

**ESTRUCTURA DE COMPUTADORES (GradoII)**  
**RECUPERACIÓN DEL PRIMER PARCIAL (13 de enero de 2016)**

**1 (3 puntos)** Teniendo en cuenta el contenido de los registros y direcciones de memoria que se indican en la primera tabla y que se usa direccionamiento a memoria a nivel de palabra, rellene en la segunda tabla los campos que sean significativos para cada modo de direccionamiento.

Registros					Direcciones de Memoria											
R1	R2	R3	R4	PC		1000	1001	1002	1003	1004	1005	1006	1007	1008	1009	
1000	1005	5	1	1005		1005	5	1000	1005	1	1000	1	1000	1005	5	

**SOLUCIÓN**

	Denominación del modo de direccionamiento	Dirección efectiva	Contenido de la dirección efectiva	Objeto	Dónde reside el objeto
.R1	Absoluto a registro			1000	Registro
[.R2]	Indirecto a registro	1005	1000	1000	Memoria
#1000	Inmediato			1000	Instrucción
/1005	Absoluto a memoria	1005	1000	1000	Memoria
#5[R1]	Relativo a registro base	1005	1000	1000	Memoria
[/1005]	Indirecto a memoria	1000	1005	1005	Memoria
\$-5	Relativo a PC	1000	1005	1005	Memoria
\$3	Relativo a PC	1008	1005	1005	Memoria

**2 (7 puntos)** Se desea corregir un conjunto de exámenes de test. Cada una de las preguntas tiene **cuatro** posibles respuestas, de las cuales solo una es la correcta. A pesar de existir una sola respuesta correcta, el alumno puede marcar varias respuestas. De esta forma, la calificación del examen se obtiene sumando todas las respuestas correctas y restando  $\frac{1}{3}$  de la suma de todas las incorrectas.

Las cuatro respuestas de una pregunta del examen se representan mediante cuatro bytes, cada uno de los cuales contendrá un 1 si la respuesta ha sido marcada y un 0 en caso contrario.

**a) (30%)** Programe en ensamblador del MC88110 la subrutina `evalua(examen,correcto)`. El parámetro `examen` es una cadena de 40 caracteres que contiene las cuatro respuestas para las 10 preguntas de un examen. El parámetro `correcto` contiene la solución del examen, es decir, es una cadena de 40 caracteres (cuatro por pregunta) en la que solo una de las 4 posibles respuestas para cada pregunta contiene el valor 1 y las otras tres el valor 0. Ambas cadenas de caracteres se pasan en la pila por dirección. La subrutina debe devolver en `r29` el entero que resulta de restar al número de respuestas correctas la parte entera del número de las incorrectas dividido entre tres. A modo de ejemplo, si la subrutina recibe los parámetros que se muestran a continuación, devolvería el valor 6, puesto que tiene 7 respuestas correctas y 4 incorrectas:  $7 - E(\frac{4}{3})$

`examen = 1010100001101000100000010010000101000000`

`correcto = 0010010010001000100000010010000101001000`

**b) (70%)** Programe en ensamblador del MC88110 la subrutina `califica_examen(examenes,correcto)`, donde `correcto` es una cadena de 40 caracteres pasada por dirección en la pila que contiene la solución del examen. `examenes` es una tabla, que se pasa por dirección en la pila, en la que cada entrada está formada por tres campos:

- `id`: Es un entero de 4 bytes que contiene la identificación del alumno.
- `cal`: Es un entero de 4 bytes que deberá contener la calificación del alumno.
- `examen`: Es una cadena de caracteres de 40 bytes que contiene el examen del alumno que hay que calificar.

La función rellenará para cada alumno el campo `cal` de la tabla. el final de la tabla está marcado por una entrada cuyo `id` contiene el valor 0.

Para la realización de este ejercicio se llevará a cabo el tratamiento del Marco de Pila descrito en clase y se supondrá que están definidas todas las macros que se han explicado en la parte teórica de la asignatura; que las subrutinas llamantes dejan disponibles todos los registros excepto `r1`, `r30` (SP) y `r31` (FP); que la pila crece hacia direcciones de memoria decrecientes y el puntero de pila apunta a la última posición ocupada.

## SOLUCIÓN

**a)** En este caso las variables que se van a utilizar se pueden ubicar en registros y no se va a realizar ninguna llamada anidada a ninguna subrutina por lo que no se crea marco de pila. La subrutina va recorriendo las dos cadenas de caracteres hasta alcanzar el carácter número 40. Para cada carácter de la respuesta se compara si es 0. Si lo es no es necesario sumar nada ni a las respuestas correctas ni incorrectas, puesto que no hay respuesta. Si ha marcado la respuesta hay que comprobar si la respuesta es correcta. Si es así, se incrementa el registro que lleva la cuenta de las respuestas correctas (`r2`). Si no, se incrementa el registro que lleva la cuenta de las respuestas incorrectas (`r3`).

```

evalua: ld r20,r30,0      ; Examen
        ld r21,r30,4      ; Resultado correcto del examen
        or r2,r0,r0       ; Número de respuestas correctas
        or r3,r0,r0       ; Número de respuestas incorrectas
        or r4,r0,r0       ; Contador y desplazamiento a r20 y r21
bucle:  cmp r7,r4,40      ; Si se ha llegado a 40 respuestas se acaba
        bbl eq,r7,fin
        ld.bu r5,r20,r4   ; Se carga una respuesta del examen
        ld.bu r6,r21,r4   ; Se carga la respuesta de la solución
        and r5,r5,0x0f    ; Se tienen en cuenta solo los cuatro bits
        and r6,r6,0x0f    ; menos significativos del dígito

```

```

    addu r4,r4,1    ; Se incrementa la cuenta
    cmp r7, r5,r0   ; Si la respuesta es 0 no se hace nada
    bbl eq,r7,bucle
    cmp r7,r5,r6    ; Si la respuesta es 1: hay que sumar 1 a las correctas o incorrectas.
    bbl eq,r7,cont
    addu r3,r3,1    ; La respuesta es 1 y no es igual a la respuesta correcta. Se suma 1
    br bucle       ; a las incorrectas
cont:  addu r2,r2,1
      br bucle
fin:   divu r3,r3,3 ; Se divide las incorrectas entre tres y
      subu r29,r2,r3 ; se calcula la calificación
      jmp(r1)

```

b) Puesto que la solución incluye una llamada a la subrutina del apartado anterior, se crea el marco de pila y se almacena la dirección de retorno en la pila. Se crea una variable local ubicada en la pila que contendrá el puntero a la tabla de exámenes. La subrutina recorre la tabla de exámenes comprobando si el identificador es 0. Cuando lo es, se sale de la subrutina. Si no lo es, se invoca a la subrutina *evalua* con la dirección de comienzo del campo *examen* y el parámetro *correcto*. El valor de retorno de la subrutina se almacena en el campo *cal* de la entrada en curso de la tabla.

```

califica_examenes:
    PUSH(r1)
    PUSH(r31)
    or r31,r30,r30
    subu r30,r30,4 ; Se reserva espacio para el puntero a la tabla de exámenes
    ld r20,r31,8  ; Puntero a la tabla
b_cal:  ld r3,r20,r0
      cmp r7,r3,r0 ; Si el id es 0 se acaba
      bbl eq,r7,fin_cal
      st r20,r31,-4 ; Se almacena el puntero en su variable local.
      ld r21,r31,12 ; Puntero a la solución del examen
      PUSH(r21)
      addu r20,r20,8 ; Puntero al examen del alumno
      PUSH(r20)
      bsr evalua
      addu r30,r30,8
      ld r20,r31,-4
      st r29,r20,4 ; Se almacena la calificación en el campo cal de la tabla
      addu r20,r20,48
      br b_cal
fin_cal: or r30,r31,r31
      POP(r31)
      POP(r1)
      jmp(r1)

```

NOTAS: 22 de enero de 2016  
 REVISIÓN: 25 de enero de 2016

DURACIÓN: 1 hora y 45 minutos  
 PUNTUACIÓN: 10 puntos