



CENTRO DE CIENCIAS BÁSICAS
DEPARTAMENTO DE SISTEMAS ELECTRÓNICOS
LENGUAJE ENSAMBLADOR
7° "A"

PRÁCTICA 6

Profesor: Cristian Jael Mejía Aguirre

Alumno: Joel Alejandro Espinoza Sánchez

Fecha de Entrega: Aguascalientes, Ags., **20** de octubre de 2021

Práctica 6

Objetivo: Realizar un programa en ASM que utilice al menos 6 directivas diferentes.

Desarrollo: Se buscó organizar las directivas realizadas dentro del código bajo las investigaciones con el siguiente código:

```
%include "io64.inc"

cpu x86-64 ;Elementos únicos de la arquitectura

section .bss ;Declarar variables no inicializadas
var resd 4 ;Uso de 4 bytes

section .data ;Declarar variables inicializadas
cons equ 3 ;Constante
var2 dw 128 ;Variable
cadena db `á é í ó ú` ;Cadena UTF
vector db 18,12,30 ;Vector de 3 valores (byte)
vector2 times 5 dw 0 ;Vector de 5 palabras como 0
var3 dw 10

section .text
global CMAIN
CMAIN:
    mov rbp, rsp
    ;Uso de var
    GET_STRING var, 11
    PRINT_STRING var

    ;Uso de var2
    mov eax, [var2]
    NEWLINE
    PRINT_DEC 1, eax

    ;Uso de cons y cadena
    add ebx, cons
    NEWLINE
    PRINT_DEC 1, ebx
    PRINT_STRING cadena

    ;Uso de vector
    NEWLINE
    mov ecx, vector
    PRINT_DEC 1, ecx
```

```

;Uso de var3
mov ax, [var3]
NEWLINE
control:
PRINT_DEC 1, ax
NEWLINE
dec ax
jnz control
ret

```

Sólo basta otorgar una entrada de cadena como a continuación:

Entrada

Prueba

El programa anterior daba la siguiente salida con la entrada anterior:

Salida

Prueba
128
4Ãi Ã© Ã ¢ Ã³ Ã°
16416
10
9
8
7
6
5
4
3
2
1

Dentro del programa SASM, el programa se veía de la siguiente forma:

```
1  %include "io64.inc"
2  |
3  cpu x86-64 ;Elementos únicos de la arquitectura
4
5  section .bss ;Declarar variables no inicializadas
6  var resd 4 ;Uso de 4 bytes
7
8  section .data ;Declarar variables inicializadas
9  cons equ 3 ;Constante
10 var2 dw 128 ;Variable
11 cadena db `á é í ó ú` ;Cadena UTF
12 vector db 18,12,30 ;Vector de 3 valores (byte)
13 vector2 times 5 dw 0 ;Vector de 5 palabras como 0
14 var3 dw 10
15
16 section .text
17 global CMAIN
18 CMAIN:
19     mov rbp, rsp
20     ;Uso de var
21     GET_STRING var, 11
22     PRINT_STRING var
23
24     ;Uso de var2
25     mov eax, [var2]
26     NEWLINE
27     PRINT_DEC 1, eax
28
29     ;Uso de cons y cadena
30     add ebx, cons
31     NEWLINE
32     PRINT_DEC 1, ebx
33     PRINT_STRING cadena
34
35     ;Uso de vector
36     NEWLINE
37     mov ecx, vector
38     PRINT_DEC 1, ecx
39
40     ;Uso de var3
41     mov ax, [var3]
42     NEWLINE
43     control:
44     PRINT_DEC 1, ax
45     NEWLINE
46     dec ax
47     jnz control
48     ret
```

Conclusión: El uso de directivas para esta práctica nos permite ver cierta flexibilidad del lenguaje ensamblador, junto con algún uso personal, nos da creatividad para realizar una práctica donde hemos aprendido una buena organización de código.