

Guía de ejercicios # 10: Q6: Arreglos

Organización de computadoras

UNQ

Procesamiento de arreglos

1. Completar los espacios en blanco de las oraciones según corresponda:

- (a) Los datos de un arreglo deben ser y
- (b) Un arreglo se delimita por un y una
- (c) La condición de corte puede ser o
- (d) Se llama al dato que determina el fin de un arreglo pero no se debe procesar
- (e) El modo de direccionamiento no es suficiente para recorrer un arreglo
- (f) El modo de direccionamiento se utiliza para direccionar un registro, a diferencia del modo de direccionamiento
- (g) El es una variable que se utiliza para acceder a cada elemento a procesar
- (h) Se utiliza alguno de los modos de direccionamiento como índice

2. Se necesita una rutina que, dado un arreglo de 4 valores en *BSS(16)*, retorne la suma de sus elementos. El arreglo se encuentra almacenado a partir de la celda con dirección *ADDD*.

- (a) Dar un ejemplo de un posible escenario con los datos iniciales y el resultado esperado
- (b) Documentar la rutina
- (c) Definir la rutina que cumpla con la documentación previa

3. Se solicita documentar y definir la rutina *cantElementos* que, a partir de un arreglo de valores en *BSS(16)*, retorne su cantidad de elementos. El arreglo se encuentra a partir de la dirección indicada por *R0*, y finaliza con el primer elemento cuyo valor sea 0.

4. Definir la rutina *copiarArreglo* en función de su documentación:

copiarArreglo	
Requiere	un arreglo de cadenas, almacenado a partir de la celda con dirección <i>8400</i> y que finalice con el primer elemento cuya cadena sea <i>0xFFFF</i>
Retorna	una copia del arreglo original a partir de la celda con dirección <i>9400</i>
Modifica	COMPLETAR

5. Documentar y definir la rutina *aplicarAbsolute* que a partir de un arreglo de valores en *SM(16)*, retorne dicho arreglo pero con sus elementos en valor absoluto. El arreglo se encuentra almacenado a partir de la celda con dirección *0x4486*, y su longitud se encuentra en la celda con dirección *0x4485*.

Nota: usar la rutina absolute de la práctica anterior.

6. ¿Qué se debe modificar en la rutina *aplicarAbsolute* si el parámetro de la dirección inicial del arreglo se recibe en *R0*?

7. En una fábrica de ventanas se codifican los pedidos en cadenas de 16 bits, y para el caso de que la ventana tenga vidrio de seguridad o esmerilado, es necesario usar un embalaje distinto, por lo que se considera que el pedido es Premium. Se solicita documentar y definir la rutina *esPedidoPremium* que determine si el pedido es premium o no, indicando en *R7* un 1 en caso afirmativo.

Para esto, tener en cuenta que el pedido se encuentra en *R5*, y cuenta con el siguiente formato:

RRRRRRR(7b)	S(1b)	CCCCC(5b)	E(1b)	PP(2b)
-------------	-------	-----------	-------	--------

Donde: *R* se utiliza como relleno, *S* para indicar si el vidrio es de seguridad o no, *C* para el código del pedido, *E* para indicar si el vidrio es esmerilado o no, y *P* para el precio.

8. A partir de la documentación dada, definir la rutina *cantPedidosBasicos*, y al finalizar completar las variables que se modifican.

cantPedidosBasicos	
Requiere	un arreglo de pedidos a partir de la celda con dirección <i>9ED1</i> , y cuya longitud está en en la celda con dirección <i>9ED0</i>
Retorna	en <i>R6</i> la cantidad de pedidos que son básicos, es decir que no son premium
Modifica	COMPLETAR

9. Definir una rutina de test para probar el funcionamiento de la rutina *cantPedidosBasicos*.

Bajo nivel: arquitectura

10. Considerando el siguiente mapa de memoria y el valor de los registros:

	...
0000	0001
0001	0004
0002	0000
0003	0002
0004	0001
	...

Registro	Valor almacenado
R0	0x0001
R1	0x0003

Indicar el **efecto** para cada instrucción:

- MOV [R0], [[0x0003]]
 - MOV [[0x0003]], [R0]
 - MOV [[0x0003]], [[0x0001]]
 - MOV [[0x0001]], [R1]
11. Para cada instrucción dada a continuación:
- Indicar el valor de los registros de uso específico luego de la búsqueda de la instrucción.
 - Indicar a qué celdas se acceden en cada etapa del ciclo de ejecución, en base a la información dada en cada una.

Cada instrucción está ensamblada a partir de la celda con dirección 1A40 (son independientes entre sí).

- MOV R1, [[0x0000]] y sabiendo que en la celda con dirección 0000 se encuentra el dato 0A44
 - MOV [R2], [R1], y sabiendo que R1 contiene el dato F57A y R2 el dato C32F.
 - MOV [[0x0000]], [[0x0001]], y sabiendo que en las celdas con dirección 0000 y 0001 se encuentran los datos 0A44 y B312 respectivamente.
 - ADD [R2], [R1], y sabiendo que R1 contiene el dato F57A y R2 el dato C32F.
 - ADD [[0x0000]], [[0x0001]], y sabiendo que en las celdas con dirección 0000 y 0001 se encuentran los datos 0A44 y B312 respectivamente.
12. Dada la siguiente rutina:

```
rutina: MOV R0, [R1]
        ADD [[0x1983]], R0
        RET
```

- Ensamblarla y alojar en memoria a partir de la celda con dirección 1810
- Simular la ejecución de la rutina, teniendo en cuenta el ciclo de ejecución de cada instrucción.