUB002.
IEF285I SYS1.UCAT.MVS KEPT *-----0
IEF285I VOL SER NOS= MVSCAT.
IEF285I SYS20147.T175755.RA000.FF#PAS.OBJECT DELETED *-----8
IEF285I JES2.JOB00002.S00103 SYSOUT
IEF285I PASCAL.PASLIB KEPT *-----17
IEF285I VOL SER NOS= PUB002.
IEF285I SYS1.FORTLIB KEPT *-----67
IEF285I VOL SER NOS= MVSRES.
IEF285I FATT.TEST.DATA KEPT *-----2
IEF285I VOL SER NOS= MV0001.
IEF285I SYS20147.T175755.RA000.FF#PAS.TABLES DELETED *-----4
IEF285I JES2.JOB00002.S00104 SYSOUT
IEF285I JES2.JOB00002.S00105 SYSOUT

```

IEF285I SYS20147.T175755.RA000.FF#PAS.R0000001 DELETED *-----3

IEF373I STEP /GO / START 20147.1757

IEF374I STEP /GO / STOP 20147.1758 CPU OMIN 05.50SEC SRB OMIN 00.01SEC VIRT 2068K SYS 280K

* PRC-CCI 370/033 VS2 R03.8 TK4- STEP STATISTICS *

* STEP NAME GO USER CORE 2068K TAPES USED/IO 000/000000000 START TIME 17:57:56 TCB TIME 00:00:05.50 *

* PGM NAME LOADER SYSTEM CORE 280K DISKS USED/IO 004/000000086 STOP TIME 17:58:01 SRB TIME 00:00:00.01 *

* COND CODE 0000 PRIVATE AREA SZ 2048K ALLOC TIME 17:57:56 ELAPSED TIME 00:00:06 PGM LOAD 17:57:56 *

** PGNO * NR SRV UNITS * ACTIVE TIME ** PAGES IN *** PAGES OUT ** # SWAPS * PGS SWAP IN * PGS SWAP OUT * VIO PGS IN * VIO PGS OUT **

* 004 61286 00:00:05.57 0 0 0 0 0 0 0 0 0 2 *

***** VIO IO 000000015 *****

* CPU \$ (1.98) + EXCP \$ (0.13) + MEMORY \$ (32.41) = TOTAL \$ (34.52) *

IEF375I JOB /FF#PAS / START 20147.1757

IEF376I JOB /FF#PAS / STOP 20147.1758 CPU OMIN 05.83SEC SRB OMIN 00.01SEC

```

1      ) (* NEL LIBRO SHOW DI MAGIA MATEMATICA DI MARTIN GARDNER NEL CAPITOLO 4      *)
2      ) (* STRANEZZE DEI FATTORIALI, VIENE CITATO IL LIBRO DI ROBERT E. SMITH      *)
3      ) (* THE BASES OF FORTRAN, DOVE SI TROVA UN PROGRAMMA PER STAMPARE LE CIFRE DI *)
4      ) (* N! FATTORIALE IN FORMA DI TRIANGOLO, LOSANGA, ESAGONO ED OTTAGONO.      *)
5      ) (* QUESTO LIBRO NON SI TROVA ON-LINE, QUINDI BASANDOMI POCO SUL TESTO E MOLTO *)
6      ) (* SULLE FIGURE DEL LIBRO DI GARDNER ECCO LA MIA VERSIONE IN PASCAL.      *)
7      ) (*                                                                            *)
8      ) (* ORIGINALI:                                                                *)
9      ) (* SCIENTIFIC AMERICAN VOL. 217, NO. 2 (AUGUST 1967), PP. 104-109        *)
10     ) (* MATHEMATICAL MAGIC SHOW - ALFRED A. KNOPF, INC., NEW YORK, NY, 1977      *)
11     ) (*                                                                            *)
12     ) (* N.B. NELLA VECCHIA VERSIONE DEL COMPILATORE NON È POSSIBILE PASSARE LE  *)
13     ) (* FUNZIONI COME PARAMETRO                                                  *)
14     ) (* ALTRE LIMITAZIONI SONO LA MANCANZA DI SET DI CARATTERI E LA MANCANZA    *)
15     ) (* DELLA CLAUSOLA OTHERWISE NEL CASE                                       *)
16     ) (*                                                                            *)
17     ) (* MI TROVO MEGLIO CON WRITELN('1') RISPETTO A PAGE(OUTPUT)              *)
18     )
19     ) PROGRAM FIGUREFATTORIALI(INPUT,OUTPUT);
20     ) (* FATTORIALE MASSIMO IL QUALE NELLA FORMA VOLUTA SIA STAMPABILE IN 132    *)
21     ) (* COLONNE                                                                  *)
22     ) CONST MAXTRI=1494; (* 4.096 CIFRE, LATO 64, LINEA MASSIMA 64*2 - 1 = 127 *)
23     )      MAXLOS=2545; (* 7.565 CIFRE, LATO 62, LINEA MASSIMA 62*2 - 1 = 123 *)
24     )      MAXESA=2236; (* 6.521 CIFRE, LATO 41, LINEA MASSIMA 41*3 - 2 = 121 *)
25     )      MAXOTT=2739; (* 8.229 CIFRE, LATO 35, LINEA MASSIMA 35*3 - 2 = 103 *)
26     )      MAXNC=8229; (* OVVIAMENTE BASTA FINO A 2.739! *)
27     ) TYPE FORME=(TRIANGOLO,LOSANGA,ESAGONO,OTTAGONO); (* FORME AMMISSIBILI *)
28     ) VAR CIFRE:ARRAY[1..MAXNC] OF 0..9; (* FATTORIALE NELLA FORMA VOLUTA *)
29     33220 1) MIN,MAX:INTEGER; (* INPUT: INTERVALLO DI RICERCA VOLUTO *)
30     33228 1) MOD0:CHAR; (* INPUT: MODO DI ESECUZIONE DEL PROGRAMMA *)
31     33229 1)
32     33229 1) FUNCTION LATOTRIANGOLO(N:INTEGER):REAL;
33     )      BEGIN
34     )      LATOTRIANGOLO:=SQRT(N)
35     7 2)      END;
36     10 2)
37     )      FUNCTION LATOLOSANGA(N:INTEGER):REAL;
38     )      BEGIN
39     )      LATOLOSANGA:=(SQRT(2*N-1)+1)/2
40     15 2)      END;
41     30 2)
42     )      FUNCTION LATOESAGONO(N:INTEGER):REAL;
43     )      BEGIN
44     )      LATOESAGONO:=(SQRT(16*N-7)+5)/8
45     15 2)      END;
46     50 2)
47     )      FUNCTION LATOOTTAGONO(N:INTEGER):REAL;
48     )      BEGIN
49     )      LATOOTTAGONO:=(SQRT(7*N-3)+5)/7
50     15 2)      END;
51     70 2)
52     )      PROCEDURE STAMPAFORMA(FORMA:FORME; N:INTEGER); (* STAMPA N CIFRE IN FORMA *)
53     )
54     )      PROCEDURE STAMPALINEA(LUNGHEZZA:INTEGER); (* STAMPA UNA RIGA CENTRATA *)
55     )      VAR FINE,I:INTEGER;

```

```

56      92 3) BEGIN
57      ) FINE:=N-LUNGHEZZA;
58      6 3) WRITE(' ',' ': (132-LUNGHEZZA) DIV 2);
59      19 3) FOR I:=N DOWNT0 FINE+1
60      22 3) DO WRITE(CIFRE[I]:1);
61      48 3) WRITELN;
62      52 3) N:=FINE
63      52 3) END;
64      125 3)
65      ) (* STAMPA IN FORMA DI TRIANGOLO *)
66      ) (* N^2 SLOANE A000290 *)
67      ) PROCEDURE STAMPATRIANGOLO(LATO:INTEGER);
68      ) VAR I:INTEGER;
69      88 3) BEGIN
70      ) FOR I:=1 TO LATO
71      4 3) DO STAMPALINEA(2*I-1)
72      17 3) END;
73      152 3)
74      ) (* STAMPA IN FORMA DI LOSANGA (ROMBO) *)
75      ) (* 2*N*(N+1)+1 ANCHE N^2+(N+1)^2 SLOANE A001844 *)
76      ) PROCEDURE STAMPALOSANGA(LATO:INTEGER);
77      ) VAR I:INTEGER;
78      88 3) BEGIN
79      ) FOR I:=1 TO LATO
80      4 3) DO STAMPALINEA(2*I-1);
81      26 3) FOR I:=LATO-1 DOWNT0 1
82      30 3) DO STAMPALINEA(2*I-1)
83      43 3) END;
84      205 3)
85      ) (* STAMPA IN FORMA DI ESAGONO *)
86      ) (* 4*N^2+3*N+1 SLOANE A033951 *)
87      ) PROCEDURE STAMPAESAGONO(LATO:INTEGER);
88      ) VAR I:INTEGER;
89      88 3) BEGIN
90      ) FOR I:=0 TO LATO-1
91      5 3) DO STAMPALINEA(LATO+2*I);
92      28 3) FOR I:=LATO-2 DOWNT0 0
93      32 3) DO STAMPALINEA(LATO+2*I)
94      45 3) END;
95      260 3)
96      ) (* STAMPA IN FORMA DI OTTAGONO *)
97      ) (* 7*N^2+4*N+1 SLOANE A005892 *)
98      ) PROCEDURE STAMPAOTTAGONO(LATO:INTEGER);
99      ) VAR I:INTEGER;
100     88 3) BEGIN
101     ) FOR I:=0 TO LATO-1
102     5 3) DO STAMPALINEA(LATO+2*I);
103     28 3) FOR I:=1 TO LATO-2
104     31 3) DO STAMPALINEA(3*LATO-2);
105     54 3) FOR I:=LATO-1 DOWNT0 0
106     58 3) DO STAMPALINEA(LATO+2*I)
107     71 3) END;
108     341 3)
109     ) BEGIN (* STAMPAFORMA *)
110     ) CASE FORMA OF

```



```

111      4 2)      TRIANGOLO:STAMPATRIANGOLO(TRUNC(LATOTRIANGOLO(N)));
112     13 2)      LOSANGA:STAMPALOSANGA(TRUNC(LATOLOSANGA(N)));
113     21 2)      ESAGONO:STAMPAESAGONO(TRUNC(LATOESAGONO(N)));
114     29 2)      OTTAGONO:STAMPAOTTAGONO(TRUNC(LATOOTTAGONO(N)))
115     35 2)      END;
116     41 2)      WRITELN
117     41 2)      END;
118    387 2)
119      )      (* CALCOLA N! SLOANE A000142 *)
120      )      (* RITORNA IL NUMERO DI CIFRE DI N! E LE CIFRE NEL VETTORE GLOBALE *)
121      )      FUNCTION FATTORIALE(N:INTEGER):INTEGER;
122      )      VAR I,J,NC,RIPORTO:INTEGER;
123    100 2)      BEGIN
124      )      NC:=1;
125      4 2)      CIFRE[1]:=1;
126     10 2)      FOR I:=N DOWNT0 2
127     12 2)      DO BEGIN
128     19 2)          RIPORTO:=0;
129     21 2)          FOR J:=1 TO NC
130     23 2)          DO BEGIN
131     30 2)              RIPORTO:=RIPORTO+CIFRE[J]*I;
132     40 2)              CIFRE[J]:=RIPORTO MOD 10;
133     48 2)              RIPORTO:=RIPORTO DIV 10
134     49 2)          END;
135     60 2)          WHILE RIPORTO>0
136     61 2)          DO BEGIN
137     65 2)              NC:=NC+1;
138     69 2)              CIFRE[NC]:=RIPORTO MOD 10;
139     77 2)              RIPORTO:=RIPORTO DIV 10
140     78 2)          END
141     81 2)      END;
142     90 2)      FATTORIALE:=NC
143     90 2)      END;
144    480 2)
145      )      PROCEDURE ESEMPILIBRO; (* ZANICHELLI, BOLOGNA, 1980 *)
146      )
147      )      PROCEDURE TESTATAESEMPI(PRIMA:BOOLEAN); (* PRIMA RIGA DI OGNI PAGINA *)
148      )      BEGIN
149      )          IF NOT PRIMA
150      2 3)          THEN WRITELN('1');
151     13 3)          WRITELN('MARTIN GARDNER, SHOW DI MAGIA MATEMATICA':87);
152     21 3)          WRITELN
153     21 3)      END;
154    506 3)
155      )      BEGIN
156      )          TESTATAESEMPI(TRUE);
157      5 2)          STAMPAFORMA(TRIANGOLO,FATTORIALE(12));
158     11 2)          WRITELN('9 CIFRE DI 12! NEL TESTO A PAGINA 35':85);
159     19 2)          TESTATAESEMPI(FALSE);
160     22 2)          STAMPAFORMA(TRIANGOLO,FATTORIALE(105));
161     28 2)          WRITELN('169 CIFRE DI 105! FIGURA 10 A PAGINA 37':86);
162     36 2)          TESTATAESEMPI(FALSE);
163     39 2)          STAMPAFORMA(TRIANGOLO,FATTORIALE(508));
164     45 2)          WRITELN('1.156 CIFRE DI 508! FIGURA 11 A PAGINA 38':87);
165     53 2)          TESTATAESEMPI(FALSE);

```

```

166      56 2) STAMPAFORMA(LOSANGA,FATTORIALE(35));
167      62 2) (* NEL LIBRO LA CIFRA CENTRALE '6' NON È STAMPATA *)
168      62 2) (* COME SFIDA AL LETTORE VIENE CHIESTO DI TROVARLA *)
169      62 2) WRITELN('41 CIFRE DI 35! NEL TESTO ALLE PAGINE 37 E 38':90);
170      70 2) TESTATAESEMPI(FALSE);
171      73 2) STAMPAFORMA(ESAGONO,FATTORIALE(477));
172      79 2) WRITELN('1.073 CIFRE DI 477! FIGURA 12 A PAGINA 39':87);
173      87 2) TESTATAESEMPI(FALSE);
174      90 2) STAMPAFORMA(OTTAGONO,FATTORIALE(2206));
175      96 2) WRITELN('6.421 CIFRE DI 2.206! FIGURA 13 PAGINA 40':87)
176     104 2) END;
177     611 2)
178          ) (* CALCOLA, VERIFICA E SE È NELLA FORMA VOLUTA STAMPA *)
179          ) (* NON PARTICOLARMENTE INTELLIGENTE OD EFFICIENTE *)
180          ) PROCEDURE TESTFORMA(FORMA:FORME; MIN,MAX:INTEGER);
181          ) VAR N,NC:INTEGER;
182          100 2) TROVATO1,PRIMA:BOOLEAN;
183          102 2)
184          102 2) (* NUMERO DI CIFRE DI N! SLOANE A034886 *)
185          102 2) (* FUNZIONE VALIDA FINO A N = 6.561.101.970.383 *)
186          102 2) (* CAUSA LIMITI PRECISIONE ARRIVA FINO A N = 268.609.166 PER IL QUALE *)
187          102 2) (* RITORNA 2.147.483.642 (CIRCA MAXINT) *)
188          102 2) FUNCTION KAMENETSKY(N:INTEGER):INTEGER;
189          ) CONST C1=1.83787706640934548; (* LN(2*PI) *)
190          ) C2=4.60517018598809136; (* 2*LN(10) *)
191          ) BEGIN
192          ) IF N<2
193          3 3) THEN KAMENETSKY:=1
194          7 3) ELSE KAMENETSKY:=TRUNC((C1-2*N+LN(N)*(1+2*N))/C2)+1
195          33 3) END;
196          648 3)
197          ) (* DATO N, RITORNA IL NUMERO MASSIMO IL CUI FATTORIALE HA N CIFRE *)
198          ) (* N.B. PER ORA NON È USATA! *)
199          ) (*
200          ) FUNCTION INVKAMENETSKY(N:INTEGER):INTEGER;
201          ) VAR L,H,M,K:INTEGER;
202          ) BEGIN
203          ) IF N=1
204          ) THEN INVKAMENETSKY:=3
205          ) ELSE BEGIN
206          ) L:=0;
207          ) H:=N;
208          ) M:=2*N;
209          ) K:=KAMENETSKY(M);
210          ) WHILE (L<=H)AND(K<>N)
211          ) DO BEGIN
212          ) IF K<N
213          ) THEN L:=M+1
214          ) ELSE H:=M-1;
215          ) M:=(L+H) DIV 2;
216          ) K:=KAMENETSKY(M)
217          ) END;
218          ) INVKAMENETSKY:=M
219          ) END
220          ) END;

```

```

221      )      *)
222      )
223      )      (* VERO SE N È RAPPRESENTABILE IN FORMA *)
224      )      FUNCTION INFORMA(FORMA:FORME; N:INTEGER):BOOLEAN;
225      )      VAR LATO:REAL;
226      96 3)      BEGIN
227      )          CASE FORMA OF
228      4 3)              TRIANGOLO:LATO:=LATOTRIANGOLO(N);
229      11 3)              LOSANGA:LATO:=LATOLOSANGA(N);
230      17 3)              ESAGONO:LATO:=LATOESAGONO(N);
231      23 3)              OTTAGONO:LATO:=LATOOTTAGONO(N)
232      26 3)          END;
233      33 3)          INFORMA:=LATO=TRUNC(LATO)
234      36 3)      END;
235      688 3)
236      )      (* LUNGHEZZA DI N CON SEPARATORI DI MIGLIAIA *)
237      )      FUNCTION LUN1000(N:INTEGER):INTEGER;
238      )      CONST C1=2.30258509299404568; (* LN(10) *)
239      )      VAR L:INTEGER;
240      88 3)      BEGIN
241      )          IF N=0
242      3 3)              THEN LUN1000:=1
243      7 3)              ELSE BEGIN
244      10 3)                  L:=TRUNC(LN(N)/C1);
245      20 3)                  LUN1000:=L+L DIV 3+1
246      25 3)              END
247      28 3)      END;
248      717 3)
249      )      (* STAMPA N CON SEPARATORE DI MIGLIAIA, IN CAMPO M *)
250      )      PROCEDURE STAMPA1000(N,M:INTEGER);
251      )
252      )      (* STAMPA N<1000 EVENTUALMENTE PRECEDUTO DA ZERI *)
253      )      PROCEDURE STAMPA0(N:INTEGER);
254      )      BEGIN
255      )          CASE N OF
256      3 4)              0..9:WRITE('00');
257      13 4)              10..99:WRITE('0')
258      20 4)              (* NON ESISTE OTHERWISE, SE NESSUN CASO *)
259      20 4)              (* È VERIFICATO PROSEGUE IN SILENZIO... *)
260      20 4)          END;
261      121 4)          WRITE(N:1)
262      127 4)      END;
263      845 4)
264      )      BEGIN (* STAMPA1000 *)
265      )          IF N<1000
266      3 3)              THEN WRITE(N:M)
267      13 3)              ELSE BEGIN
268      14 3)                  STAMPA1000(N DIV 1000,M-4);
269      22 3)                  WRITE('.');
270      28 3)                  STAMPA0(N MOD 1000)
271      32 3)              END
272      33 3)      END;
273      879 3)
274      )      (* STAMPA CENTRATA DELLA PRIMA RIGA DI OGNI PAGINA *)
275      )      (* RICALCOLA OGNI VOLTA, INEFFICIENTE MA NEL COMPLESSO PIÙ LEGGIBILE *)

```

```

276      )      PROCEDURE TESTATA(FORMA:FORME; MIN,MAX:INTEGER; PRIMA:BOOLEAN);
277      )      VAR CMIN,CMAX,CTST:INTEGER;
278      108 3)      BEGIN
279      )          CMIN:=LUN1000(MIN);
280      6 3)          CMAX:=LUN1000(MAX);
281      10 3)          CASE FORMA OF
282      12 3)              TRIANGOLO:CTST:=(98-CMIN-CMAX) DIV 2;
283      23 3)              LOSANGA:CTST:=(100-CMIN-CMAX) DIV 2;
284      33 3)              ESAGONO:CTST:=(100-CMIN-CMAX) DIV 2;
285      43 3)              OTTAGONO:CTST:=(100-CMIN-CMAX) DIV 2
286      49 3)          END;
287      57 3)          IF NOT PRIMA
288      57 3)              THEN WRITELN('1');
289      68 3)          WRITE(' ':CTST);
290      74 3)          WRITE('IN FORMA DI ');
291      81 3)          CASE FORMA OF
292      83 3)              TRIANGOLO:WRITE('TRIANGOLO');
293      93 3)              LOSANGA:WRITE('LOSANGA');
294      102 3)              ESAGONO:WRITE('ESAGONO');
295      111 3)              OTTAGONO:WRITE('OTTAGONO')
296      119 3)          END;
297      124 3)          WRITE(' DA ');
298      131 3)          STAMPA1000(MIN,CMIN);
299      135 3)          WRITE('! FINO A ');
300      142 3)          STAMPA1000(MAX,CMAX);
301      146 3)          WRITELN('!');
302      153 3)          WRITELN
303      153 3)      END;
304      1037 3)
305      )      (* NUMERO DI ZERI FINALI DI N! SLOANE A027868 *)
306      )      FUNCTION ZEROFINALI(N:INTEGER):INTEGER;
307      )      VAR T,P,S:INTEGER;
308      96 3)      BEGIN
309      )          T:=0;
310      4 3)          P:=5;
311      6 3)          S:=N DIV P;
312      10 3)          WHILE S>0
313      11 3)              DO BEGIN
314      15 3)                  T:=T+S;
315      19 3)                  P:=P*5;
316      23 3)                  S:=N DIV P
317      24 3)              END;
318      28 3)          ZEROFINALI:=T
319      28 3)      END;
320      1068 3)
321      )      (* STAMPA CENTRATA DELLA ULTIMA RIGA DI OGNI PAGINA *)
322      )      PROCEDURE FONDO(N,NC,ZF:INTEGER);
323      )      VAR CN,CNC,CZF,CFND:INTEGER;
324      108 3)      BEGIN
325      )          CN:=LUN1000(N);
326      6 3)          CNC:=LUN1000(NC);
327      10 3)          CZF:=LUN1000(ZF);
328      14 3)          IF ZF=0
329      15 3)              THEN CFND:=(122-CN-CNC) DIV 2
330      24 3)              ELSE CFND:=(105-CN-CNC-CZF) DIV 2;

```

```

331      38 3)      WRITE(' ':CFND);
332      44 3)      STAMPA1000(N,CN);
333      48 3)      WRITE('! DI ');
334      55 3)      STAMPA1000(NC,CNC);
335      59 3)      WRITE(' CIFR');
336      66 3)      IF NC>1
337      67 3)      THEN WRITE('E')
338      77 3)      ELSE WRITE('A');
339      84 3)      IF ZF=0
340      85 3)      THEN WRITELN
341      89 3)      ELSE BEGIN
342      94 3)          WRITE(' CON ');
343     101 3)          STAMPA1000(ZF,CZF);
344     105 3)          WRITE(' ZER');
345     112 3)          IF ZF>1
346     113 3)          THEN WRITELN('I FINALI')
347     125 3)          ELSE WRITELN('O FINALE')
348     134 3)      END
349     134 3)      END;
350     1203 3)
351      ) BEGIN (* TESTFORMA *)
352      )      TROVATO1:=FALSE; (* INDICA SE TROVATO ALMENO UN FATTORIALE DA STAMPARE *)
353      4 2)      PRIMA:=TRUE; (* INDICA SE È LA PRIMA PAGINA STAMPATA *)
354      6 2)      FOR N:=MIN TO MAX
355      8 2)      DO BEGIN
356     15 2)          NC:=KAMENETSKY(N);
357     19 2)          IF INFORMA(FORMA,NC)
358     22 2)          THEN BEGIN
359     25 2)              TROVATO1:=TRUE;
360     27 2)              TESTATA(FORMA,MIN,MAX,PRIMA);
361     33 2)              PRIMA:=FALSE;
362     35 2)              STAMPAFORMA(FORMA,FATTORIALE(N));
363     41 2)              FONDO(N,NC,ZEROFINALI(N))
364     47 2)          END
365     48 2)      END;
366     56 2)      IF NOT TROVATO1
367     56 2)      THEN BEGIN
368     60 2)          TESTATA(FORMA,MIN,MAX,PRIMA);
369     66 2)          WRITELN('NESSUNO':70)
370     74 2)      END
371     74 2)      END;
372     1278 2)
373      ) (* RIPORTA TIPO ERRORE E MUORE *)
374      ) PROCEDURE ERRORE(ERR:INTEGER);
375      ) BEGIN
376      )      WRITE(' ERRORE: ');
377      9 2)      CASE ERR OF
378     10 2)          1:WRITELN('MODO DEVE ESSERE SOLO G, T, L, E, O');
379     21 2)          2:WRITELN('LIMITI INCONSISTENTI');
380     31 2)          3:WRITELN('ECCEDE IL LIMITE DI 132 CARATTERI')
381     40 2)      END;
382     44 2)      EXIT(ERR) (* MUORE CON ERR COME RETURN CODE *)
383     46 2)      END;
384     1325 2)
385      ) BEGIN (* MAIN *)

```

```

386      ) (* CI SONO 2 MODI DI ESECUZIONE: *)
387      ) (* - STAMPARE QUELLO CHE SI TROVA NEL LIBRO DI GARDNER (G) *)
388      ) (* - STAMPARE I FATTORIALI RAPPRESENTABILI IN UNA DELLE POSSIBILI FORME *)
389      ) (* (T,L,E,O) SEGUITO DAI LIMITI MINIMO E MASSIMO NEI QUALI CERCARE *)
390      ) READLN(MODO,MIN,MAX);
391      21 1) (* NON POSSO VERIFICARE CON: MODO IN ['G','T','L',...: *)
392      21 1) IF (MODO<>'G')AND(MODO<>'T')AND(MODO<>'L')AND(MODO<>'E')AND(MODO<>'O')
393      39 1) THEN ERRORE(1);
394      45 1) IF MODO='G'
395      46 1) THEN ESEMPILIBRO
396      50 1) ELSE BEGIN
397      53 1)     IF (MIN<0)OR(MAX<0)OR(MIN>MAX)
398      63 1)     THEN ERRORE(2);
399      69 1)     IF (MODO='T')AND(MAX>MAXTRI)OR(MODO='L')AND(MAX>MAXLOS)OR
400      84 1)     (MODO='E')AND(MAX>MAXESA)OR(MODO='O')AND(MAX>MAXOTT)
401      98 1)     THEN ERRORE(3);
402      105 1)     CASE MODO OF
403      107 1)         'T':TESTFORMA(TRIANGOLO,MIN,MAX);
404      115 1)         'L':TESTFORMA(LOSANGA,MIN,MAX);
405      122 1)         'E':TESTFORMA(ESAGONO,MIN,MAX);
406      129 1)         'O':TESTFORMA(OTTAGONO,MIN,MAX)
407      134 1)     END
408      167 1) END
409      167 1) END.

```

```

**** NO SYNTAX ERROR(S) DETECTED.
**** 409 LINE(S) READ, 23 PROCEDURE(S) COMPILED,
**** 1494 P_INSTRUCTIONS GENERATED, 0.05 SECONDS IN COMPILATION.

```

CROSS REFERENCE OF IDENTIFIERS, LABEL DECLARATIONS AND GOTO STATEMENTS:

AND	392	392	392	392	399	399	400	400												
ARRAY	28																			
BEGIN	33	38	43	48	56	69	78	89	100	109	123	127	130	136	148	155	191	226	240	243
	254	264	267	278	308	313	324	341	351	355	358	367	375	385	396					
BOOLEAN	147	182	224	276																
CASE	110	227	255	281	291	377	402													
CFND	323	329	330	331																
CHAR	30																			
CIFRE	28	60	125	131	132	138														
CMAX	277	280	282	283	284	285	300													
CMIN	277	279	282	283	284	285	298													
CN	323	325	329	330	332															
CNC	323	326	329	330	334															
CONST	22	189	238																	
CTST	277	282	283	284	285	289														
CZF	323	327	330	343																
DIV	58	133	139	245	268	282	283	284	285	311	316	329	330							
DO	60	71	80	82	91	93	102	104	106	127	130	136	313	355						
DOWNT0	59	81	92	105	126															
ELSE	194	243	267	330	338	341	347	396												
END	35	40	45	50	63	72	83	94	107	115	117	134	140	141	143	153	176	195	232	234
	246	247	260	262	271	272	286	296	303	317	319	348	349	364	365	370	371	381	383	407
	408	409																		
ERR	374	377	382																	
ESAGONO	27	113	171	230	284	294	405													
EXIT	382																			
FALSE	159	162	165	170	173	352	361													
FINE	55	57	59	62																
FOR	59	70	79	81	90	92	101	103	105	126	129	354								
FORMA	52	110	180	224	227	276	281	291	357	360	362	368								
FORME	27	52	180	224	276															
FUNCTION	32	37	42	47	121	188	224	237	306											
I	55	59	60	68	70	71	77	79	80	81	82	88	90	91	92	93	99	101	102	103
	105	106	122	126	131															
IF	149	192	241	265	287	328	336	339	345	357	366	392	394	397	399					
INPUT	19																			
INTEGER	29	32	37	42	47	52	54	55	67	68	76	77	87	88	98	99	121	121	122	180
	181	188	188	224	237	237	239	250	253	276	277	306	306	307	322	323	374			
J	122	129	131	132																
L	239	244	245	245																
LATO	67	70	76	79	81	87	90	91	92	93	98	101	102	103	104	105	106	225	228	229
	230	231	233	233																
LN	194	244																		
LOSANGA	27	112	166	229	283	293	404													
LUNGHEZZA	54	57	58																	
M	250	266	268																	
MAX	29	180	276	280	300	354	360	368	390	397	397	399	399	400	400	403	404	405	406	
MIN	29	180	276	279	298	354	360	368	390	397	397	403	404	405	406					
MOD	132	138	270																	
MOD0	30	390	392	392	392	392	392	394	399	399	400	400	402							
N	32	34	37	39	42	44	47	49	52	57	59	62	111	112	113	114	121	126	181	188
	192	194	194	194	224	228	229	230	231	237	241	244	250	253	255	261	265	266	268	270

[illegible]

STAMPATRIA	67	111					
STAMPA0	253	270					
STAMPA1000	250	268	298	300	332	334	343
TESTATA	276	360	368				
TESTATAESE	147	156	159	162	165	170	173
TESTFORMA	180	403	404	405	406		
ZEROFINALI	306	318	363				

LIST OF PROCEDURES AND FUNCTIONS:

PROGRAM 19

OF IDENTIFIERS: 104, # OF OCCURENCES: 799, # OF PROCEDURES: 1.

```
****      STANFORD PASCAL POST-PROCESSOR, VERSION OF OCT.-79.  
****      NO ASSEMBLY ERROR(S) DETECTED.  
****      9300 BYTES OF CODE GENERATED,  0.04 SECONDS IN POST_PROCESSING.
```

VS LOADER

OPTIONS USED - PRINT,MAP,NOLET,CALL,RES,NOTERM,SIZE=307200,NAME=**GO

NAME	TYPE	ADDR	NAME	TYPE	ADDR	NAME	TYPE	ADDR	NAME	TYPE	ADDR	NAME	TYPE	ADDR
LATOT001	SD	AC010	LATOL002	SD	AC0A8	LATOE003	SD	AC168	LAT00004	SD	AC228	STAMP006	SD	AC2F0
STAMP007	SD	AC400	STAMP008	SD	AC4A0	STAMP009	SD	AC5A0	STAMP010	SD	AC6B0	STAMP005	SD	AC828
FATT0011	SD	AC9E0	TESTA013	SD	ACB60	ESEMP012	SD	ACC20	KAMEN015	SD	ACF90	INFOR016	SD	AD0B0
LUN10017	SD	AD210	STAMP019	SD	AD2F8	STAMP018	SD	AD498	TESTA020	SD	AD580	ZEROF021	SD	AD8F0
FOND0022	SD	AD998	TESTF014	SD	ADC58	ERROR023	SD	ADE68	\$MAINBLK	SD	ADFF8	IHCLSQRT*	SD	AE490
DSQRT *	LR	AE490	IHCLLOG *	SD	AE5F0	DLOG10 *	LR	AE5F0	DLOG *	LR	AE608	\$PASENT *	SD	AE7F0
\$PASINT *	LR	AED58	IFYVERRM*	LR	AF422	IHCERRM *	LR	AF422	IHOERRM *	LR	AF422	IHNERRM *	LR	AF422
ERRMON *	LR	AF422	IFYVERRE*	LR	AF456	IHOERRE *	LR	AF456	IHNERRE *	LR	AF456	IHCERRE *	LR	AF456
\$PASCSP *	LR	AF460	\$WRADDR *	SD	B0AD8	\$IDLEN *	SD	B0C88	\$GETPNAM*	SD	B0D18	\$ALIGN *	SD	B0EA8
\$ERRMSG *	SD	B0EF0	\$TRANSVA*	SD	B11F8	\$CNVTNUM*	SD	B1280	\$PRNT *	SD	B13C0	\$IVSCAN *	SD	B2150
\$PRNTVAR*	SD	B27C8	\$FROMLIN*	SD	B2D48	\$PRNTLNK*	SD	B2EE8	\$PRNTSYS*	SD	B30A8	SNAPSHOT*	SD	B3B80
#MAINBLK*	SD	B4080	IHCECOMH*	SD	B40E8	IBCOM# *	LR	B40E8	FDIOCS# *	LR	B41A4	INTSWTCH*	LR	B502E
IHCCOMH2*	SD	B5050	SEQDASD *	LR	B53C8	IHCFCVTH*	SD	B56B0	ADCON# *	LR	B56B0	FCVAOUTP*	LR	B575A
FCVLOUTP*	LR	B57EA	FCVZOUTP*	LR	B5942	FCVIOUTP*	LR	B5CF6	FCVEOUTP*	LR	B61F8	FCVCOUTP*	LR	B6412
INT6SWCH*	LR	B66FB	IHCFIOS*	SD	B6868	FIOCS# *	LR	B6868	FIOCSBEP*	LR	B686E	IHCFIOS2*	SD	B7790
IHCEFNTN*	SD	B7CC0	ARITH# *	LR	B7CC0	ADJSWTCH*	LR	B805C	IHCLOPT *	SD	B8208	IHCUTBL*	SD	B8508

TOTAL LENGTH CB30
ENTRY ADDRESS AE7F0

MARTIN GARDNER, SHOW DI MAGIA MATEMATICA

4
790
01600

9 CIFRE DI 12! NEL TESTO A PAGINA 35

MARTIN GARDNER, SHOW DI MAGIA MATEMATICA

1
081
39675
8240290
900504101
30580032964
9720646107774
902579144176636
57322653190990515
3326984536526808240
339776398934872029657
99387290781343681609728
000000000000000000000000

169 CIFRE DI 105! FIGURA 10 A PAGINA 37

[illegible]

1.156 CIFRE DI 508! FIGURA 11 A PAGINA 38

MARTIN GARDNER, SHOW DI MAGIA MATEMATICA

1
033
31479
6638614
492966665
1337523
20000
000
0

41 CIFRE DI 35! NEL TESTO ALLE PAGINE 37 E 38

MARTIN GARDNER, SHOW DI MAGIA MATEMATICA

17108972589718074
1439528307936299026
080765545554532458183
43255130543516432376912
4663791911119657860822050
367340495642348613717749611
38104459104482535212494659899
5225079402598873366451131040234
240130493689852679573590918519290
66647636392705738600295487428650940
0535103538524596394743595531728001643
083783948745781956212836911156587085000
40781396853030778257813849856692950471963
5089328018573725755534194119396813233357487
709737509271413007324171020350516977549843435
61187933295519151457453789138048055187827977590
7750007855795139817496078270462761613125177420579
97170554688538703689036095806399241086011592997
020790226888203087101533653915806041722653430
0377642434651424325601245917031000886439794
86942002854170097571338930915447098888372
333024657251637441276280296188483408232
2723195014038951851520634322622612616
12431271509190879459978732133255390
601413833379281814639023615443036
2338368861798856260050552801204
22598617062894203619648023806
8096438328360960000000000000
0000000000000000000000000000
0000000000000000000000000000
0000000000000000000000000000
0000000000000000000000000000
0000000000000000000000000000

1.073 CIFRE DI 477! FIGURA 12 A PAGINA 39

MARTIN GARDNER, SHOW DI MAGIA MATEMATICA

1025729611602273508352741640301
162679407322869608383074471138746
30490176947592788989158028363440992
4881004274871081412845389545578617964
937451453403559961733083620099625538065
41010396598345035653688604954874028604784
6935414801400008952354538805490908409959861
660246045191650260507136760688118903350996713
21823947514423961017742558259080494678177458085
2586022822207646670820883488582338159458352125708
362194967520635351099568538807160144980504808697217
60361821401601138846975754268977469634297805191393877
0721740965493361497666505546284509191953032898888959567
069201586371013497509110063361133485051253334853981163412
07851202228866097513788527648919064612804715051516533550808
5628864147138448743521002113548909441417847473055608205047228
050888506002351831385707706597375486196289582204458204639226419
39988685801691888740823242074655630309966434682847648783012369690
0949679573713722032502963934989938775237593102275000626771342793017
571237823459954555639364520012107232014386673952152555563687059868597
30561699128318275778511690671009105944054429948225877769250180772939089
5482468231085915745358203314543675656839465279720697501109412886408456325
452410148100374780426054880997809853185180536108897611555630604942279452179
84296686240652468994135036474242868453665349623925844722715297861094652833533
6243076317214023025703473585182809668132851399986647037941978752777468292847695
603640621614001511781845118088875779459341557554605484271581941683408763840520361
23170898026795922847394635627557990754271854045696455075681612990097706515709337585
3609457155707951999956996602370489306141334822832959732146934986498711106505143423006
463954096916259897686557876127375781093939500600724947477213915033139433655578405917439
66529210437027529365792843517778279323319868406572635522503159691924609965063914289434510
7019821565442168516926969752389318995661246814055912699808919159203294055178987900314759208
0442506495067999489407758571492797940488049296440489928011632417128028690854571931206525088
0641218070711371216774493418451658654662421419901954748914071330317835008057673531461465561
2768515633920521576618161998879098027906520489507115754887782147626888335639598691326086662
8591561763561263074922316182055408616512641833590647698679206429535534489790054789746645046
7541595913406494278781641172660577772496746978451472658691427962204322571171000546469639193
2790901159275847484128114857934498847063422270277559602059269990527077182952388889333136592
5112259111214461121960558583085300900857728673001457908605480987931391634861299625358963146
9354877285883646562527594000195845750815246146536989440348537192331062397234558495885351131
0486319190754485743708055141816253285423820670998154301257121647181580736841247433302909291
8978304729883750528094105685468933501043414202907313725590875531859893702065889677849388404
1409345973933364835255409965082222341074607394942553036358497877027519189446464290292514742
8894719069197344614264735325576405364746890544013692524832899026218553163092629405073307432
5883237879257553223715273633812159745601142017086546047693697449507986197545225600849607508
4225614110584212738361884639106673278902989788537027787173903234357897043773626379633919708
2650317682667625547582587775747027139684279901717692515347217940718330972281022406020576955
1923109334083392446906107789059722221024694681063461315307372556082106306372694424873419069
0964340389530663549733052473631156529648322803073354434281448023017979554302704018756348368
8235987859335034065876692541582476375532676240789963230134591939936849075772844192830282673
5947374533149522043979463889533209159979640809623687593870067491951481812767444483279372007
2479979995918726706098991265562174581709978283817056885289631559290783185857144934100631908
9346302000944243304245025830533367966786168167762945518348245854102054733012367306021265699
714441274496819176365724054932016483081835869405802223004571257406858066063952523933901667
7230224963193557961156169360046766935581419151926024167053911783185412817174305853343800679
1914069190803158006986526614048105699422168264057749481333683845736260715525170992590292274
4512642655765465531121253764415693728736415931091968744923431544233653464036812168293962954
9767299210770870181089859892303670680851719783785348200229620448896647181023724949915937363
8125724983730036346858277218777076810719393224395941146613075583124238948926153797306851203
3222908555754737090939960701071272925206188572711353732688814416502259766751986220308762254
0147673137413584101747534961199349889138751673014525041430085687942534349828532365955990586

LINE #	RANGE	RUN CNT	CONSTRUCT	PAGE 1
---	---	---	---	---
34	- 35	3	PROCEDURE: LATOTRIANGOL	
39	- 40	1	PROCEDURE: LATOLOSANGA	
44	- 45	1	PROCEDURE: LATOESAGONO	
49	- 50	1	PROCEDURE: LATOOTTAGONO	
57	- 63	183	PROCEDURE: STAMPALINEA	
60	- 60	8869	FOR STMT	
70	- 72	3	PROCEDURE: STAMPATRIANG	
71	- 72	50	FOR STMT	
79	- 83	1	PROCEDURE: STAMPALOSANG	
80	- 80	5	FOR STMT	
82	- 83	4	FOR STMT	
90	- 94	1	PROCEDURE: STAMPAESAGON	
91	- 91	17	FOR STMT	
93	- 94	16	FOR STMT	
101	- 107	1	PROCEDURE: STAMPAOTTAGO	
102	- 102	31	FOR STMT	
104	- 104	29	FOR STMT	
106	- 107	31	FOR STMT	
110	- 117	6	PROCEDURE: STAMPAFORMA	
111	- 111	3	CASE CLAUSE	
112	- 112	1	CASE CLAUSE	
113	- 113	1	CASE CLAUSE	
114	- 115	1	CASE CLAUSE	
124	- 143	6	PROCEDURE: FATTORIALE	
127	- 141	3337	FOR STMT	
130	- 134	8208427	FOR STMT	
136	- 141	8863	WHILE STMT	
149	- 153	6	PROCEDURE: TESTATAESEMP	
150	- 150	5	THEN CLAUSE	
156	- 176	1	PROCEDURE: ESEMPILIBRO	
192	- 195	0	PROCEDURE: KAMENETSKY	
193	- 193	0	THEN CLAUSE	
194	- 195	0	ELSE CLAUSE	
227	- 234	0	PROCEDURE: INFORMA	
228	- 228	0	CASE CLAUSE	
229	- 229	0	CASE CLAUSE	
230	- 230	0	CASE CLAUSE	
231	- 232	0	CASE CLAUSE	
241	- 247	0	PROCEDURE: LUN1000	
242	- 242	0	THEN CLAUSE	
243	- 247	0	ELSE CLAUSE	
255	- 262	0	PROCEDURE: STAMPA0	
256	- 256	0	CASE CLAUSE	
257	- 260	0	CASE CLAUSE	
265	- 272	0	PROCEDURE: STAMPA1000	
266	- 266	0	THEN CLAUSE	
267	- 272	0	ELSE CLAUSE	
279	- 303	0	PROCEDURE: TESTATA	
282	- 282	0	CASE CLAUSE	
283	- 283	0	CASE CLAUSE	
284	- 284	0	CASE CLAUSE	
285	- 286	0	CASE CLAUSE	
288	- 288	0	THEN CLAUSE	
292	- 292	0	CASE CLAUSE	
293	- 293	0	CASE CLAUSE	

LINE #	RANGE	RUN CNT	CONSTRUCT	PAGE	2
----	----	----	-----	----	----
294	- 294	0	CASE CLAUSE		
295	- 296	0	CASE CLAUSE		
309	- 319	0	PROCEDURE: ZEROFINALI		
313	- 317	0	WHILE STMT		
325	- 349	0	PROCEDURE: FONDO		
329	- 329	0	THEN CLAUSE		
330	- 330	0	ELSE CLAUSE		
337	- 337	0	THEN CLAUSE		
338	- 338	0	ELSE CLAUSE		
340	- 340	0	THEN CLAUSE		
341	- 349	0	ELSE CLAUSE		
346	- 346	0	THEN CLAUSE		
347	- 348	0	ELSE CLAUSE		
352	- 371	0	PROCEDURE: TESTFORMA		
355	- 365	0	FOR STMT		
358	- 365	0	THEN CLAUSE		
367	- 371	0	THEN CLAUSE		
376	- 383	0	PROCEDURE: ERRORE		
378	- 378	0	CASE CLAUSE		
379	- 379	0	CASE CLAUSE		
380	- 381	0	CASE CLAUSE		
390	- 409	1	PROCEDURE: \$MAINBLK		
393	- 393	0	THEN CLAUSE		
395	- 395	1	THEN CLAUSE		
396	- 409	0	ELSE CLAUSE		
398	- 398	0	THEN CLAUSE		
401	- 401	0	THEN CLAUSE		
403	- 403	0	CASE CLAUSE		
404	- 404	0	CASE CLAUSE		
405	- 405	0	CASE CLAUSE		
406	- 407	0	CASE CLAUSE		

JJJJJJJJJ	222222222	AAAAAAAAAAA
JJJJJJJJJ	22222222222	AAAAAAAAAAAAA
JJ	22	AA
JJ	22	AA
JJ	22	AA
JJ	22	AAAAAAAAAAAAA
JJ	22	AAAAAAAAAAAAA
JJ	22	AA
JJ	22	AA
JJ	22	AA
JJJJJJJ	22222222222	AA
JJJJJ	22222222222	AA

****A	END	JOB	2	FF#PAS	ROOM	5.58.01	PM	26	MAY	20	PRINTER1	SYS	TK4-	JOB	2	END	A****
****A	END	JOB	2	FF#PAS	ROOM	5.58.01	PM	26	MAY	20	PRINTER1	SYS	TK4-	JOB	2	END	A****
****A	END	JOB	2	FF#PAS	ROOM	5.58.01	PM	26	MAY	20	PRINTER1	SYS	TK4-	JOB	2	END	A****
****A	END	JOB	2	FF#PAS	ROOM	5.58.01	PM	26	MAY	20	PRINTER1	SYS	TK4-	JOB	2	END	A****