```
begin
comment permanent material
routine spec read
integer fn spec addr
routine spec print (real x, integer m,n)
routine spec print fl (real x, integer n)
routine spec read symbol (integer name n)
integer fn spec next symbol
routine spec skip symbol
routine spec print symbol (integer n)
routine spec newline
routine spec newlines (integer n)
routine spec space
routine spec spaces (integer n)
real fn spec sin (real x)
real fn spec cos (real x)
real fn spec tan (real x)
real fn spec arctan (real x,y)
real fn spec exp (real x)
real fn spec log (real x)
real fn spec sqrt (real x)
real fn spec radius (real x,y)
real fn spec fracpt (real x)
real fn spec mod (real x)
integer fn spec int (real x)
integer fn spec int pt (real x)
integer fn spec parity (integer n)
routine spec tab
routine spec tabs(integer n)
routine spec colour change
routine spec stop code
routine spec run out(integer n)
routine spec newpage
routine spec read binary(integer name n)
routine spec punch binary(integer n)
real map spec real(integer s)
integer map spec integer(integer s)
real fn spec arcsin(real x)
real fn spec arccos(real x)
```

accept m/c instructions

```
        *JS 999
        *J 1000
        *
41P:    *41120/0/0
        *41744/0/0
        *43516/20000/0
        *46737/57020/0
        *55507/6744/157602
        *72716/161326/152025
42P:    *37715/10100/0

        *advca 32
35P:    *0/242/277
        *0/0/302
        *advca 30
44P:    *0/302/337
        *0/0/242

30P:    *0/0/0
31P:    *0/0/0
36P:    *0/0/0

40P:    *0/0/0
        *177777/177777/177777
        *advca 119
43P:    *0/0/0
        *0/0/0

        *advca 32
38P:    *0/542/577
        *0/0/602
        *advca 30
45P:    *0/602/637
        *0/0/542

        *advca 2
```

```
32P:    *0/0/0
        *0/0/0
        *0/0/4
        *0/0/5
        *0/0/2
        *0/0/0
        *0/0/0
        *0/0/0
        *177777/177777/177750
        *177777/177777/177747
        *177777/177777/177741
        *177777/177777/177745
        *177777/177777/177744
        *177777/177777/177743
        *177777/177777/177742
        *0/0/17
        *0/0/20
        *0/0/21
        *0/0/22
        *0/0/23
        *0/0/24
        *0/0/25
        *0/0/26
        *0/0/27
        *0/0/30
        *0/0/31
        *177777/177777/177755
        *177777/177777/177754
        *177777/177777/177753
        *0/0/35
        *0/0/36
        *0/0/37
        *177777/177777/177760
```

```
33P:    *0/0/0
        *0/0/0
        *0/0/0
        *0/0/0
        *0/0/0
        *0/0/0
        *0/0/0
        *0/0/0
        *0/0/0
        *0/0/11
        *0/0/0
        *0/0/0
        *0/0/14
        *0/0/0
        *0/0/0
        *177777/177777/177761
        *0/0/0
        *177777/177777/177757
        *177777/177777/177756
        *0/0/0
        *0/0/0
        *0/0/0
        *177777/177777/177752
        *177777/177777/177751
        *177777/177777/177746
        *0/0/0
        *0/0/32
        *0/0/33
        *0/0/34
        *0/0/0
        *0/0/0
        *0/0/0
        *0/0/0
```

```
39P:    *0/0/117
        *0/0/40
        *0/0/121
        *0/0/122
        *0/0/32
        *0/0/33
        *0/0/34
        *0/0/126
        *0/0/127
        *0/0/10
        *0/0/11
        *0/0/130
        *0/0/13
        *0/0/14
        *0/0/15
        *0/0/16
        *0/0/12

93P:    *24240/43050/10612
        *40407/150101/172020
91P:    *0/0/0
92P:    *0/0/2
```

```
1P:     comment routine entry
        *LINK
        * = MOM12N
        * = M13
        * M11M13
        * = LINK
        * EXIT


2P:     comment print time
        *SET 36P:
        *DUP
        *=M13
        *MOM13
        *=Q15
        *REV
        *J900C15NZ
        *SET 7
        *JS 909
        *ZERO
        *NOT
        *NEG
        *=C15
900:    *SET 93P:
        *=M13
        *MOM13
        *REV
        *FRB
        *ZERO
        *SHLD+6
        *DUP
        *J 901 =Z
        *JS 908
        *ZERO
        *SHLD+6
902:    *JS 908
```

```
        *SET 64         ;comment # HR ##
        *JS 909
        *SET 104
        *JS 909
        *SET 50
        *JS 909
        *SET 64
        *DUP
        *JS 909
        *JS 909
        *ZERO
        *J 903
901:    *SHLD+6
        *DUP
        *J 902 ≠Z
903:    *SHLD+12
        *DUP
        *J 904 =Z
        *JS 908
        *ZERO
        *SHLD+6
905:    *JS 908
        *SET 64         ;comment # MIN ##
        *JS 909
        *SET 109
        *JS 909
        *SET 41
        *JS 909
        *SET 110
        *JS 909
        *SET 64
        *DUP
        *JS 909
        *JS 909
        *ZERO
        *J 906
904:    *SHLD+6
        *DUP
        *J 905 ≠Z
```

```
906:    *SHLD+12
        *DUP
        *J 907 =Z
        *JS 908
        *ZERO
907:    *SHLD+6
        *JS 908
        *ERASE
        *SET 64          ;comment # SEC #
        *JS 909
        *SET 115
        *JS 909
        *SET 37
        *JS 909
        *SET 35
        *JS 909
        *SET 2
        *JS 909
        *=M13
        *Q15
        *=MOM13
        *EXIT 1
```

```
        comment caption sequence
3P:     *SET36P:
        *DUP
        *=M13
        *MOM13
        *=Q15
        *REV
        *I11
        *+
        *=RM14
        *MOM14
        *=C14
        *J401C14Z
        *M+I14
402:    *MOM14Q
404:    *ZERO
        *SHLD+8
        *DUP
        *J403=Z
        *JS47
        *J404
403:    *ERASE
        *ERASE
        *J402C14NZ
        *=M13
        *Q15
        *=MOM13
        *EXIT 1
401:    *ERASE
        *EXIT 1
```

```
4P:     comment return
        * MOM12N
        * = LINK
        * EXIT 1


5P:     comment test for valid cycle
        * PERM
        * DUP
        * J1 = Z
        * PERM
        * -
        * REV
        * + I
        * J1 ≠ Z
        * J1 < Z
        * EXIT 1
1:      caption ↑ NON-INTEGRAL ↓ CYCLE
        -> 2
```

```
6P:2:   comment stop
        newlines(3)
        *SET 92P:
        *=M13
        *MOM13
        *JS 3P
        caption # RUNNING # TIME # # #
        *SET 9
        *OUT
        *SHL-24
        *SET 91P:
        *=M13
        *MOM13
        *-
        *JS 2P
        caption # / # #
        *SET 3
        *OUT
        *SHL-24
        *JS 2P
        *SET 2
        *JS 909
        *SET 2
        *JS 909
        *SET 76
        *JS 909
        *M15
        *DUP
        *SHL-5
        *NOT
        *NEG
        *SHL+5
        *=M15
        *MOM15
        *=Q13
        *=M13
        *POCQ13
        *C13
        *SET 6
        *OUT
```

```
*SET 418
*=M15
*MOM15
*SET 4
*OUT            ;comment pick up compiler tape
*=C15
*PMDQ15         ;comment rewind
*SET 2
*=M15
*PMAQ15         ;comment skip label and sentinel
*SET 8
*=I15
*SET 100
*=M15
*PIAQ15         ;comment read down call
*ZERO
*=LINK
*EXIT 27        ;comment go out to call at 3/13
```

```
7P:     comment -> switch label
        * = M13
        * I11
        * = M10
        * DUP
        * M13M10
        * -
        * J4 < Z
        * DUP
        * M13M10N
        * -
        * J4 > Z
        * M13M10
        * -
        * SET+2
        * +
        * = +M13
        * M13M10
        * DUP
        * J4 = Z
        * = LINK
        * EXIT
4:      caption ≠ SWITCH ≠ VARIABLE ≠ NOT ≠ SET
        -> 2
```

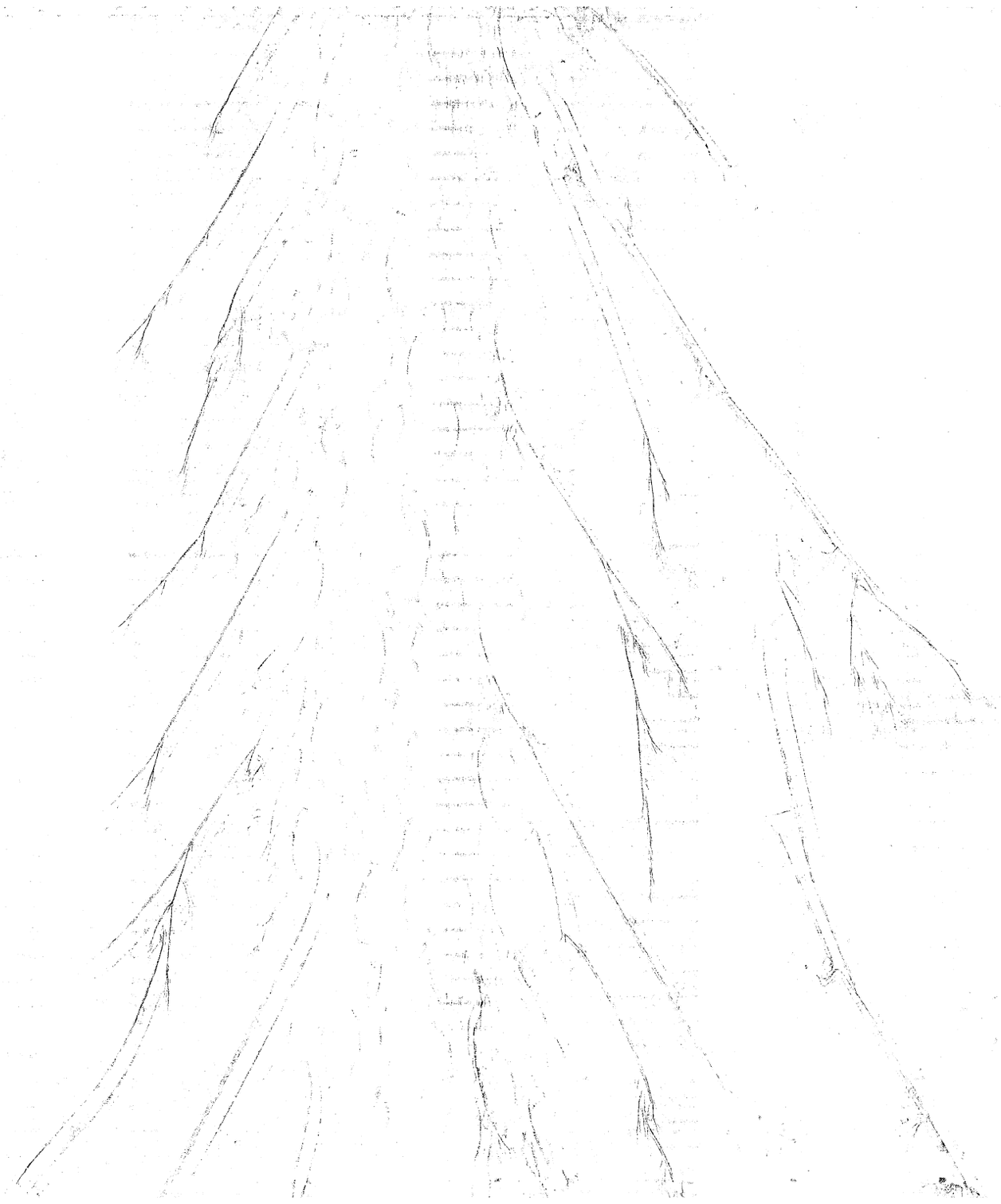
```
8P:     comment OVR set
        caption ∧ OVERFLOW ∧ SET
        -> 2


9P:     comment non-integral integer quotient
        caption ∧ NON-INTEGRAL ∧ QUOTIENT
        -> 2


10P:    comment * n1/n2/n3
        * = M13
        * = I13
        * = C13
        * Q13
        * EXIT 1


11P:    comment exit from fn or map other than by result = etc
        caption ∧ RESULT ∧ NOT ∧ SPECIFIED
        -> 2
```

```
        comment read sequence
12P:    *SET 41P:
        *=M14
        *SET+1000
        *=C14
        *SET 31P:
        *=M15
        *ZERO
        *ZERO
        *ZERO
411:    *ERASE
        *JS49
        *SET+65
        *J411=
        *SET+4
        *J411=
        *DUP
        *SET+16
        *-
        *J415<Z
        *DUP
        *SET+25
        *-
        *J413≤Z
        *SET+29
        *J433=
        *SET+30
        *J414≠
        *CAB
        *ERASE
        *ZERO
        *NOT
        *NEG
        *PERM
        *J433
414:    *SET+31
        *J415≠
        *CO TO Q14
433:    *ERASE
```

```
412:    *J549
        *DUP
        *SET+16
        *-
        *J425<Z
        *DUP
        *SET+25
        *-
        *J416>Z
        *REV
        *MOM14
        *XF
        *J8PV
        *REV
        *DC14
413:    *SET+16
        *-
        *I1
        *FLOAT
        *+F
        *J8PV
        *J412
416:    *SET+31
        *J417=
        *SET+90
        *J418=
425:    *ERASE
        *C14
        *J420<Z
        *ZERO
423:    *=C14
420:    *REV
        *J421=Z
        *NEGF
421:    *C14
        *DUP
        *J434=Z
        *DUP
        *J419>Z
        *NEG
        *J474
        *J424
```

```
419:    *J575
        *J424
415:    begin
        integer char
        caption ß SYMBOL ß ß
        "**=char
        print symbol(char)
        caption ß ß IN ß DATA
        stop
        end
417:    *ERASE
        *C14
        *J420<Z
        *ZERO
        *=C14
        *J412
418:    *ERASE
        *ZERO
        *ZERO
        *ZERO
427:    *ERASE
        *J549
        *SET+65
        *J427=
        *DUP
        *SET+16
        *-
        *J415<Z
        *DUP
        *SET+25
        *-
        *J428>Z
429:    *REV
        *SET+10
        *xD
        *CONT
        *+
        *SET+16
        *-
```

```
430:    *J549
        *DUP
        *SET+16
        *-
        *J422<Z
        *DUP
        *SET+25
        *-
        *J429≤Z
422:    *ERASE
        *REV
        *J431=Z
        *NEG
431:    *C14
        *J423>Z
        *C14
        *+
        *J423
428:    *SET+29
        *J435=
        *SET+30
        *J415≠
        *CAB
        *ERASE
        *ZERO
        *NOT
        *NEG
        *PERM
435:    *ERASE
        *J430
434:    *ERASE
424:    *REV
        *NEG
        *NOT
        *J436=Z
        *DUP
        *FIX
        *DUP
        *J437≤Z
        *=C13
```



```
        *STR
        *SHADC13
        *J8PV
        *REV
        *J441≠Z
        *REV
        *ERASE
436:    *MOM15
        *=Q13
        *M-I13
        *Q13
        *=MOM15
        *REV
        *=M13
        *=MOM13
        *EXIT 1
437:    *ERASE
        *STR
        *ERASE
        *J442≠Z
        *SHA-46
        *J436
441:    *ERASE
442:    begin
        real x
        newline
        **=x
        print fl(x,10)
        caption ß ß INSTEAD ß OF ß INTEGER ß IN ß DATA
        stop
        end
```

```
908:    *SET 16
        *+
        comment print character
13P:    *DUP
        *BITS
        *NOT
        *SHL+47
        *SHL-41
        *+
909:97P:*M+I15
        *M15
        *SET B 37
        *AND
        *J 449 =Z
448:    *=MOM15
        *EXIT 1
449:    *MOM15
        *=Q13
        *POCQ13
        *MOM15N
        *=M15
        *J 448
```

```
        comment divide N2  by 10↑N1
74:14P: *ZERO
        *NOT
        *=RM13
452:    *DUP
        *ZERO
        *NOT
        *NEG
        *AND
        *M+I13
        *REV
        *SHL-1
        *PERM
        *J451=Z
        *M13M14
        *÷F
451:    *REV
        *DUP
        *J452≠Z
        *ERASE
        *EXIT 1

        comment multiply N2 by 10↑N1
75:15P: *ZERO
        *NOT
        *=RM13
462:    *DUP
        *ZERO
        *NOT
        *NEG
        *AND
        *M+I13
        *REV
        *SHL-1
        *PERM
        *J461=Z
        *M13M14
        *xF
        *J8PV
461:    *REV
        *DUP
        *J462≠Z
        *ERASE
        *EXIT 1
```

```
        comment set up for print routines
16P:    *SET 36P:
        *DUP
        *=M13
        *MOM13
        *=Q15
        *REV
        *SET 41P:
        *=M14
        *J 471 C15 NZ
        *SET+7
        *JS 97P
        *ZERO
        *NOT
        *NEG
        *=C15
471:    *ZERO
        *REV
        *DUP
        *J472>Z
        *NEOF
        *REV
        *ERASE
        *SET+30
        *REV
472:    *EXIT 1
```

```
comment print N1 digits of N2
17P:    *REV
        *FIX
        *=C13
        *ZERO
        *SHAD C13
        *J481
18P:    *REV
        *SET+10
        *xD
481:    *JS 908
        *REV
        *NEG
        *NOT
        *DUP
        *J18P≠Z
        *ERASE
        *EXIT 1
```

```
        comment standardise acc
19P:    *DUP
        *FIX
        *REV
        *ERASE
        *I1
        *FLOAT
        *SET 42P:
        *=M13
        *MOM13
        *xF
        *FIX
        *=C13
        *STR
        *SHADC13
        *REV
        *ERASE
        *NEG
        *NOT
        *DUP
        *J493#Z
        *REV
        *J494
493:    *DUP
        *J495>Z
        *DUP
        *PERM
        *NEG
        *JS 15P
        *J494
```

```
495:    *DUP
        *PERM
        *JS 14P
494:    *DUP
        *MOM14
        *-F
        *J496<Z
        *MOM14
        *+F
        *REV
        *NOT
        *NEG
        *EXIT 1
496:    *REV
        *EXIT 1
```

```
20P:    begin
        comment Query Print integer
        integer QI
        * DUP
        ** = QI
        newline
        print (QI,1,0)
        end
        * EXIT 1

21P:    begin
        comment Query print real
        real QR
        * DUP
        ** = QR
        newline
        print fl(QR,10)
        end
        * EXIT 1

22P:    caption /\ SQRT / -VE
5:      * JS21P
        -> 2

23P:    caption /\ LOG / -VE    ; -> 5

24P:    caption /\ EXP / TOO / LARGE   ; -> 2

25P:    caption /\ TAN / TOO / LARGE   ;  -> 2

26P:    caption /\ ARCTAN(0,0)   ;  -> 2

27P:    caption /\ TRIG / FUNCTION / INACCURATE   ;  -> 2

28P:    caption /\ INT / OR / INTPT / TOO / LARGE   ; -> 2
```

```
46P:999:*ZERO           ;comment reset output device number
        *NOT
        *NEG
        *J 920 C11 NZ
        *SET 8
        *+
920:    *SET 5
        *OUT
        *DUP
        *SET 38 P:
        *=M13
        *MO M13
        *=Q15
        *=C15
        *Q15
        *=MO M13
        *SET 45 P:
        *=M13
        *MO M13
        *=Q15
        *=C15
        *Q15
        *=MO M13

        *SET 2          ; comment reset input device number
        *J 921 C11 NZ
        *SET 8
        *+
921:    *SET 5
        *OUT
        *DUP
        *SET 35 P:
        *=M 13
        *MO M13
        *=Q15
        *=C15
        *Q15
        *=MO M13
        *SET 44 P:
        *=M13
        *MO M13
        *=Q14
        *=C14
        *Q14
        *=MO M13
```

```
        *SET 47
        *=I1


        *J 400 C11 Z
        *EXIT 1


400:    *PICQ15
        *Q14
        *ZERO           ;comment set output buffer pointer to Q15
        *NOT
        *NEG
        *DUP
        *=C15
        *=I15
        *SET 353
        *=M15

        *SET 50                 ; comment put out 5'' run-out
        *=C14
        *SET 7
998:    *DUP
        *JS 97 P
        *DC14
        *J998 C14 NZ
        *ERASE

        *SET 2
        *JS 97P
        *SET 2
        *JS 97P
```

```
        *SET 7
        *=M13
        *MOM13          ;comment date
        *SET 11
        *=C14
997:    *ZERO
        *SHLD+6
        *JS 13P
        *DC14
        *J997 C14 NZ
        *ERASE
        *SET 93P:       ;comment time from midnight
        *=RM13
        *MOM13Q
        *SET 9
        *OUT
        *SHL-24
        *DUP
        *=MOM13N
        *FRB
        *MOM13
        *OR
996:    *ZERO
        *SHLD+6
        *JS 13P
        *DUP
        *J996#Z
        *ERASE
```

```
*SET 36P:                ;comment dump output buffer pointer
*=M13
*Q15
*=MOM13

caption ΔΔ EDINBURGH Δ UNIVERSITY Δ ATLAS Δ AUTOCODE ΔΔΔ

*=Q15
*PICQ15
*ZERO            ;comment set input buffer pointer
*NOT
*NEG
*DUP
*=C15
*=I15
*SET 161
*=M15
*SET 30P:
*=M13
*Q15
*=MOM13

*SET 40 P:               ; comment set up reconstructed line pointer
*=RM 15
*SET 31 P:
*=M13
*Q15
*=MO M13


*EXIT 1
```

```
        comment print single symbol
47:47P: *DUP
        *SET+97
        *-
        *J501>Z
        *DUP
        *SET+64
        *-
        *DUP
        *J502>Z
        *SET+31
        *+
        *J503>Z
        *SET 32P:
        *J504
501:    *SET+64
        *-
        *J 505C15Z
508:    *SET+70
        *JS 97P
        *CO TO Q15
505:    *JS 13P
        *EXIT 1
502:    *REV
        *ERASE
        *SET 33P:
504:    *+
        *=M13
        *MOM13
```

```
        *DUP
        *J506<Z
        *DUP
        *SET+12
        *-
        *J507≤Z
        *J507C15NZ
509:    *SET+7
        *JS 97P
        *ZERO
        *NOT
        *NEG
        *=C15
507:    *JS 13P
        *EXIT 1
506:    *NEG
        *J 508 C15NZ
        *JS 13P
        *EXIT 1
503:    *J509 C15Z
        *JS 13P
        *EXIT 1
```

```
        comment line reconstruct
48P:    *Q14
        *Q15

        begin
        integer char
        switch h(0:14)
        *SET 30P:
        *=M13
        *MOM13
        *=Q14
        *SET 40P:
        *=RM15
        *M15
        *NEG
        *=C15
        *J511
515:    *MOM14
        *=Q13
        *PICQ13
        *MOM14N
        *=M14
        *J527
511:    *M+I15
520:    *M+I14
        *M14
        *SET B 37
        *AND
        *J 515=Z
527:    *MOM14
        *DUP
        *BITS
        *SHC-1
        *J514<Z
        caption ∧ PARITY ∧ FAULT ∧ ON ∧ TAPE
        stop
```

```
514:    *SETB 77
        *AND
        *DUP
        *SET+14
        *-
        *J512>Z
        **=char
        ->h(char)
h(0):   *M15
        *C15
        *+
        *J511≤Z
        *M15
        *SET 43P:
        *-
        *J526 ≥Z
        *DC15
        *SET+65
        *=MOM15
        *J511
h(1):h(8):h(10):h(11):h(13):h(14):caption ∧ UNASSIGNED ∧ CHAR ∧ ON ∧ TAPE
        stop
526:    caption ∧ > ∧ 120 ∧ CHARS ∧ ON ∧ LINE
        stop
h(4):   *M15
        *SET 40P:
        *DUP
        *PERM
        *-
        *SHL-3
        *DUP
        *SET 3
        *-
```

```
        *J 517 ≤Z
        *I15
        *OR
        *SET 15
        *J 526=
517:    *NOT
        *NEG
        *SHL+3
        *+
        *C15
        *NEG
        *DUP
        *=M15
        *REV
        *DUP
        *PERM
        *-
        *J528≥Z
529:    *M+I15
        *SET+65
        *=MOM15
        *M15
        *J529≠
        *NEG
        *=C15
        *J511
528:    *=M15
        *J511
h(5):   *M-I15
        *M15
        *SET 40P:
        *-
        *J520≠Z
        *M+I15
        *J520
```

```
h(6):   *CO TO Q14
        *J520
h(7):   *ZERO
        *NOT
        *NEG
        *=C14
        *J520
512:    *C14
        *J516=Z
        *DUP
        *SET+26
        *-
        *J513<Z
        *DUP
        *SET+28
        *-
        *J530<Z
        *SET+63
        *J513≠
h(9):h(12):530:*SET+64
        *+
        *J513
516:    *SET+64
        *+
        *DUP
        *SET+95
        *-
        *J513>Z
        *SET+79
        *-
        *SET 39P:
        *+
        *=M13
        *MOM13
```

```
513:    *M15
        *C15
        *+
        *J518<Z
        *M15
        *SET 43P:
        *-
        *J526>Z
        *DC15
519:    *=MOM15
        *J511
518:    *SET+127
        *J519=
        *MOM15
        *SET+65
        *J531≠
        *ERASE
        *=MOM15
        *J511
531:    *SET+127
        *J532≠
533:    *ERASE
        *ERASE
        *J511
532:    *DUP
        *=Q13
        *SETB 177
        *AND
        *DUP
        *PERM
        *J533=
        *REV
        *Q13
        *J521≠
        *DUP D
        *-
        *J522<Z
        *REV
```

```
522:    *SHL+7
        *+
        *=MOM15
        *J511
521:    *REV
        *Q13
        *SHL+34
        *SHL-41
        *DUP
        *PERM
        *J534≠
535:    *ERASE
        *ERASE
        *ERASE
        *J511
534:    *Q13
        *SHL-14
        *DUP
        *J523≠Z
        *ERASE
        *CAB
        *DUP D
        *-
        *J524>Z
        *ERASE
        *Q13
        *SHL+7
        *+
        *=MOM15
        *ERASE
        *J511
```

```
524:    *PERM
        *DUP D
        *-
        *J525<Z
        *REV
        *SHL+7
        *+
        *SHL+7
        *+
        *=MOM15
        *J511
525:    *SHL+14
        *Q13
        *+
        *=MOM15
        *ERASE
        *ERASE
        *J511
523:    *J535=
        caption ≠ > ≠ 3 ≠ SYMBOLS ≠ IN ≠ POSITION
        stop
536:    *ERASE
        *M-I13
        *J 537
```

```
h(2):h(3):*C15
          *NEG
          *=RM13
537:      *MOM13
          *SET+65
          *J 536=
          *SET 127
          *J
             536=
          *ERASE
          *M+I13
          *SET+4
          *=MOM13
          *M+I13
          *ZERO
          *NOT
          *=MOM13
          *SET 30P:
          *=M13
          *Q14
          *=MOM13
          *SET 31P:
          *=M13
          *SET 40P:
          *=RM15
          *Q15
          *=MOM13
          end
          *=Q15
          *=Q14
          *EXIT 1
```

```
        comment read character
541:    *ERASE
         *J542
49:49P: *MOM15
         *=Q13
542:     *M+I13
         *MOM13
         *SET+127
         *J541=
         *DUP
         *J 543 <Z
544:     *Q13
         *=MOM15
         *EXIT 1
543:     *ERASE
         *JS 48P
         *J 49
```

```
66P:    *=M13
        *I13=-1
        *MOM13
        *DUP
        *=+M13
        *=C13
        *MOM13Q
        *
934:    *+
        *REV
        *MOM13Q
        *xD
        *CONT
        *J934C13NZS
        *+
        *M13
        *+
        *EXIT 1
```

```
69P:    *advca M12
67P:    *SET 69P:
        *=RM13
        *DUP
        *=MOM13
        *DUP
        *NEG
        *NOT
        *=+M13
        *=C13
        *ZERO
        *DUP
        *NOT
        *NEG
931:    *M-I12
        *M-I12
        *DUP
        *MOM12
        *xD
        *CONT
        *CAB
        *+
        *REV
        *MOM12N
        *MOM12
        *-
        *NOT
        *NEG
        *xD
        *CONT
        *DC13
        *J932C13Z
        *DUP
        *=MOM13
        *M-I13
        ->931
```

```
932:    *MOM13
        *DUP
        *=+M13
        *CAB
        *NEG
        *NOT
        *NEG
        *+
        *=MOM13
        *=MOM13N
        *EXIT 1

68P:    *M12
        *SET 69P:
        *=RM13
        *MOM13
        *NOT
        *NEG
        *=C13
        *
933:    *MOM13Q
        *=MOM12Q
        *
        *J933C13NZS
        *MOM13
        *REV
        *EXIT 1
```

```
81P:    comment exponent in real expression
        *DUP
        *ABS
        *DUP
        *=C13
        *SHL-8
        *J17#Z
        *ZERO
        *SIGN
        *=I13
        *=Q10
        *I12
        *I1
        *FLOAT
        *I13
        *J12=Z
        *

11:     *Q10
        *xF
        *DC13
        *
        *J11C13NZS

        *I13
        *J12>Z
        *I12
        *I1
        *FLOAT
        *REV
        *+F
12:     *EXIT 1
```

```
82P:    comment exponent in integer expression
        *DUP
        *SHL-6
        *J17≠Z
        *=C13
        *=Q10
        *I12
        *C13
        *J16=Z
        *

15:     *Q10
        *xD
        *CONT
        *DC13
        *J15C13NZS

16:     *EXIT 1

17:     caption ¤ ILLEGAL ¢ EXPONENT
        *J5 20P
        stop
        comment end of exponentiation

89P:    *0/0/0
        *0/0/0
        *0/0/0
        *177777/177777/177777
        *0/0/0
        *177777/177777/177777
        *0/0/0
        *177777/177777/177777
        *0/0/0
        *177777/177777/177777
        *0/0/0
        *177777/177777/177777
```

```
98P:    *SET+16
        *=C14
76:     *ZERO
        *SHLD+3
        *JS 908
        *DC14
        *J76C14NZ
        *ERASE
        *EXIT 1

99P:    *SET 36P:
        *DUP
        *=M13
        *MOM13
        *=Q15
        *REV
        *SET+2
        *JS97P
        *SET+2
        *JS97P
        *J 77 C15 NZ
        *ZERO
        *NOT
        *NEG
        *=C15
        *SET+7
        *JS97P
```

```
77:     *SET73
        *JS97P
        *LINK
        *DUP
        *=LINK
        *JS98P
        *SET 64
        *DUP
        *JS97P
        *JS 97P
        *JS98P
        *SET+73
        *JS97P
        *SET+2
        *JS97P
        *=M13
        *Q15
        *=MOM13
        *EXIT 1
```

```
routine read symbol(integer name n)
        *SET 31P:
        *=M15
        *JS 49P
        **=n
end

routine skip symbol
        *SET 31P:
        *=M15
        *JS 49P
        *ERASE
end

integer fn next symbol
        *SET 31P:
        *=M15
        *JS 49P
        *MOM15
        *NEG
        *NOT
        *=MOM15
        return
end
```

```
routine print (real x, integer m,n)
        **x
        *JS 16P
        *SET B 401
        *SHL+38
        **n
        *JS 14P
        *+F
        *JS 19P
        **m
        *DUP
        *J3>Z
        *ERASE
        *ZERO
        *NOT
        *NEG
3:      *REV
        *DUP
        *J4<Z
        *DUP
        *PERM
        *-
        *NEG
        *NOT
        *DUP
        *J1>Z
2:      *ERASE
        *CAB
        *JS 13P
        *NOT
        *NEG
        *JS 17P
        **n
        *DUP
        *J10≤Z
        *SET+31
        *JS 13P
        *JS 18P
```

```
12:     *ERASE
        *=M13
        *Q15
        *=MOM13
        return
4:      *REV
6:      *NEG
        *NOT
        *DUP
        *J5=Z
        *ZERO
        *JS 13P
        *J6
5:      *ERASE
        **n
        *J9>Z
        *ZERO
        *JS 13P
        *SET+16
        *JS 13P
        *ERASE
10:     *ERASE
        *J 12
9:      *CAB
        *JS 13P
        *SET+16
        *JS 13P
        *SET+31
        *JS 13P
        *DUP
        **n
        *DUP
        *PERM
        *+
        *J 11< Z
        *REV
        *DUP
8:      *NOT
        *NEG
        *DUP
        *J7=Z
```

```
        *SET+16
        *JS 13P
        *J8
7:      *ERASE
        *+
        *NOT
        *NEG
        *DUP
        J10=Z
        *JS17P
        *J 12
11:     *SET+16
        *JS 13P
        *NEG
        *NOT
        *DUP
        *J11≠Z
        *ERASE
        *ERASE
        *J 12
1:      *ZERO
        *JS 13P
        *NEG
        *NOT
        *DUP
        *J2=Z
        *J1
        end
```

```
routine print fl(real x, integer n)
        **x
        *JS 16P
        *JS 19P
        *REV
        *SETB 401
        *SHL+38
        **n
        *JS14P
        *+F
        *DUP
        *MOM14
        *-F
        *J3<Z
        *MOM14
        *+F
        *REV
        *NOT
        *NEG
        *REV
3:      *CAB
        *JS 13P
        *ZERO
        *NOT
        *NEG
        *JS 17P
        **n
        *J4≤Z
        *SET+31
        *JS 13P
        **n
        *JS 18P
4:      *ERASE
        *SET+26
        *JS 13P
        *ZERO
        *REV
```

```
        *DUP
        *J5>Z
        *NEG
        *REV
        *ERASE
        *SET+30
        *REV
5:      *SET+10
        *DUP
        *PERM
        *-
        *DUP
        *J6<Z
        *CAB
        *JS 13P
        *SET+16
        *=RM13
7:      *M+I13
        *REV
        *DUP
        *PERM
        *-
        *DUP
        *J7>Z
        *M13
        *JS 13P
8:      *+
        *SET+16
        *+
        *JS 13P
        *=M13
        *Q15
        *=MOM13
        return
6:      *ZERO
        *JS 13P
        *CAB
        *JS 13P
        *J8
end
```

```
routine print symbol(integer n)
        *SET 36P:
        *=M14
        *MOM14
        *=Q15
        **n
2:      *DUP
        *SET B 177
        *AND
        *JS 47P
        *SHL-7
        *DUP
        *J1=Z
        *SET69
        *JS 97P
        *J2
1:      *ERASE
        *Q15
        *=MOM14
end
```

```
34P:    *SET 36P:
        *=M14
        *MOM14
        *=Q15
        *JS 97P
        *Q15
        *=MOM14
        *EXIT 1

routine space
        *SET 64
        *JS 34P
end

routine newline
        *SET 2
        *JS 34P
end

routine tab
        *SET 4
        *JS 34P
end

routine colour change
        *SET 73
        *JS 34P
end

routine stop code
        *SET 76
        *JS 34P
end
```

```
37P:    *SET 36P:
        *DUP
        *=M13
        *PERM
        *MOM13
        *=Q15
        *=C14
        *J551
552:    *DUP
        *JS 97P
        *DC14
551:    *C14
        *J 552>Z
        *ERASE
        *=M13
        *Q15
        *=MOM13
        *EXIT 1

routine spaces(integer n)
        *SET 64
        **n
        *JS 37P
end

routine newlines(integer n)
        *SET 2
        **n
        *JS 37P
end

routine tabs(integer n)
        *SET 4
        **n
        *JS 37P
end
```

```
routine run out(integer n)
        *SET 7
        **n
        *JS 37P
        *ZERO
        *NOT
        *NEG
        *=C15
        *Q15
        *=MOM13
end

routine newpage
        *SET 2
        *SET 30
        *JS 37P
end
```

```
routine read binary(integer name n)
        *SET 3OP:
        *DUP
        *=M13
        *MOM13
        *=Q14
        *M+I14
        *M14
        *SETB 37
        *AND
        *J1=Z
2:      *ZERO
        *MOM14
        *SHLD-4
        *SHC-2
        *SHLD-1
        *SHL-45
        *SHLD+5
        **=n
        *ERASE
        *=M13
        *Q14
        *=MOM13
        return
1:      *MOM14
        *=Q15
        *PICQ15
        *MOM14N
        *=M14
        *J2
end
```

```
routine punch binary(integer n)
        *SET 36P:
        *=M14
        *MOM14
        *=Q15
        *ZERO
        **n
        *SHLD-4
        *SHC-1
        *SHLD-2
        *SHL-45
        *SHLD+6
        *JS 97P
        *ERASE
        *Q15
        *=MOM14
end


routine read
end

integer fn addr
end

real map real(integer s)
        **s
        return
end

integer map integer(integer s)
        **s
        return
end
```

```
real fn arcsin(real x)
        **arctan(sqrt(1-x²),x)
        return
end

real fn arccos(real x)
        **arctan(x,sqrt(1-x²))
        return
end


7OP:    begin
        comment SUM SERIES, FIXED POINT, SHIFTED
        integer WO
        * = Q15 ; * = C14 ; ** = WO ; *MOM15Q ; *J2
        *
1:      ** WO ; * xD ; * SHADC14 ; * ROUND ; * MOM15Q ; * +
2:      * J1C15NZS ;
        end
        * EXIT 1
```

```
        real fn sin (real x)
        real V3 ; integer V0
        * J51 ; * J527P ; return
        *

71P:    * 0/101/174200
        * 177777/170326/134524
        * 2/120402/134452
        * 177663/45504/141225
        * 2431/127433/7626
        * 153250/103063/113073
        * 62207/166521/10402

1:      V3 = 0.6366 1977 2368
        * SET 7 ; * SET 1 ; * SET 71P : ; * J510P ; ** = V0
        ** x ; ** V3 ; * xF
2:      * ZERO ; * J10NV ; * NOT
10:     * = M14 ; * FIX ; * DUP ; * J8<Z
        * DUP ; * SET29 ; * - ; * J9>Z
        * = RC15 ; * DC15 ; * SHLC15 ; * I15 ; * SHC-2
        * DUP ; * PERM ; * + ; * ABS ; * REV
        * - ; * I15 ; * VR

8:      * FLOAT ; * DUP ; * DUP ; * x F ; * FIX
        ** V0 ; * J570P ; * SET +1 ; * FLOAT ; * xF
        * J511 ; * EXIT 2

9:      * ERASE ; * ERASE

11:     * M14 ; * SHA+48 ; * ERASE ; * EXIT 1
        end


        real fn cos (real x)
        result = sin (x + 1.5707 9632679)
        end
```

```
        real fn tan (real x)
        real y,V3,V2; integer V0

        y=1
        x=π*fracpt(x/π)
        if x<0.5π-1α-10 then ->31
        unless x>0.5π+1α-10 then ->21
        x=π-x
        y=-1
        ->31
21:     *JS 25P
31:     * JS1 ;*JS27P ;*JS25P ; **y ; *xF ; return
        *
72P:    * 0/12775/77742
        * 177777/162203/20046
        * 0/111742/61362
        * 1/10450/170205
        * 5/30271/133230
        * 24/54163/72425
        * 121/77004/102564
        * 505/177762/5055
        * 2431/127431/15065
        * 12253/136346/23211
        * 62207/166521/10402

1:      V3 = 1.273239544737
        * SET 11 ; * SET 1 ; * SET 72P: ; * JS10P ; ** = V0
        V2 = 1
        ** x ; ** V3 ; * ZERO ; * DUP ; * = I14 ; * J12NV ; * NOT
12:     * = M14 ; * xF ; * FIX ; * DUP ; * J7≤Z
        * DUP ; * SET 29 ; * - ; * J10>Z ; * = RC15
        * DC15 ; * SHLC15 ; * DUP ; * I15 ; * SHC-1
        * - ; * J9=Z ; * DUP ; * J8=Z ; *DUP
        * DUP ; * J3<Z ; * SHA+1 ; * J4V ; * J5
3:      * SHA+1 ; * J5V
4:      * ERASE ; * I15 TO Q14 ; * J6
5:      * ERASE
```

```
6:      * I15 ; * SHC-2 ; * DUP ; * PERM ; * +
        * ABS ; * REV ; * - ; * I15

7:      * FLOAT ; * DUP ; * DUP ; * xF ; * FIX ; ** V0 ; * VR ; * JS70P ; * ZERO ; * FLOAT
        * xF ; * I14 ; * J11 = Z ; ** V2 ; * REV ; * ↑ F
11:     * J9V
8:      * JS13 ; * EXIT 3 ;
9:      *ERASE ; * JS13 ; * EXIT 2 ;

10:     *ERASE ; *ERASE
13:     * M14 ; * SHA+48 ; * ERASE ; * EXIT 1

        end



        real fn sqrt (real x)
        integer V0
        * SET B 7333 ; * SET B 73263 ; * SET B 135603 ; * JS10P \\\\ ; ** = V0
        ** x ; * DUP ; *J1<Z ; * FIX ; * DUP ; * SHLD-1 ; * = M15
        * J2<Z ; * SHL-1
2:      * SHA-3 ; * DUP ; * SHA+1 ; ** V0 ; * +
        * DUPD ; * ↑ ; * + ;
        * DUPD ; * ↑ ; * REV ; * SHA-2 ; * +
        * DUP ; * PERM ; * ↑ ; * +
        * M15 ; * NOT ; * NEG ; * FLOAT

3:      return

1:      * DUP ; * J3=Z ; * JS22P
        end
```

```
        real fn log (real x)

        * J82 ;*J823P ; return
        *
73P:    * 123506/172004/13421
        * 26501/36314/177121
        * 30522/126670/151645
        * 2345/76167/33656
        * 61335/71445/63660
        * 160501/165330/173614
        * 11123/161232/171026

2:      * SET 73P: ; * = RM14
        ** x ; * DUP ; * J1>Z ; * EXIT 1

1:      * FIX ; * SHA+40
        * MOM14Q ; * DUP ; * SHA-8 ; * PERM
        * x ; * - ; * REV ; * SHA-1
        * MOM14Q ; * DUPD ; * - ; * PERM
        * + ; * + ; * DUP ; * SHA-1 ; * DUP
        * x ; * DUP ; * MOM14Q ; * -
        * MOM14Q ; * REV ; * +
        * MOM14Q ; * + ; * -
        * MOM14Q ; * REV ; * +
        * MOM14Q ; * + ; * x ; * SHA-5
        * + ; * SET+7 ; * FLOAT ; * EXIT 2
        end
```

```
        real fn exp (real x)
        real V5

        * J84 ; * J5V ; return
5:      * J824P
        *
74P:    * 56125/16624/127014
        * 21566/44437/1054
        * 47654/77167/65123
        * 53562/33342/7501
        * 154526/134640/150035

4:      V5 = 88
        * SET 74P: ; * = RM14
        ** x ; * DUP ; * ABSF ; ** V5 ; * -F ; * J2>Z
        * FIX ; * NOT ; * NEG ; * = C15 ; * MOM14Q ; * x ; * STR
        * SHADC15 ; * DUPD ; * ROUND ; * DUP ; * = Q15
        * - ; * CONT ; * DUP ; * DUP ; * x ; * SHA-7 ; * DUP
        * MOM14Q ; * x; * MOM14Q ; * + ; * REV
        * MOM14Q ; * + ; * MOM14Q ; * REV ; * + ; * + ; * REV
        * SHA-4 ; * DUPD ; * + ; * PERM ; * - ; * SHA+1
        * + ; * Q15 ; * NOT ; * NEG ; * REV ; * DUP ; * J1+Z ; * SHLD-48

1:      * ZERO ; * FLOAT ; * FIX ; * SET+128 ; * + ; * CAB ; * + ; * STR ; * NOT
        * DUP ; * CAB ; * SET+128 ; * - ; * AND ; * PERM ; * AND ; * REV
        * DUP ; * SHA+40 ; * ERASE ; * FLOAT ; * EXIT 1

2:      * J3>Z ; * ZERO ; * EXIT 1

3:      * ZERO ; * NOT ; * SHA+48 ; * EXIT 1
        end

        real fn mod (real x)
        ** x
        * ABSF
        return
        end
```

```
        real fn arctan (real x,y)
        real z ; integer W0, W1
        if x ≠ 0 then -> 10
        if y ≠ 0 then -> 11
        * JS26P
11:     z = 0.5π
        -> 20
10:     if y ≠ 0 then -> 12
        z = 0
        -> 20
12:     z = |y/x| ; ** z ; * JS13 ; ** = z
20:     if x ≥ 0 and y ≥ 0 then result = z
        if x ≥ 0 and y < 0 then result = -z
        if y>0 then result = z - π
        result = z + π
        *
75P:    * 25303/70125/67763
        * 15202/74631/176357
        * 6272/175360/25230
76P:    * 171557/1125/156752
77P:    * 2122/46626/27222
79P:    * 175067/33606/47022
        * 3434/27617/135077
        * 173333/67307/12122
        * 6314/146312/106313
        * 165252/125252/126305
78P:    * 040000/0/0

13:     * SET B 403 ; * SHA+38
1:      * DUPD ; * SHLD+9 ; ** = W0 ; * ERASE
        * ABSF ; * REV ; * ABSF ; * ZERO ; * J4NV ; * NOT
4:      ** = W1 ; * MAXF ; * + F ; * FIX
        * = C15 ; * DC15 ; * SHAC15 ; * SET 3 ; * = RC15
2:      * SET 75P: ; * = M13
        *
22:     * M13M15Q ; * DUMMY ; * DUPD ; * -
        * J3≥ Z ; * ERASE
        * J22C15NZS ; * ZERO ; * DC15
3:      * C15 ; * NOT ; * SET 76 P: ; * = M13 ; * MOM13 ; * xD ; * CONT ; * PERM
        * DUPD ; * - ; * PERM ; * x ; * SHA+1 ; * SET 78P: ; * = M13 ; * MOM13
        * + ; * + ; * SET 6 ; * = RC14; * DUP ; * DUP
```

```
        * x ; * = Q15 ; * SET 77P: ; * = M13 ; * MOM13
5:      * SET 79P: ; * = M13
        *
55:     * Q15 ; *x ; * M13M14Q ; * DUMMY ; * + ; * J55C14NZS
        *x ; * + ; * J6V ; * SET 76P: ; * = M13 ; * MOM13 ; * SHA+3 ;  * + ; * NEG
6:      ** W0 ; * DUP ; * PERM ; * J7≤Z ; * SET 76 P: ; * = M13
        * MOM13 ; * SHA+4 ; * + ; * NEG
7:      * REV ; * SHC-9 ; * J8≥Z ; * NEG
8:      * I14 ; * FLOAT ; ** W1 ; * SHA+48 ; * ERASE ; * EXIT 1
        end



        real fn radius (real x,y)
        result = sqrt (x²+y²)
        end



        real fn fracpt (real x)

        **x ; * FIX ; * DUP ; * SET 39 ; * - ; * J1<Z ; * ERASE ; * ERASE
        * ZERO ; -> 2
1:      * = C15 ; * STR ; * SHADC15 ; * ERASE ; * ZERO ; * FLOAT
2:      return
        end


        integer fn int (real x)
        * SETB 401  ; *SHL+38  ;  comment ½ to NS
        **x ; * +F ; * FIX ; * DUP ; * SET 39 ; *- ; * J1≤Z
        * ERASE ; * ERASE ; * JS28P
1:      * = C15 ; * STR ; * SHADC15 ; * REV ; * ERASE ; return
        end
```

```
        integer fn intpt (real x)
        **x ; * FIX ; * DUP ; * SET 39 ; * - ; * J1<Z
        * ERASE ; * ERASE ; * J828P
1:      * = C15 ; * STR ; * SHADC15 ; * REV ; * ERASE ; return
        end


        integer fn parity ( integer n)
        * SET +1
        ** n
        * SHC-1
        * J1>Z
        * NEG
1:      return
        end
```

```
        begin
61P:    comment entry sequence
1:      *  PIAQ 1  ; comment read one block
        *  SET 100
        *  = RC9
        *  ZERO
        *  VR
3:      *  M8M9Q
        *  +
        *  J2NV
        *  NOT
        *  NEG
2:      *  J3C9NZ
        *  MOM7N
        *  J199 /
        *  +
        *  J4NV
        *  NOT
        *  NEG
4:      *  SET 100
        *  = RC9
        *  MOM7
        *  NEG
        *  NOT
        *  DUP
        *  J100<Z
        *  DUP
        *  J110 = Z
        *  SET 2
        *  -
        *  DUP
        *  J120<Z
        *  DUP
        *  J130=Z
        *  NEG
        *  NOT
        *  J140=Z
        -> 170
```

```
100:    *  ERASE ; comment program
101:    *  SET 100
        *  M5
        *  +
        *  M6
        *  -
        *  J104<Z
        *  JS10
        -> 101
        *
65P:    * 0/0/160000
        * 0/0/177777
64P:    * 4170/60632/15154 ; comment AA FAIL
        * 103020/222/16441  ; comment AA DATA
104:    *  M8M9Q
        *  MOM5
        *  OR
        *  = MOM5Q
        *  J104 C9 NZS
        -> 1

120:    *  ERASE  ; comment set
124:    *  JS30
121:    *  DC3
        *  DUP
        *  J122=Z
        *  JS40
        *  MOM13N
        *  AND
        *  JS50
123:    *  J121 C3 NZ
        *  J124 C9 NZ
        -> 1
122:    *  ERASE
        *  ERASE
        -> 123
```

```
110:    *  ERASE  ;  comment labels
114:    *  J530
111:    *  DC3
        *  DUP
        *  J112=Z
        *  J540
        *  DUP
        *  DUP
        *  SET B 7777
        *  AND
        *  CAB
        *  SET B 10000

        *  AND
        *  SHL 7
        *  CAB
        *  MOM13
        *  AND
        *  SHL 3
        *  OR
        *  OR
        *  J550
113:    *  J111 C3 NZ
        *  J114 C9 NZ
        -> 1
112:    *  ERASE
        *  ERASE
        -> 113


130:    *  ERASE  ; comment stack
        *  I11
        *  J60≠ Z
        *  M5
        *  = I11
        -> 60
```

```
140:    *  M11  ;  comment routine addresses
        *  J60≠Z
        *  I11
        *  = M5
        *  MOM5
        *  = + M5
        *  M5 TO Q11
        => 60


170:    *  MOM8  ;  comment checksum
        *  VR
        *  MOM7N
        *  +
        *  J171 NV
        *  NOT
        *  NEG
171:    *  -
        *  J199 ≠ Z
        *  M11 TO Q12 ; comment ENTRY
        *  I11
        *  = M5
        *  MOM5N
        *  = +M12
        *  I12 = +1
        *MOM14N
        *SET 4
        *=M14
        *=MOM14
        *  DC11
        comment time limit is in NS
        *  SHL 24
        *  SET 2
        *  OUT  ; comment to compiled program
```

```
199:    *  MOM14
        *  =MOMO
        *  POAQO
        stop


10:     *  SET 100  ; comment clear 100 spaces
        *  = C6
11:     *  ZERO
        *  = MOM6Q
        *  J11 C6 NZ
        *  EXIT 1

30:     *  M8M9Q  ;  comment unpack 2 words
        *  = Q2
        *  M8M9Q
        *  = Q3
        *  M3
        *  I3
        *  C3
        *  M2
        *  I2
        *  C2
        *  SET 3
        *  = C3
        *  EXIT 1


40:     *  DUP  ;  comment load Q10
        *  SET B 17777
        *  AND
        *  = M10
        *  SHL -13
        *  SET 7
        *  AND
        *  = C10
        *  EXIT 1
```

```
50:     *  SHL 24  ;  comment load pattern as Q10
        *  ZERO
        *  REV
        *  J51 C10 Z
52:     *  SHLD-8
        *  DC10
        *  J52 C10 NZ
51:     *  M6
        *  M10
        *  NOT
        *  +
        *  J53 > Z
        *  JS10
        -> 51
53:     *  MOM10
        *  OR
        *  = MOM10

        *  MOM10N
        *  OR
        *  = MOM10N
        *  EXIT 1
        *

60:     *  M8M9Q  ;  comment dump stack /RA
        *  = MOM5Q
        *
        *  J60 C9 NZS
        ->1
end

*


1000: 58P:  end of perm
```