

INSTITUTO TECNOLÓGICO SUPERIOR DE EL MANTE

LENGUAJES DE INTERFAZ

UNIDAD 4



Docente: JOSE PAULINO RAMIREZ JUAREZ

Equipo 4:

Miguel Ángel González Martínez

André Ibarra Pérez

Adriana Reséndiz Mares

Ernesto Antonio Tinajero Almaraz

Osiel Alejandro Treto Cedillo

Clase: Lunes a Jueves 7:00 – 7:55

INGENIERÍA EN SISTEMAS COMPUTACIONALES

MANUAL DE PRÁCTICA DE: TALLER

Nombre de la práctica	PROGRAMA FINAL
Nombre del alumno	EQUIPO 4

Carrera	Asignatura	Unidad	Tema
INGENIERIA EN SISTEMAS COMPUTACIONALES	LENGUAJE DE INTERFAZ	UNIDAD 4	LENGUAJE ENSAMBLADOR

1- Desarrollo de la práctica

Esta práctica constó en hacer un programa que ingresáramos un número después este número sería pasado por un método de división entre 10, así guardando el cociente y el residuo, el residuo son los dígitos del numero pero traídos del final al principio, entonces una vez identificados todos estos solo era de mostrar, o sea imprimir cada uno de estos dígitos en pantalla, entonces imprimiendo el número que en un principio de ingresó, todo esto de manera dinámica y mostrando correctamente los dígitos, del número que en un principio se ingresaron.

2- Resultados y Conclusiones

En este programa a realizar había tres grandes problemas, el primero era como hacer un ciclo dinámico que diera solo las vueltas de acuerdo al número de dígitos, el segundo era como íbamos a mostrar los números si todo esto se imprime en código ASCII y el tercero era uno que no estaba contemplado y era que al hacer la división y querer imprimir el número, este se imprime al revés o sea si ingresamos un 1024 entonces lo que se imprimía era 4201 y eso no era lo que se buscaba, entonces el hacer el ciclo dinámico se tuvo que hacer con un CMP y con JMP entre etiquetas debido a que el ciclo continuaba siempre y cuando se cumpliera la condición de que el número a dividir fuera mayor que el divisor que en este caso era el 10, si no era así entonces se rompía el ciclo y seguía de otra forma, entonces después se imprimió todo y salían en código ASCII y al revés, entonces se nos ocurrió utilizar pilas, debido a que estas son ultima que entra primera que sale, entonces si le hacíamos pop con todo ahí ingresado entonces se imprimiría el numero bien y efectivamente al hacer push y un pop el número se imprimía correctamente pero, aun salía en ASCII, entonces lo único que se tuvo que hacer, como todos eran números entre el 1 y el 9 entonces se tuvo que sumar a cada uno de esos un 48 al hacer push en la pila y todo esto para qué? Para que cuando se hiciera un pop y se imprimieran estos salieran correctamente con su número, ahora sí, si ingresáramos en el programa un 429, de resultado sale un 429, si ponemos un 43832 efectivamente da 43832, y así con algunos números, no tan grandes porque no soporta tantos, pero sí que se puede, por ende la conclusión es que es un programa un tanto fácil de entender a mano, pero ya programarlo en ensamblador es difícil porque se tienen pocos registros y hay que estar jugando con ellos y con variables temporales que nos ayuden a conseguir los resultados que queremos porque ciertas instrucciones solo se nos permiten con registros.

3- Anexos y Evidencia

```
PROGFINAL.asm x
1  DATOS SEGMENT
2  VALOR DW 5992
3  COCIENTE DW ?
4  RESIDUO DW ?
5  VALORCICLO DW 0
6  DATOS ENDS
7
8  PILA SEGMENT STACK"STACK"
9  DW 32 DUP(0)
10 PILA ENDS
11
12 CODIGO SEGMENT
13 ASSUME DS:DATOS,SS:PILA,CS:CODIGO
14 PRINCIPAL PROC
15 MOV AX,DATOS
16 MOV DS,AX
17
18
19 MOV CX,1
20 CICLO:
21 ADD VALORCICLO,1
22 MOV BX,VALOR
23 CMP BX,10
24 JA ET2
25 JB ET1
26
27 ET1:
28
29 MOV DX,BX
30 ADD DX,48
31 PUSH DX
32
33 JMP ET3
34
```

```

34
35 ET2:
36
37 MOV DX,0
38 MOV AX,VALOR
39 MOV BX,10
40 DIV BX
41
42 ;AX COCIENTE
43 ;DX RESIDUO
44 ;♦♦♦♦
45
46 MOV VALOR,AX
47 ADD DX,48
48 PUSH DX
49
50 INC CX
51
52 LOOP CICLO
53
54
55 ET3:
56 MOV CX,VALORCICLO
57 CICLOO:
58 POP DX
59 MOV AH,02H
60 INT 21H
61 LOOP CICLOO
62
63 MOV AH,4CH
64 INT 21H
65 PRINCIPAL ENDP
66 CODIGO ENDS
67 END PRINCIPAL

```

```

C:\Windows\system32\CMD.exe
C:\masm32\masm32\bin>ml/c PROGFINAL.asm
Microsoft (R) Macro Assembler Version 6.14.8444
Copyright (C) Microsoft Corp 1981-1997. All rights reserved.

Assembling: PROGFINAL.asm
C:\masm32\masm32\bin>link16 PROGFINAL.obj
Microsoft (R) Segmented Executable Linker Version 5.60.339 Dec 5 1994
Copyright (C) Microsoft Corp 1984-1993. All rights reserved.

Run File [PROGFINAL.exe]:
List File [nul.map]:
Libraries [libl]:
Definitions File [nul.def]:

C:\masm32\masm32\bin>PROGFINAL
5992
C:\masm32\masm32\bin>

```