

# Guía de ejercicios # 7

## Estructura Condicional, Flags y Saltos

Organización de Computadoras 2020

UNQ

Los ejercicios de esta guía permiten:

- En alto nivel: Incorporar el manejo de la estructura condicional y comprender los conceptos de saltos absolutos o condicionales
- En bajo nivel: Comprender qué son y para qué se utilizan los Flags, conocer qué operaciones modifican los Flags y cómo los modifican, y distinguir los saltos de la invocación a rutinas.

### Alto nivel: escribir rutinas

1. Se cuenta con una la rutina `avg` cuya documentación es:

Documentación	
<b>Requiere</b>	En R0 y R1 dos valores en BSS(16)
<b>Modifica</b>	R1
<b>Retorna</b>	En R3 el promedio entre R1 y R2

Escribir una nueva rutina `promedioMayorADiez` que determine si el promedio de los valores almacenados en las celdas 1000 y 1004 es mayor al valor 10, en dicho caso ponga 1 en R1 o un 0 en caso contrario.

2. Dada la rutina `avg` mencionada arriba, escribir una rutina que calcule el promedio de los valores almacenados en las celdas 1000 y 1004 o los de 1005 y 1009 dependiendo de si R6 es mayor a R5 ó no (asumiendo que ambos contienen valores en CA2).
3. Implemente las dos rutinas del punto 10 de la practica anterior, las cuales poseen la siguiente documentación cada una:

minSinSigno	
<b>Requiere</b>	en R0 y R1 cadenas en BSS(16)
<b>Modifica</b>	?
<b>Retorna</b>	en R2 el valor minimo entre dichas cadenas, si son iguales, devuelve la primera

minConSigno	
<b>Requiere</b>	en R0 y R1 cadenas en CA2(16)
<b>Modifica</b>	?
<b>Retorna</b>	en R2 el valor minimo entre dichas cadenas, si son iguales, devuelve la primera

4. Implementar la rutina `multiplo` según su documentación:

Documentación	
<b>Requiere</b>	En R0 y R1 dos valores a analizar en CA2(16)
<b>Modifica</b>	?
<b>Retorna</b>	Un 1 en R5 si el número que esta en R0 es múltiplo de R1. En caso contrario R5 debe tener un 0

5. Implementar la rutina `negativo` según su documentación:

Documentación	
<b>Requiere</b>	En R1 un valor en CA2(16)
<b>Modifica</b>	?
<b>Retorna</b>	Un 1 en R0 si el valor de R1 es negativo, 0 en caso contrario.

6. Usando la rutina `multiplo`, hacer una rutina `esPar` que dado un numero en R1, retorne en R0 un 1 si el número de R1 es par, un 0 en caso contrario. Documente la rutina.
7. Usando la rutina `min`, hacer una rutina que le sume R3 el valor más chico entre lo que está guardado en la celdas CAFE y 1882.

## Bajo nivel: flags y simulación

8. Realizar las siguientes operaciones en `BSS(4)` y calcular los flags a partir de los resultados de las mismas.
- `1010 + 1001`
  - `1011 - 1011`
  - `0010 + 1101`
  - `0010 - 0111`
  - `1100 - 1000`
9. Dar los valores de R3 y R4 (calculando los flags de la primer instrucción) que evite que se llame a la rutina "boom":
- rutina: `CMP R3, R4`  
`JLE fin`  
`CALL boom`  
`fin: RET`
  - rutina: `CMP R3, R4`  
`JLEU fin`  
`CALL boom`  
`fin: RET`
  - rutina: `CMP R3, R4`  
`JL fin`  
`CALL boom`  
`fin: RET`

```
(d) rutina: CMP R3, R4
      JCS fin
      CALL boom
      fin: RET
```

10. Diseñar un circuito que calcule el **flag Z** a partir de una suma en *BSS(4)*. Considerar que se tiene disponible un sumador de 4 bits.

11. Considere la siguiente rutina.

```
rutina: MOV R3, [0x0A0A]
      SUB R0, 0x0001
      JE fin
      MOV R3, 0xFFFF
      fin: RET
```

- (a) Ensamblar a a partir de la celda *CAFE*.
- (b) ¿Que valor tiene el desplazamiento del salto *JE*?
- (c) ¿A que celda queda asociada la etiqueta *fin*?
- (d) Explique que hace la rutina

12. Dado el siguiente mapa de memoria, simule la ejecución de la rutina que comienza en la celda *A893*, asumiendo que *R0 = 0000* y *R1 = F000*

	...
A893	6821
A894	FC04
A895	1980
A896	FFFF
A897	A000
A898	A89B
A899	1980
A89A	AAAA
A89B	C000
	...

13. Simule la ejecución de la rutina dada en el ejercicio 12, asumiendo que *R0 = F000* y *R1 = 0000*

14. Simule la ejecución de la rutina dada en el ejercicio 12, reemplazando el valor de la celda *A894* por el valor *FA04* y asumiendo que *R0 = 0000* y *R1 = F000*.

15. Suponer que la instrucción *CMP [AAAA],R1* está ensamblada a partir de la celda *0000*.

- (a) ¿Qué celdas se acceden durante la búsqueda de instrucción?
- (b) ¿Qué celdas se acceden durante la búsqueda de operandos?
- (c) ¿Qué celdas se acceden durante la almacenamiento de operandos?

16. Suponer que la instrucción `JE esIgual` está ensamblada a partir de la celda 0000
- (a) ¿Qué celdas se acceden durante la búsqueda de instrucción?
  - (b) ¿Qué celdas se acceden durante la búsqueda de operandos?
  - (c) ¿Qué celdas se acceden durante la almacenamiento de operandos?
17. Suponer que la instrucción `JMP salir` está ensamblada a partir de la celda 0000
- (a) ¿Qué celdas se acceden durante la búsqueda de instrucción?
  - (b) ¿Qué celdas se acceden durante la búsqueda de operandos?
  - (c) ¿Qué celdas se acceden durante la almacenamiento de operandos?