

```

; Computation program example
; Eliav Gnessin, Fall 2002
; =====
; This is an example program in 8086 Assembly
; The user enters a 4 digits positive number and
; we will print the result of (3*Number+10)
; =====

TITLE COMPUTATION

; This instruction defines the memory model that MASM or TASM use
.model small

; Define the stack size. This instruction initializes the SP
.stack 160h

; Variables & other definitions section
.data
infor      db "Enter number",10,13      ; infor is the string to be printed
inforlen   equ 14                       ; this is the string's length
arg        dw 10                         ; our computation base = 10
number     dw 4 dup(?)                  ; array of 4 words
zeroh      dw 0
three      dw 3

; This is the program itself
.code
start:     mov ax,@data                  ; Since the .data instruction doesn't initialize
          mov ds,ax                    ; the ds register we have to do it manually

          mov cx,inforlen               ; This is the string's length for our loop
          mov si,0                      ; This will be the index of the string pointer
again:     mov al,infor[si]             ; Get char from loop
          call printch                  ; Print it
          inc si                         ; Increment our index
          loop again                    ; Do it for all the chars in the string

          ; Get digits from keyboard and store in array
          mov cx,4                      ; We will always get 4 chars from the user
          mov si,0                      ; This is the index of the GETCH loop
again2:    call getch
          call printch
          sub al,30h                    ; We got the ASCII code, convert to number
          mov ah,0
          mov number[si],ax             ; number is array of 4 Words
          add si,2                      ; so step two bytes at a time
          loop again2

          ; Convert digits in array to one number
          mov cx,4                      ; number of digits
          mov si,0                      ; our loop inx
          mov ax,0                      ; AX will hold the resulting number
again3:    mul arg                      ; AX=AX*10
          add ax,number[si]             ; AX=AX+[next digit]
          add si,2                      ; next digit in array
          loop again3

          ; Do the computation - AX=AX*3+10
          mul three
          add ax,10

          ; Convert result to string, push to stack

```

```

; si will count the num of digits
again4: mov si,0
        div dx,0
        add dx,30h
        push dx
        inc si
        cmp zeroh,ax
        mov cx,2
        loopnz again4

; Move down to next line - Carriage Return + Line Feed
mov cx,si

mov al,10
call printch
mov al,13
call printch

again5: pop ax
        call printch
        loop again5

mov ax,4c00h
int 21h

; =====
; Procedure definitions
; =====

; =====
; Procedure name: printch - Print a char to console
; Input:         AL - the char's ASCII code
; Output:        None
; =====
printch proc near
        mov bx,0
        mov ah,0Eh
        int 10h
        ret
printch endp

; =====
; Procedure name: getch - Get a char from console
; Input:         None
; Output:        AL - the char's ASCII code
; =====
getch proc near
        mov ah,0
        int 16h
        ret
getch endp

; End of program
end start

```