

The algorithm is from Section 1.3.2 of *The Art of Computer Programming, Vol. 1.*

```

; -*- MIDAS -*- Print first 500 primes
        TITLE   PRIME
L==500. ; Decimal 500
J=1
N=2
K=3
A=4
R=5
PRIME:  BLOCK   L
BUF:    BLOCK   30
PDL:    BLOCK   10
START:  MOVE    J,[-L-1,,PRIME-1]      ; setup J
        MOVEI   N,2
        PUSH   J,N
        MOVEI   N,3      ; N <- 3
P2:     PUSH   J,N
        HLRO   A,J
        AOJE   A,PRIN  ; found
P4:     ADDI   N,2
        MOVEI   K,1
P6:     MOVE   A,N
        IDIV   A,PRIME(K)
        JUMPE  R,P4      ; Goto P4 if remainder is zero
        CAMG   A,PRIME(K) ; if Q <= P[K]
        JRST   P2      ; Goto P2
        ADDI   K,1
        JRST   P6
PRIN:   .OPEN   1,[.UAD,,TTY] ; The rest are just about
        .LOSE   %LSSYS      ; putting numbers in PRIME[
        MOVE   J,[-10,,PDL-1] ; on screen
        MOVE   A,[440700,,TITL]
        PUSHJ  J,OUT
        MOVEI  K,0      ; Setup line
CONV:   MOVE   N,[440700,,BUF]
NUM:    MOVE   A,PRIME(K)
        IDIVI  A,1000.
        PUSHJ  J,DPC
        IDIVI  A,100.
        PUSHJ  J,DPC
        IDIVI  A,10.
        PUSHJ  J,DPC
        CAIL  K,450.
        JRST  PUTL
        MOVEI  A,40
        IDPB  A,N
        ADDI  K,50.
        JRST  NUM
PUTL:   MOVE   A,[440700,,BUF]
        PUSHJ  J,OUT
        ADDI  K,1
        CAIL  K,L
        .LOGOUT 1,
        SUBI  K,450.
        JRST  CONV

DPC:    ADDI  A,60
        IDPB  A,N
        MOVE  A,R
        POPJ  J,
OUT:    ILDB  N,A
        JUMPE N,EXIT
        .IOT  1,N
        JRST  OUT
EXIT:   .IOT  1,[^M]
        .IOT  1,[^J]
        POPJ  J,
TITL:   ASCIZ  $FIRST FIVE HUNDRED PRIMES$
END     START

```