```
( EG00.ASM - one can't avoid this one! )

"Hello world"

HALT

(-------------------------------------------------------)

( EG01.ASM - example showing reverse polish expression evaluation )

40  5  +  3  *  .  ( should print 135 )

HALT

(-------------------------------------------------------)

( EG02.ASM - will divide by zero and crash )

VARIABLE A VARIABLE B VARIABLE C

C  A  @  B  @  /  !  ( c = a / b )

HALT

(-------------------------------------------------------)

( EG03.ASM - store and retrieve a variable )

VARIABLE MeaningOfLife

MeaningOfLife  42  !

"The Meaning Of Life is "

MeaningOfLife @ .

HALT

(-------------------------------------------------------)

( EG04.ASM - some variables and constants )

VARIABLE x
CONSTANT pi 22

pi .          ( should print 22 )

x pi !        ( x should now be 22 )

x @ x @ * .   ( should display 484 )

HALT

(-------------------------------------------------------)

( EG05.ASM - simple tests of logical operations )

true false and .B   ( should display false )

true true  and .B   ( should display true )

false true or  .B   ( should display true )

false not      .B   ( should display true )

HALT

(-------------------------------------------------------)

( EG06.ASM - simple tests of DUP and ABS words )
( note that we can write really silly code to test these
  operations; we do not need meaningful programs! )

3 dup * .      ( should display 9 )
cr             ( new line )

-56 abs .      ( should display 56 )
cr             ( new line )

HALT

(-------------------------------------------------------)

```

```
( EG07.ASM - simple tests of character operations )

VARIABLE ch

ch ?c               ( read ch )
ch @ .              ( should display ascii value for character just read )
cr                  ( new line )
ch @       .c       ( should echo character read )

"\nLess than Z? "
ch @ 'Z' < .b       ( will display true if character read was less than 'Z' )

"\nLetter/digit? "
ch @ ISLD  .b       ( will display true if it was a letter or digit )

"\nUppercase equivalent "
ch @ CAP   .c       ( should display upper case equivalent of character read )

HALT

(-----------------------------------------------------------)

( EG08.ASM - simple tests of ++ and -- )

VARIABLE A

A 12 ! A ++  A @ .    ( should display 13 )

A 25 ! A --  A @ .    ( should display 24 )

HALT

(-----------------------------------------------------------)

( EG09.ASM - test some escape sequences )

(   1       2       3       4
    Hello
    dear
    world
)

"1\t2\t3\t4\nHello\ndear\nworld"

HALT

(-----------------------------------------------------------)

( EG10.ASM - test some array operations )

VARIABLE list VARIABLE index

list  3 new !   ( Allocate space for small array )

"supply subscript 0 - 2 or larger to break it "
index ?            ( read index )

list @ index @ [] 45 !   ( list[index] = 45 )
list @ index @ [] ++     ( list[index]++ )
list @ index @ [] @  .   ( should display 46 )

HALT

(-----------------------------------------------------------)

( EG11.ASM - simple IF statement - deeply significant, no doubt )

VARIABLE age
age ?                                  ( read age )

age @ 60 > IF                          ( if ( age > 60 ) )
  "Is the meaning of life really " 42 .  (   write("Is the meaning of life really ", 42) )
ENDIF

HALT

(-----------------------------------------------------------)

( EG12.ASM - a silly sequence for testing the if - endif construct )

true
if
   "You should see this message first and nothing more"
endif

false
if
   "If you see this there is a problem in your compiler"
endif

HALT

(-----------------------------------------------------------)

```

```
178   ( EG13.ASM - Form the sum of a list of numbers terminated with zero )
179
180   VARIABLE x VARIABLE Total
181
182   Total 0 !               ( Total = 0 )
183
184   REPEAT
185     x ?                   ( Read(x) )
186     Total x @ Total @ + ! ( Total = x + Total )
187     x @ 0 ==              ( x == 0 ? )
188   UNTIL
189
190   "Total is"  Total @ .   ( write ("Total is" , Total) )
191
192   HALT
193
194   (-------------------------------------------------------)
195
196   ( EG14.ASM - simple repeat loop )
197
198   VARIABLE x
199
200   repeat         ( read list and reflect absolute values )
201     x ?
202     x @ abs .
203     cr
204     x @
205     0 ==
206   until
207
208   HALT
209
210   (-------------------------------------------------------)
211
212   ( EG15.ASM - the STK word is a useful one for inspecting the stack )
213
214   1 2 3 4 5 6 7 8 9 10 STK  ( 10 elements )
215   + + + STK                 ( 7 elements - top value is now 10+9+8+7 = 34 )
216
217   HALT
218
219   (-------------------------------------------------------)
220
221   ( EG16.ASM - an upper case conversion without CAP )
222
223   VARIABLE ch
224
225   repeat              ( repeat                    )
226     ch ?c             (   read(ch);               )
227     ch @
228     dup 'a' >=
229     if dup 'z' <=
230       if 32 - endif
231     endif
232     .c                (   display CAP(ch) )
233     ch @ '.' ==       ( until (ch == '.') ;        )
234   until
235
236   HALT
237
238   (-------------------------------------------------------)
239
240   ( EG17.ASM - truth tables and nested loops )
241
242   VARIABLE x VARIABLE y
243
244   "   x      y    x & y  x | y" cr cr
245
246   x false ! y false !
247
248   repeat
249     repeat
250       x @ .b
251       y @ .b
252       x @ y @ and .b
253       x @ y @ or .b
254       cr
255     y y @ not ! y @ not
256     until
257     x x @ not ! x @ not
258   until
259
260   HALT
261
262   (-------------------------------------------------------)
263
```

```
File: eg00.pvm

ASSEM
BEGIN
  {    0 } DSP      0
  {    2 } PRNS     "Hello world"
  {    4 } HALT
END.

 ------------------------------

File: eg01.pvm

ASSEM
BEGIN
  {    0 } DSP      0
  {    2 } LDC      40
  {    4 } LDC      5
  {    6 } ADD
  {    7 } LDC      3
  {    9 } MUL
  {   10 } PRNI
  {   11 } HALT
END.

 ------------------------------

File: eg02.pvm

ASSEM
BEGIN
  {    0 } DSP      3
  {    2 } LDA      2
  {    4 } LDA      0
  {    6 } LDV
  {    7 } LDA      1
  {    9 } LDV
  {   10 } DIV
  {   11 } STO
  {   12 } HALT
END.

 ------------------------------

File: eg03.pvm

ASSEM
BEGIN
  {    0 } DSP      1
  {    2 } LDA      0
  {    4 } LDC      42
  {    6 } STO
  {    7 } PRNS     "The Meaning Of Life is "
  {    9 } LDA      0
  {   11 } LDV
  {   12 } PRNI
  {   13 } HALT
END.

 ------------------------------

File: eg04.pvm

ASSEM
BEGIN
  {    0 } DSP      1
  {    2 } LDC      22
  {    4 } PRNI
  {    5 } LDA      0
  {    7 } LDC      22
  {    9 } STO
  {   10 } LDA      0
  {   12 } LDV
  {   13 } LDA      0
  {   15 } LDV
  {   16 } MUL
  {   17 } PRNI
  {   18 } HALT
END.

 ------------------------------
```

File: eg05.pvm

```
ASSEM
BEGIN
  {    0 } DSP      0
  {    2 } LDC      1
  {    4 } LDC      0
  {    6 } AND
  {    7 } PRNB
  {    8 } LDC      1
  {   10 } LDC      1
  {   12 } AND
  {   13 } PRNB
  {   14 } LDC      0
  {   16 } LDC      1
  {   18 } OR
  {   19 } PRNB
  {   20 } LDC      0
  {   22 } NOT
  {   23 } PRNB
  {   24 } HALT
END.
```

File: eg06.pvm

```
ASSEM
BEGIN
  {    0 } DSP      0
  {    2 } LDC      3
  {    4 } DUP
  {    5 } MUL
  {    6 } PRNI
  {    7 } PRNL
  {    8 } LDC      -56
  {   10 } ABS
  {   11 } PRNI
  {   12 } PRNL
  {   13 } HALT
END.
```

File: eg07.pvm

```
ASSEM
BEGIN
  {    0 } DSP      1
  {    2 } LDA      0
  {    4 } INPC
  {    5 } LDA      0
  {    7 } LDV
  {    8 } PRNI
  {    9 } PRNL
  {   10 } LDA      0
  {   12 } LDV
  {   13 } PRNC
  {   14 } PRNS     "\nLess than Z? "
  {   16 } LDA      0
  {   18 } LDV
  {   19 } LDC      90
  {   21 } CLT
  {   22 } PRNB
  {   23 } PRNS     "\nLetter/digit? "
  {   25 } LDA      0
  {   27 } LDV
  {   28 } ISLD
  {   29 } PRNB
  {   30 } PRNS     "\nUppercase equivalent "
  {   32 } LDA      0
  {   34 } LDV
  {   35 } CAP
  {   36 } PRNC
  {   37 } HALT
END.
```

File: eg08.pvm

```
ASSEM
BEGIN
  {    0 } DSP      1
  {    2 } LDA      0
  {    4 } LDC      12
  {    6 } STO
  {    7 } LDA      0
  {    9 } INC
  {   10 } LDA      0
  {   12 } LDV
  {   13 } PRNI
  {   14 } LDA      0
  {   16 } LDC      25
  {   18 } STO
  {   19 } LDA      0
  {   21 } DEC
  {   22 } LDA      0
  {   24 } LDV
  {   25 } PRNI
  {   26 } HALT
END.
```

File: eg09.pvm

```
ASSEM
BEGIN
  {    0 } DSP      0
  {    2 } PRNS     "1\t2\t3\t4\nHello\ndear\nworld"
  {    4 } HALT
END.
```

File: eg10.pvm

```
ASSEM
BEGIN
  {    0 } DSP      2
  {    2 } LDA      0
  {    4 } LDC      3
  {    6 } ANEW
  {    7 } STO
  {    8 } PRNS     "supply subscript 0 - 2 or larger to break it "
  {   10 } LDA      1
  {   12 } INPI
  {   13 } LDA      0
  {   15 } LDV
  {   16 } LDA      1
  {   18 } LDV
  {   19 } LDXA
  {   20 } LDC      45
  {   22 } STO
  {   23 } LDA      0
  {   25 } LDV
  {   26 } LDA      1
  {   28 } LDV
  {   29 } LDXA
  {   30 } INC
  {   31 } LDA      0
  {   33 } LDV
  {   34 } LDA      1
  {   36 } LDV
  {   37 } LDXA
  {   38 } LDV
  {   39 } PRNI
  {   40 } HALT
END.
```

File: eg11.pvm

```
ASSEM
BEGIN
  {    0 } DSP      1
  {    2 } LDA      0
  {    4 } INPI
  {    5 } LDA      0
  {    7 } LDV
  {    8 } LDC      60
  {   10 } CGT
  {   11 } BZE      18
  {   13 } PRNS     "Is the meaning of life really "
  {   15 } LDC      42
  {   17 } PRNI
  {   18 } HALT
END.
```

File: eg12.pvm

```
ASSEM
BEGIN
  {    0 } DSP      0
  {    2 } LDC      1
  {    4 } BZE      8
  {    6 } PRNS     "You should see this message first and nothing more"
  {    8 } LDC      0
  {   10 } BZE      14
  {   12 } PRNS     "If you see this there is a problem in your compiler"
  {   14 } HALT
END.
```

File: eg13.pvm

```
ASSEM
BEGIN
  {    0 } DSP      2
  {    2 } LDA      1
  {    4 } LDC      0
  {    6 } STO
  {    7 } LDA      0
  {    9 } INPI
  {   10 } LDA      1
  {   12 } LDA      0
  {   14 } LDV
  {   15 } LDA      1
  {   17 } LDV
  {   18 } ADD
  {   19 } STO
  {   20 } LDA      0
  {   22 } LDV
  {   23 } LDC      0
  {   25 } CEQ
  {   26 } BZE      7
  {   28 } PRNS     "Total is"
  {   30 } LDA      1
  {   32 } LDV
  {   33 } PRNI
  {   34 } HALT
END.
```

File: eg14.pvm

```
ASSEM
BEGIN
  {    0 } DSP      1
  {    2 } LDA      0
  {    4 } INPI
  {    5 } LDA      0
  {    7 } LDV
  {    8 } ABS
  {    9 } PRNI
  {   10 } PRNL
  {   11 } LDA      0
  {   13 } LDV
  {   14 } LDC      0
  {   16 } CEQ
  {   17 } BZE      2
  {   19 } HALT
END.
```

File: eg15.pvm

```
ASSEM
BEGIN
  {    0 } DSP      0
  {    2 } LDC      1
  {    4 } LDC      2
  {    6 } LDC      3
  {    8 } LDC      4
  {   10 } LDC      5
  {   12 } LDC      6
  {   14 } LDC      7
  {   16 } LDC      8
  {   18 } LDC      9
  {   20 } LDC      10
  {   22 } STK
  {   23 } ADD
  {   24 } ADD
  {   25 } ADD
  {   26 } STK
  {   27 } HALT
END.
```

File: eg16.pvm

```
ASSEM
BEGIN
  {    0 } DSP      1
  {    2 } LDA      0
  {    4 } INPC
  {    5 } LDA      0
  {    7 } LDV
  {    8 } DUP
  {    9 } LDC      97
  {   11 } CGE
  {   12 } BZE      23
  {   14 } DUP
  {   15 } LDC      122
  {   17 } CLE
  {   18 } BZE      23
  {   20 } LDC      32
  {   22 } SUB
  {   23 } PRNC
  {   24 } LDA      0
  {   26 } LDV
  {   27 } LDC      46
  {   29 } CEQ
  {   30 } BZE      2
  {   32 } HALT
END.
```

File: eg17.pvm

```
ASSEM
BEGIN
  {    0 } DSP      2
  {    2 } PRNS     "   x      y    x & y  x | y"
  {    4 } PRNL
  {    5 } PRNL
  {    6 } LDA      0
  {    8 } LDC      0
  {   10 } STO
  {   11 } LDA      1
  {   13 } LDC      0
  {   15 } STO
  {   16 } LDA      0
  {   18 } LDV
  {   19 } PRNB
  {   20 } LDA      1
  {   22 } LDV
  {   23 } PRNB
  {   24 } LDA      0
  {   26 } LDV
  {   27 } LDA      1
  {   29 } LDV
  {   30 } AND
  {   31 } PRNB
  {   32 } LDA      0
  {   34 } LDV
  {   35 } LDA      1
  {   37 } LDV
  {   38 } OR
  {   39 } PRNB
  {   40 } PRNL
  {   41 } LDA      1
  {   43 } LDA      1
  {   45 } LDV
  {   46 } NOT
  {   47 } STO
  {   48 } LDA      1
  {   50 } LDV
  {   51 } NOT
  {   52 } BZE      16
  {   54 } LDA      0
  {   56 } LDA      0
  {   58 } LDV
  {   59 } NOT
  {   60 } STO
  {   61 } LDA      0
  {   63 } LDV
  {   64 } NOT
  {   65 } BZE      16
  {   67 } HALT
END.
```

```
( EG101.asm - simple tests of swap, over and drop words )
( note that we can write really silly code to test these
  operations; we do not need meaningful programs! )

1 2 swap . .   ( should display 1 2 )
cr             ( new line )

stk            ( should be empty )

4 5 drop .     ( should display 4 )
cr             ( new line )

stk            ( should be empty )

234 126 -45 130
stk            ( four elements )
over
stk            ( five elements )

drop drop drop drop drop  ( pop them all off )

stk            ( should be empty )
cr             ( new line )

HALT

(-------------------------------------------------------)

( EG102.ASM - mystery code )

VARIABLE x VARIABLE y

x ? y ?

OVER OVER > IF SWAP ENDIF DROP .

HALT

(-------------------------------------------------------)

( EG103.ASM - simple test of IF - ELSE - ENDIF )

VARIABLE  x
CONSTANT age 63

"Guess my age! "

x ? x @ age ==

IF "correct" ELSE "incorrect" ENDIF

HALT

(-------------------------------------------------------)

( EG104.ASM - boolean check a sequence of characters for being digits )

VARIABLE ch

repeat
  ch ?c
  ch @
  dup '0' <
  if
    drop false
  else
    '9' > if false else true endif
  endif
  .b cr
  ch @ '.' ==
until

HALT

(-------------------------------------------------------)

( EG105.ASM - new word 0== )

VARIABLE x

repeat
  x ?         ( read x )
  x @ 0==
  if "zero" else "non-zero" endif
  x @ 0==
until

HALT

(-------------------------------------------------------)

```

```
( EG106.ASM - bad constructs - can you detect the errors? )

repeat
  "in loop"
   endif
repeat
  "in second loop"
  true
until

HALT

(-------------------------------------------------------)

( EG107.ASM - simple colon definition )

VARIABLE x

: =0  0 == ;  ( test for zero )

: @.  @ . ;   ( dereference and print )

repeat
  "\nPlease supply a number "
  x ?         ( read x )
  x dup @ =0
  if @. " is zero " else @. " is non-zero " endif
  x @ =0
until

HALT

(-------------------------------------------------------)

( EG108.ASM - colon definitions )

: <0    0 <  ;    ( test negative )

: @=0   @ 0 == ;  ( deref and test zero )

: abs             ( one way of doing it! )
  dup
  <0
  if negate endif
;

VARIABLE x

repeat            ( repeat )
  x ?               ( read(x) )
  x @ abs .         ( write(abs(x)) )
  cr
  x @=0           ( until (x == 0) )
until

HALT

(-------------------------------------------------------)

( EG109.ASM - colon definitions - an upper case conversion )

: toUpper           ( if ('a' <= tos <= 'z') tos -= 32 )
  dup 'a' >=
  if dup 'z' <=
    if 32 - endif
  endif
;

VARIABLE ch

repeat              ( repeat                    )
  ch ?c             (   read(ch);               )
  ch @ toUpper .c   (   write(toUpper(ch));     )

  ch @ '.' ==       ( until (ch == '.') ;       )
until

HALT

(-------------------------------------------------------)

( ED110.ASM - colon definitions and showing reverse polish expression evaluation )

: PEEK DUP . DUP .c ;

65  PEEK 72  PEEK  3  *  + .  ( should print 281 )

HALT

(-------------------------------------------------------)
```

```
( EG111.ASM - colon definitions - count number of positive and negative values )

VARIABLE x
VARIABLE pos
VARIABLE neg

: EVALUATE @ 0 >= IF "positive" pos ++ ELSE "negative" neg ++ ENDIF cr ;

: read(x) "\nSupply next number " x ? ;

: =0 0 ! ;

: display @ . ;

: zero? @ 0 == ;

pos =0
neg =0
repeat
  read(x) x dup evaluate
  zero?
until
pos display " non-negative numbers" cr
neg display " negative numbers" cr

HALT

(-------------------------------------------------------)

```

```
File: EG101.pvm

ASSEM
BEGIN
  {    0 } DSP      0
  {    2 } LDC      1
  {    4 } LDC      2
  {    6 } SWAP
  {    7 } PRNI
  {    8 } PRNI
  {    9 } PRNL
  {   10 } STK
  {   11 } LDC      4
  {   13 } LDC      5
  {   15 } DROP
  {   16 } PRNI
  {   17 } PRNL
  {   18 } STK
  {   19 } LDC      234
  {   21 } LDC      126
  {   23 } LDC      -45
  {   25 } LDC      130
  {   27 } STK
  {   28 } OVER
  {   29 } STK
  {   30 } DROP
  {   31 } DROP
  {   32 } DROP
  {   33 } DROP
  {   34 } DROP
  {   35 } STK
  {   36 } PRNL
  {   37 } HALT
END.

 ------------------------------

File: EG102.pvm

ASSEM
BEGIN
  {    0 } DSP      2
  {    2 } LDA      0
  {    4 } INPI
  {    5 } LDA      1
  {    7 } INPI
  {    8 } OVER
  {    9 } OVER
  {   10 } CGT
  {   11 } BZE      14
  {   13 } SWAP
  {   14 } DROP
  {   15 } PRNI
  {   16 } HALT
END.

 ------------------------------

File: EG103.pvm

ASSEM
BEGIN
  {    0 } DSP      1
  {    2 } PRNS     "Guess my age! "
  {    4 } LDA      0
  {    6 } INPI
  {    7 } LDA      0
  {    9 } LDV
  {   10 } LDC      63
  {   12 } CEQ
  {   13 } BZE      19
  {   15 } PRNS     "correct"
  {   17 } BRN      21
  {   19 } PRNS     "incorrect"
  {   21 } HALT
END.

 ------------------------------
```

```
File: EG104.pvm

ASSEM
BEGIN
  {    0 } DSP      1
  {    2 } LDA      0
  {    4 } INPC
  {    5 } LDA      0
  {    7 } LDV
  {    8 } DUP
  {    9 } LDC      48
  {   11 } CLT
  {   12 } BZE      19
  {   14 } DROP
  {   15 } LDC      0
  {   17 } BRN      30
  {   19 } LDC      57
  {   21 } CGT
  {   22 } BZE      28
  {   24 } LDC      0
  {   26 } BRN      30
  {   28 } LDC      1
  {   30 } PRNB
  {   31 } PRNL
  {   32 } LDA      0
  {   34 } LDV
  {   35 } LDC      46
  {   37 } CEQ
  {   38 } BZE      2
  {   40 } HALT
END.

 ------------------------------

File: EG105.pvm

ASSEM
BEGIN
  {    0 } DSP      1
  {    2 } LDA      0
  {    4 } INPI
  {    5 } LDA      0
  {    7 } LDV
  {    8 } LDC      0
  {   10 } CEQ
  {   11 } BZE      17
  {   13 } PRNS     "zero"
  {   15 } BRN      19
  {   17 } PRNS     "non-zero"
  {   19 } LDA      0
  {   21 } LDV
  {   22 } LDC      0
  {   24 } CEQ
  {   25 } BZE      2
  {   27 } HALT
END.

 ------------------------------

File: EG106.pvm

ASSEM
BEGIN
  {    0 } DSP      0
  {    2 } PRNS     "in loop"
  {    4 } PRNS     "in second loop"
  {    6 } LDC      1
  {    8 } BZE      4
  {   10 } HALT
END.

 ------------------------------
```

File: EG107.pvm

```
ASSEM
BEGIN
  {    0 } DSP      1
  {    2 } PRNS     "\nPlease supply a number "
  {    4 } LDA      0
  {    6 } INPI
  {    7 } LDA      0
  {    9 } DUP
  {   10 } LDV
  {   11 } LDC      0
  {   13 } CEQ
  {   14 } BZE      22
  {   16 } LDV
  {   17 } PRNI
  {   18 } PRNS     " is zero "
  {   20 } BRN      26
  {   22 } LDV
  {   23 } PRNI
  {   24 } PRNS     " is non-zero "
  {   26 } LDA      0
  {   28 } LDV
  {   29 } LDC      0
  {   31 } CEQ
  {   32 } BZE      2
  {   34 } HALT
END.
```

File: EG108.pvm

```
ASSEM
BEGIN
  {    0 } DSP      1
  {    2 } LDA      0
  {    4 } INPI
  {    5 } LDA      0
  {    7 } LDV
  {    8 } DUP
  {    9 } LDC      0
  {   11 } CLT
  {   12 } BZE      15
  {   14 } NEG
  {   15 } PRNI
  {   16 } PRNL
  {   17 } LDA      0
  {   19 } LDV
  {   20 } LDC      0
  {   22 } CEQ
  {   23 } BZE      2
  {   25 } HALT
END.
```

File: EG109.pvm

```
ASSEM
BEGIN
  {    0 } DSP      1
  {    2 } LDA      0
  {    4 } INPC
  {    5 } LDA      0
  {    7 } LDV
  {    8 } DUP
  {    9 } LDC      97
  {   11 } CGE
  {   12 } BZE      23
  {   14 } DUP
  {   15 } LDC      122
  {   17 } CLE
  {   18 } BZE      23
  {   20 } LDC      32
  {   22 } SUB
  {   23 } PRNC
  {   24 } LDA      0
  {   26 } LDV
  {   27 } LDC      46
  {   29 } CEQ
  {   30 } BZE      2
  {   32 } HALT
END.
```

File: EG110.pvm

```
ASSEM
BEGIN
  {    0 } DSP      0
  {    2 } LDC      65
  {    4 } DUP
  {    5 } PRNI
  {    6 } DUP
  {    7 } PRNC
  {    8 } LDC      72
  {   10 } DUP
  {   11 } PRNI
  {   12 } DUP
  {   13 } PRNC
  {   14 } LDC      3
  {   16 } MUL
  {   17 } ADD
  {   18 } PRNI
  {   19 } HALT
END.
```

File: EG111.pvm

```
ASSEM
BEGIN
  {    0 } DSP      3
  {    2 } LDA      1
  {    4 } LDC      0
  {    6 } STO
  {    7 } LDA      2
  {    9 } LDC      0
  {   11 } STO
  {   12 } PRNS     "\nSupply next number "
  {   14 } LDA      0
  {   16 } INPI
  {   17 } LDA      0
  {   19 } DUP
  {   20 } LDV
  {   21 } LDC      0
  {   23 } CGE
  {   24 } BZE      33
  {   26 } PRNS     "positive"
  {   28 } LDA      1
  {   30 } INC
  {   31 } BRN      38
  {   33 } PRNS     "negative"
  {   35 } LDA      2
  {   37 } INC
  {   38 } PRNL
  {   39 } LDV
  {   40 } LDC      0
  {   42 } CEQ
  {   43 } BZE      12
  {   45 } LDA      1
  {   47 } LDV
  {   48 } PRNI
  {   49 } PRNS     " non-negative numbers"
  {   51 } PRNL
  {   52 } LDA      2
  {   54 } LDV
  {   55 } PRNI
  {   56 } PRNS     " negative numbers"
  {   58 } PRNL
  {   59 } HALT
END.
```
