

Guía de ejercicios # 9: Arreglos y recorridos

Organización de computadoras

UNQ

Objetivos

Que el estudiante pueda:

- Comprender el concepto de arreglo
- Entender la necesidad de un modo indirecto
- Comprender la estructura de los programas que recorren arreglos

Alto nivel

1. Implementar la rutina `sumaElementos` (y completar el campo modifica)

sumaElementos	
Requiere	Un arreglo de 5 valores en <i>BSS</i> (16) almacenado a partir de la celda 0x3333
Modifica	??
Retorna	En R6 la suma de los elementos del arreglo

2. Implementar la rutina `cantElementos` (y completar el campo modifica)

cantElementos	
Requiere	Un arreglo de valores en <i>BSS</i> (16) almacenado a partir de la dirección que indica R0, y que finaliza con el primer elemento cuyo valor es 0x0000
Modifica	??
Retorna	En R6 la cantidad de elementos del arreglo

Nota: R0 tiene la dirección del primer elemento del arreglo y [R0] el valor de dicho elemento.

3. Escribir la rutina `copiarArreglo` en función de su documentación (y completar el campo modifica)

copiarArreglo	
Requiere	Un arreglo de valores en <i>BSS</i> (16) almacenado a partir de la celda 8400 y que finaliza con el primer valor FFFF
Modifica	??
Retorna	Una copia del arreglo original a partir de la celda 9400

4. Escribir la rutina `aplicarAbsolute` (y completar el campo modifica)

<code>aplicarAbsolute</code>	
Requiere	Un arreglo de valores en <code>CA2(16)</code> almacenado a partir de la celda <code>0x4486</code> , y cuya longitud está en la celda <code>0x4485</code> .
Modifica	??
Retorna	El mismo arreglo con sus elementos en valor absoluto

Nota: usar la rutina `absolute` de la práctica anterior.

5. Simule la ejecución de la rutina `aplicarAbsolute` sobre el siguiente mapa de memoria:

	...
4485	0004
4486	000A
4487	FFFF
4488	8000
4489	00FF
	...

6. Modifique la rutina `absolute` para que reciba como parámetro la dirección inicial del arreglo en `R0`.
7. En una fábrica de ventanas se codifican los pedidos en cadenas de 16 bits, y en caso de que la ventana esté pintada o lleve vidrio de seguridad es necesario usar un embalaje distinto. Se pide escribir la rutina `necesitaEmbalajePremium` que determine si el pedido es uno de esos casos. Completar además el campo **Modifica**.

<code>necesitaEmbalajePremium</code>	
Requiere	En <code>R5</code> un código de pedido de ventana, donde el bit 3 indica si debe estar pintada y el bit 9 indica si lleva el vidrio de seguridad.
Modifica	??
Retorna	un 1 en <code>R7</code> si el pedido que está en la celda <code>0x0019</code> requiere un embalaje <i>premium</i> , es decir, si se trata de una ventana pintada o con vidrio de seguridad.

Nota: Tener en cuenta que los demás bits también representan distintas características, pero para el ejercicio solo nos interesan esos dos.

8. Escriba la rutina `cantEmbalajeComun` a partir de su documentación y complete el campo modifica.

<code>cantEmbalajeComun</code>	
Requiere	Un arreglo de pedidos a partir de la celda <code>0x0001</code> , cuya longitud está en en la celda <code>0x0000</code>
Modifica	??
Retorna	en <code>R6</code> la cantidad de pedidos que no necesitan embalaje premium

Bajo nivel

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

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

Registro	Valor almacenado
R0	0x0001
R1	0x0003

Describir el **efecto** para cada instrucción:

- MOV [R0], [[0x0003]]
- MOV [[0x0003]], [R0]
- MOV [[0x0003]], [[0x0001]]
- MOV [[0x0001]], [R1]

Nota: El efecto se denota `dest <-- valor`, donde `dest` es el registro o posición de memoria que se termina modificando

10. Se cuenta con el siguiente programa:

```
MOV R1, [[0x0000]]
MOV [R2], [R1]
MOV [[0x0000]], [[0x0001]]
ADD [R2], [R1]
ADD [[0x0000]], [[0x0001]]
```

- Ensamblarlo a partir de la celda 0FF0
- Completar la siguiente tabla de accesos, indicando **el conjunto de celdas** que se accede en cada etapa sabiendo que:

- En la celda 0x0000 se encuentra el valor 0A44
- En la celda 0x0001 se encuentra el valor B312
- En el registro R1 se encuentra el valor F57A
- En el registro R2 se encuentra el valor C32F

Instrucción	B.Inst.	B.Op.	Alm.Op.