

Guía de ejercicios # 8

Estructura Condicional, Saltos y Flags

Programación

1. Responder:

- (a) ¿Para qué sirve una alternativa condicional?
- (b) ¿Qué instrucciones intervienen para poder accionar ante una condición verdadera?
- (c) ¿Cómo se construye una alternativa condicional en el lenguaje Q? Dar un ejemplo simple
- (d) ¿Qué diferencia existe, en alto nivel, entre los saltos condicionales y el incondicional?
- (e) ¿Para qué sirven las etiquetas en los saltos?
- (f) “Los saltos condicionales siempre se ejecutan”. ¿Por qué esta proposición es verdadera? (analizar el ciclo de ejecución de una instrucción).

2. Dadas las siguientes especificaciones:

A) `calcularSegun20`: debe sumar los valores almacenados en `R0` y `R1`, sólo si el valor de la celda con dirección `076A` es menor o igual a 20, y debe multiplicarlos en caso contrario. Los valores están en `BSS(16)`.

B) `sumarIgual20`: suma los valores almacenados en `R0` y `R1`, sólo si el valor de la celda con dirección `076A` es igual a 20. Los valores están en `BSS(16)`.

- (a) Responder: ¿qué diferencia existe en términos de la alternativa condicional en cada caso?
Escribir cada algoritmo en pseudo-código (puede usar lenguaje QDraw).
- (b) Documentar cada rutina en base al pseudo-código.
- (c) Definir cada rutina en base a su documentación.
- (d) Para cada rutina ¿cómo se puede resolver el mismo problema pero con otro salto condicional? Escribir la nueva versión para cada rutina. **Nota:** se solicita utilizar etiquetas que comuniquen claramente cada caso.

3. Definir la rutina `maxSinSigno` que cumpla con la siguiente documentación:

maxSinSigno	
Requiere	en R3 y R4 dos valores en <code>BSS(16)</code>
Retorna	en R0 el valor máximo de ambos. De ser iguales, retorna el segundo.
Modifica	COMPLETAR

Responder: ¿qué modificaciones se deben realizar, tanto en la documentación como en la rutina, si los valores se encuentran en el sistema `CA2(16)`? A esta nueva rutina la llamaremos `maxConSigno`.

4. En base a la rutina `maxConSigno`, documentar y definir la rutina `sumarMax`, para que sume al valor contenido en `R7`, el máximo entre los valores de la celdas con dirección `1976` y `1982`.

Responder: ¿por qué no es necesario utilizar una alternativa condicional dentro de la rutina `sumarMax`?

5. Se cuenta con la siguiente documentación de la rutina `promedio` (**ya programada**):

promedio	
Requiere	en las celdas con dirección <code>3570</code> y <code>3571</code> dos valores en <code>CA(16)</code>
Retorna	en R3 el promedio entre ambos valores
Modifica	R1

- (a) Documentar y definir la rutina `promNeg` que retorne un 1 en la celda con dirección `D1EE` si el promedio entre los valores de `R5` y `R6` es menor a 0 (estricto), o un 0 en caso contrario.
- (b) Documentar y definir una rutina que retorne en R3 el promedio entre los valores de las celdas con dirección `BABA` y `50DA` si `R6` es menor o igual a `R5`; en caso contrario, el promedio debe ser entre los valores de las celdas con dirección `5700` y `ACDC`. Todos los valores están en `CA2(16)`
- (c) ¿Cómo se puede modularizar la rutina del ítem anterior? Escribir una rutina modularizada.

6. Escribir la rutina `múltiplo` a partir de la siguiente documentación:

múltiplo	
Requiere	en R0 y R1 dos valores en <code>CA2(16)</code>
Retorna	en R5 la cadena <code>000F</code> si el valor de R0 es múltiplo del valor de R1, o la cadena <code>F000</code> en caso contrario.
Modifica	COMPLETAR

Responder: ¿cómo se puede modularizar la rutina? Escribir una versión modularizada de la misma.

7. En base a la rutina `multiplo`, documentar y definir la rutina `esPar`, que dado un valor en la celda con dirección `DA70`, retorne la cadena `000F` en la celda con dirección `9A11` si dicho valor es par, o la cadena `F000` en caso contrario.

8. Teniendo en cuenta la resolución modularizada del ejercicio del múltiplo. Se solicita documentar y definir una rutina que, dado los valores de las celdas con dirección `DA7A` y `DA7E`, retorne el valor de `R4` incrementado en 1, sólo si el resto de dividir ambos valores es 1. Los valores están en `CA2(16)`.

9. Dada la siguiente documentación de la rutina `esNegativo`:

esNegativo	
Requiere	en R4 un valor en <code>CA2(16)</code>
Retorna	en R0 un 1 si el valor dado es negativo, o un 0 en caso contrario.
Modifica	COMPLETAR

Definir 2 versiones de la rutina que utilicen un salto diferente. Una de ellas debe contemplar un desplazamiento.

Prueba de rutinas

10. Responder:

- ¿Para qué sirve una rutina de test? ¿En qué nos basamos para escribirla?
- ¿Cuál es la estructura de una rutina de test?
- ¿Qué diferencia existe entre una rutina y su rutina de test? y ¿en cada documentación?

11. Documentar y definir 3 rutinas de test para probar la rutina `calcularSegun20` del ejercicio 2A) que contemple:

El parámetro `R0` contiene el valor 8 y `R1` el valor 2 para los 3 casos, pero el valor de la celda a analizar es diferente en cada uno:

- Contiene el valor 19, es decir menor a 20.
- Contiene el valor 20.
- Contiene el valor 21, es decir mayor a 20.

Justificar las cadenas de cada caso.

Responder: ¿algún test contempla el mismo resultado esperado? ¿Por qué?. ¿Por qué es importante probar con estos valores?

12. Documentar y definir 2 rutinas de test para probar la rutina `multiplo` del ejercicio 6. Un test para el caso en que los valores sean múltiplos, y otro con valores que no son múltiplos.

13. Documentar y definir una rutina de test para probar la rutina `esPrimo` (ya programada y cuyo código se desconoce).

La misma cumple con la siguiente documentación:

esPrimo	
Requiere	en R1 un valor en <code>BSS(16)</code>
Retorna	en la celda con dirección <code>4567</code> un 1 si el valor dado es primo, o un 0 en caso contrario.
Modifica	<code>R2</code>

Bajo nivel

14. Responder:

- ¿Cómo funciona la lógica interna de la instrucción `CMP`?
- ¿Cómo funciona la lógica de los saltos condicionales?
- Teniendo en cuenta las últimas 2 preguntas: ¿qué relación existe entre la instrucción `CMP` y los saltos condicionales?
- ¿Qué diferencia existe en bajo nivel entre los saltos condicionales y el incondicional?
- ¿Qué significa desviar el flujo? y ¿qué registro está involucrado?
- ¿Qué diferencias y similitudes comparten las instrucciones `CALL` y `JMP`?

15. Realizar las siguientes operaciones en `CA2(4)` y calcular los flags a partir de sus resultados:

- `1011 - 1011`
- `0010 - 0111`
- `0101 + 0100`
- `1010 + 1001`
- `0011 - 1000`

16. Para cada rutina, asignar valores a `R3` y `R4` (calculando los flags correspondientes), de manera de evitar la invocación a la rutina "boom":

- ```

explosionMenorIgual: CMP R3, R4
 JLE fin
 CALL boom
 fin: RET

```
- ```

explosionMayor:    CMP R3, R4
                    JG fin
                    CALL boom
                    fin: RET

```
- ```

explosionMayorSS: CMP R3, R4
 JGU fin
 CALL boom
 fin: RET

```
- ```

explosionMayorIgual: CMP R3, R4
                    JGE fin
                    CALL boom
                    fin: RET

```

17. Considerar la siguiente rutina:

```

ejemplo: CMP R0, 0x0001
          JE fin
          MOV R3, [0x0A0A]
          fin: RET
    
```

- (a) Ensamblar y alojarla en memoria a partir de la celda con dirección `CAFE`.
- (b) ¿Qué valor tiene el desplazamiento del salto `JE`?
- (c) ¿A qué dirección de celda queda asociada la etiqueta `fin`?

18. A partir del siguiente mapa de memoria, simular la ejecución de la rutina para:

- (a) Enumerar, en código fuente, las instrucciones en el orden en que se van ejecutando.
- (b) ¿Qué valor tiene el desplazamiento en el salto condicional? ¿Por qué?
- (c) Indicar el efecto asumiendo que `R0=0000` y `R1=F000`.
- (d) Indicar el efecto asumiendo que `R0=001D` y `R1=000D`.

Para ambos casos: `PC=A893`, y `SP=FFEE`.

	...
A893	6821
A894	FC04
A895	1980
A896	FFFF
A897	A000
A898	A89B
A899	1980
A89A	AAAA
A89B	C000
	...
FFEF	0500
	...

19. Para cada instrucción dada a continuación, indicar a qué direcciones de celdas se accede durante las etapas de búsqueda de instrucción (BI), búsqueda de operandos (BO) y almacenamiento de resultado (AR).

Asumir que cada instrucción está ensamblada a partir de la celda con dirección `17B0`.

- (a) `CMP [AAAA],R1`
- (b) `JE esIgual`, donde el valor de su desplazamiento es 6.
- (c) `JMP salir`, donde su etiqueta está ensamblada en la celda con dirección `9006`.

20. 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.