LABORATORIO I

Franz López Arredondo 2420191051@estudiantesunibague.edu.co

- **ENUNCIADO**

Para la elaboración del presente laboratorio se plantearon 4 puntos a desarrollar, a lo largo del documento se mostrarán los resultados obtenidos de dicho taller

- **OBJETIVOS ESPECÍFICOS**

1. Elaborar códigos que respondan correctamente a lo esperado del laboratorio.
2. Construir mediante el lenguaje ensamblador el código más óptimo posible.

- **DESARROLLO**

El primer objetivo del laboratorio es realizar un programa que imprima en pantalla un cuadrado de 10x10 con números del 0 al 9 usando un ciclo loop, ahora bien, para la elaboración de este código y asegurar el funcionamiento de el "Loop" se debió asegurar que el ciclo tuviese un límite de repeticiones, esto se aseguro dándole valores al "CX"

```
030  A30:
031                MOV     CTR,30H
032                MOV     CX,VCX
033                DEC     VCX
034                LOOP    A20
035                CALL    F10READ
036                CALL    G10MODE
037                MOV     AX,4C00H
038                INT     21H
039  MAIN          ENDP
```

Imagen.1 Fragmento del primer código

Como se observa en la función A30 se establece el "CX", sin embargo se hace un llamado a "VCX ", esto con el fin de que cuando termine el primer ciclo, el "CX" cuente con su valor original y varíe con el paso del código, sin embargo, en la línea 033 del código se decrementa, esto se hace para que no siempre conserve el número de veces establecido desde el principio, sino que, se reduzca con el fin de llegar a 0 y de esta manera deje de repetir su función.

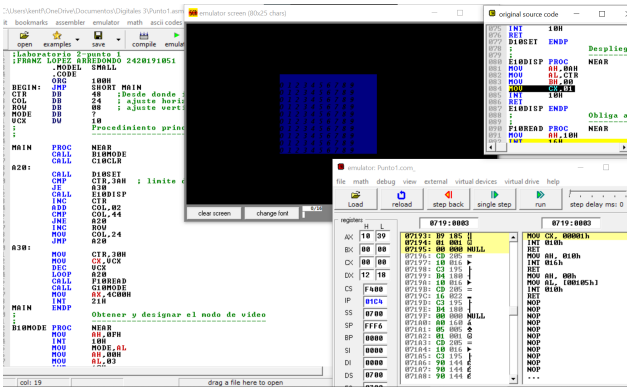

Imagen.2 Comprobación del código

Como se observa en la imagen.2 en la pantalla negra, se repite la serie de números del 0 al 9 en 10 ocasiones, sin embargo, el documento tendrá anexos en los cuales se presentará el código.

Para el segundo punto del taller, se nos pide centrar en pantalla de forma vertical los caracteres ingresados por teclado, haciendo uso de tamaños máximos menores a 15, al elaborar este código se creó una variable donde se guardará lo escrito por el usuario, no sin antes declarar un límite de valores, el cual es el 15, para que los caracteres ingresados no superan dicha cifra.

Ahora, para el ajuste vertical, se realizó un salto de línea y un borrar junto a este, con el fin de eliminar el caracter de espacio que se genera al saltar una línea como se muestra a continuación en la líneas de código 115 y 117.

```
108  ;                      Cambio
109  ;-----------------------------------
110  CAMBIO         PROC    NEAR
111                 MOV     AL,[SI]
112                 MOV     [DI],AL
113                 INC     SI
114                 INC     DI
115                 MOV     [DI],0AH
116                 INC     DI
117                 MOV     [DI],08H
118                 INC     DI
119                 MOV     CL,VARCX
120                 DEC     VARCX
121                 LOOP    CAMBIO
122                 RET
123  CAMBIO         ENDP
```

Imagen.3 Fragmento de código 2

Sin embargo para que el código cumpla la función requerida se realizó el ajuste de la línea 82 y 83, donde se les denomina un valor vertical y otro horizontal a los carácteres que se registran, tal cual se presencia en la imagen 4

```
076 ;                       Centrar y exhibir nombre:
077 ;-----------------------------------------------
078 F10CENT         PROC    NEAR
079                 MOV     DH,NAMELEN          ;Para
080                 SHR     DH,1
081                 NEG     DH
082                 ADD     DH,12               ;Mitad
083                 MOV     DL,40               ;Mitad
084                 CALL    Q20CURS
085                 MOV     AH,09H
086                 LEA     DX,SALIDA
087                 INT     21H
088                 RET
```

Imagen.4, ajuste de caracteres.

Teniendo como resultado lo mostrado en la siguiente imagen.

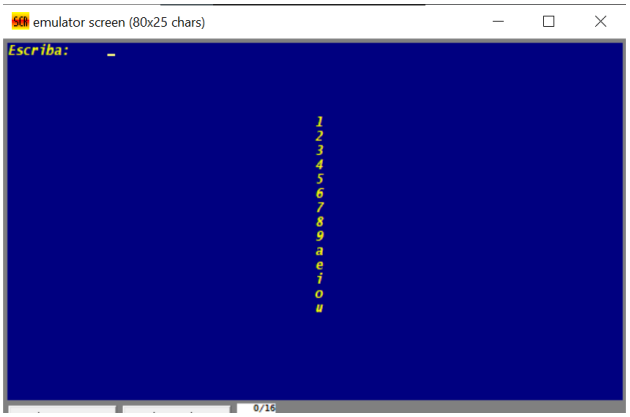

Imagen.5, resultado.

El tercer código es similar al anterior, con la diferencia de que el usuario elige si desea realizar nuevamente la opción o no, para este caso se asignó el número 1 como “Volver a repetir” y el número 2 haciendo referencia a la terminación del programa.

```
147 ;                       Analisis
148 ;-----------------------------------
149 ANALISIS        PROC    NEAR
150                 CMP     NAMEFLD,31H
151                 JE      A20LOOP
152                 CMP     NAMEFLD,32H
153                 JE      A30
154                 JMP     A30
155 ANALISIS        ENDP
156                 END     BEGIN
157
```

Imagen.6, Fragmento codigo3

Observando la Imagen.6 en la línea 150 se compará los vales y busca si el valor registrado es el correcto, el 1 que en hexadecimal es equivalente a 31H, siendo correcto esto, se permite la repetición de las acciones, sin embargo al seleccionar el 2 (en hexadecimal el 32H) el código finalizará, como se muestra en el línea 152, como prueba superficial del resultado se presentan las siguientes 2 imágenes.

Imagen7, resultado.3,1

Imagen8, resultado.3,2

Por último, se asigna realizar un código en el cual el usuario establece los límites, tanto el máximo como el mínimo de los caracteres que se muestran en pantalla, esto en números Ascii, los cuales cuentan con un rango de 0 a 255.
Al realizar este punto fue necesario el transformar los números ingresados a un solo valor Ascii, ya que el programa lee los números como “1,2,3” en lugar de “123” , para esto se realizaron las operaciones de multiplicar por 100 el el primero de los 3 dígitos (si los hay) por 10 el segundo dígito y por 1 el último dígito para después sumarlos entre ellos y obtener el resultado esperado, con esto se establecen los

límites a gusto del usuario, a continuación se presentará los resultados, sin embargo, en los anexos quedarán los códigos para ser comprendidos con mayor información.

Imagen9, resultado.4,1

Como primer paso se pide ingresar el valor mínimo a establecer.

Imagen10, resultado.4,2

Una vez seleccionado el valor mínimo el código despliega una variable para establecer el rango máximo

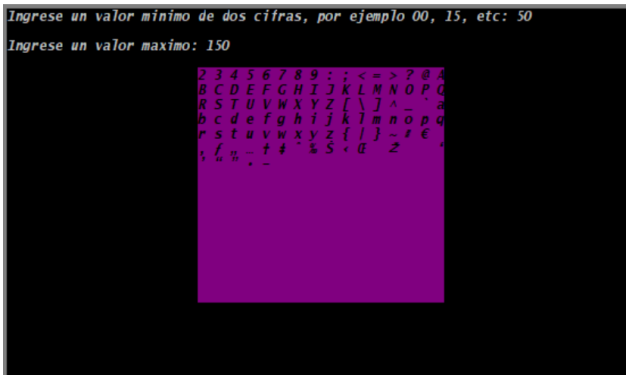

Imagen11, resultado.4,3

Ya establecidos los rangos y oprimiendo “enter” se mostrará en pantalla lo solicitado por el usuario.

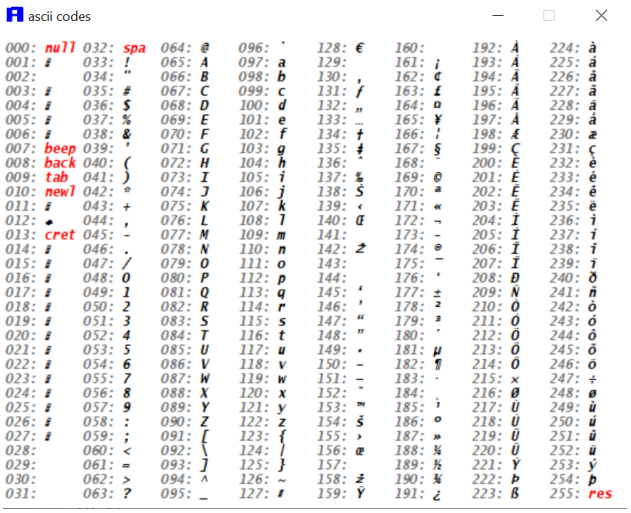

Imagen12, tabla Ascii

Una vez se encuentre todo realizado comprobamos los caracteres desplegados con la tabla Ascii, esto para comprobar su correcto funcionamiento.

- **Anexos**

https://github.com/franz-b11/Lab-2-DigIII