

# Guía de Ejercicios Prácticos

## Arquitectura del Conjunto de Instrucciones

Programar y simular los siguientes ejercicios utilizando el Debug:

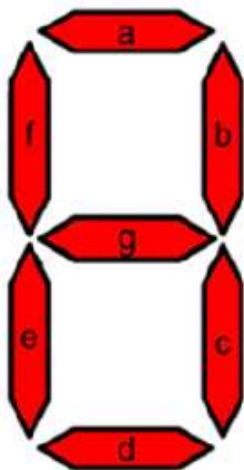
1. Escribir la letra a en la memoria de video.
2. Llenar una fila de la pantalla de la consola con la letra a.
3. Escribir una fila con el abecedario.
4. Escribir todo el abecedario con un atributo diferente para cada letra.
5. Pintar la pantalla de la consola con los colores de tu equipo de futbol favorito.

Programar y simular los siguientes ejercicios utilizando el EMU386:

6. Escribir "Organización del Computador" sin acentos en la memoria de video con letra roja sobre fondo amarillo.
7. Escribir en la pantalla la palabra microprocesador y a continuación la misma palabra invertida de derecha a izquierda.
8. Mostrar en la pantalla la suma de 10 datos.
9. Un byte se encuentra en la dirección 0100. Ubicar en la dirección 0101 su nibble más significativo.
10. En las direcciones 0100 y 0101 hay dos enteros positivos. Ubicar en la dirección 0102 el menor de los dos.
11. A partir de la dirección 0200 hay un número de 3 bytes de longitud. Luego de ese número, en la dirección siguiente de la memoria, hay otro número de 24 bits. Sumar los dos números y guardar el resultado a partir de la dirección posterior al segundo número.

12. Mediante iteraciones rotar los 8 LSB del registro BL hacia el registro BH.
13. Realizar el checksum (suma de chequeo mediante XOR) del conjunto de datos que comienzan en la dirección 202 y tiene una longitud dada por el dato almacenado en la dirección 201. Almacenar el resultado en la dirección 200.
14. Sumar el conjunto de datos de 16 bits que comienzan en la dirección 203 y tiene una longitud dada por el dato almacenado en la dirección 202. Elegir los datos para que su suma sea menor a 65.536. Almacenar el resultado a partir de la dirección 200.
15. A partir de la dirección 204 hay un conjunto de datos de longitud igual al dato almacenado en la dirección 203. Almacenar en la dirección 200 la cantidad de datos que tienen el bit 3 en 1.
16. A partir de la dirección 204 hay un conjunto de datos de longitud igual al dato almacenado en la dirección 203.  
Almacenar en la dirección 200 la cantidad de datos igual a cero, en la 201 la cantidad de datos positivos y en la 202 la cantidad de datos negativos.
17. A partir de la dirección 3002 hay un conjunto de datos de longitud dada en la dirección 3001. Almacenar en la dirección 3000 el dato menor.
18. Almacenar en la dirección 200 la cantidad de unos que tiene el dato almacenado en la dirección 201.
19. A partir de la dirección 201 comienza una cadena de caracteres ASCII. Este mensaje no incluye a los dos caracteres inicial y final ya que son:
  - Caracter de control STX de Inicio de Texto (02h)
  - Caracter de control ETX de Final de Texto (03h)Almacenar en la dirección 200 la cantidad de caracteres del mensaje.

20. El caracter ASCII de control de retorno de carro CR es, en decimal, el 13. Este caracter se encuentra al final de una cadena que comienza en la dirección 202. Ubicar en las posiciones de memoria 200 y 201 la dirección del último caracter antes del CR.
21. Una cadena de caracteres ASCII de longitud dada por el dato almacenado en la dirección 201 se inicia en la dirección 202. Almacenar FF en la dirección 200 si la paridad de todos los caracteres es par. En caso contrario almacenar 00.
22. Dos cadenas de caracteres, A y B, de longitudes iguales e igual al dato almacenado en la dirección 201 comienza en las direcciones 202 y 234 respectivamente. Si la cadena A es mayor o igual a la cadena B según el orden alfabético, almacenar 0 en la dirección 200. En caso contrario almacenar 1.
23. En un display de 7 segmentos, cada segmento se enciende con un 1 y se apaga con un 0 según la siguiente tabla.



num	g	f	e	d	c	b	a	hexa
0	0	1	1	1	1	1	1	3F
1	0	0	0	0	1	1	0	06
2	1	0	1	1	0	1	1	5B
3	1	0	0	1	1	1	1	4F
4	1	1	0	0	1	1	0	66
5	1	1	0	1	1	0	1	6D
6	1	1	1	1	1	0	1	7D
7	0	0	0	0	1	1	1	07
8	1	1	1	1	1	1	1	7F
9	1	1	0	1	1	1	1	6F

Por ejemplo, para el número 2 los segmentos que se encienden son a, b, g, e y d que van en 1 y los demás, f y c van en 0.

Se pide:

- Armar una tabla de búsqueda con los 10 valores hexadecimales a partir de la dirección 2050h.

- b. Traducir la tabla mediante la instrucción XLAT para enviar los 10 números 7 segmentos a un puerto de salida que está en la dirección 20FF.

	DIRECCIÓN	DATO
	204F	
$\xrightarrow{\text{BX}}$	2050	3F
	2051	6
	2052	5B
	2053	4F
$\xrightarrow{\text{BX} + \text{AL}}$	2054	66
	2055	6D
	2056	7D
	2057	07
	2058	7F
	2059	6F

Instruction	Operands	Description
XLAT	No operands	<p>Traducir byte de la tabla. Copia el valor del byte de memoria en DS:[BX + AL sin signo] al registro AL.</p> <p>Algorithm: AL = DS:[BX + unsigned AL]</p> <p>Example:</p> <pre> ORG 100h LEA BX, dat MOV AL, 2 XLAT    ; AL = 33h  RET  dat DB 11h, 22h, 33h, 44h, 55h </pre> <div style="border: 1px solid black; padding: 2px; display: inline-block;"> C Z S O P A  unchanged </div>

El modo de direccionamiento de XLAT es direccionamiento implícito.

El desplazamiento es BX + AL, donde BX = 2050h y AL es el dígito que se desea mostrar. Si inicialmente AL = 08h, BX + AL = 2058h es la dirección dentro de la tabla donde se encuentra la codificación de segmentos para el número 8.

```

MOV BX, 2050h
MOV AL, 08h
XLAT

```

### ***Organización del Computador***

---

Antes de XLAT el valor de AL era 08.  
La dirección a la que apunta es la 2058.  
Luego de XLAT el valor de AL es 7F.