

Test de Teoría (3.0p)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
b	b	d	c	d	b	c	c	b	b	a	a	a	d	b	d	d	b	c	d	a	c	a	d	b	a	c	a	b	c

Test de Prácticas (4.0p)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
c	b	d	c	c	a	c	c	d	d	d	a	a	a	b	d	b	b	d	a

Examen de Problemas (3.0p)

1. Ensamblador (1 punto).

11+7=18 instrucciones, 3+3=6 etiquetas, 2+2=4 directivas, alrededor de **0.05p**/instrucción, **0.02p**/etiqueta, total $18 \times 0.05p + 6 \times 0.02p = \mathbf{0.9p + 0.12p \geq 1p}$ total

Alternativamente, **0.5p** cada programa. Las puntuaciones son aproximadas, de todas formas.

Solución:

32 bits (0.5 puntos, 11 instrucciones máquina)	64 bits (0.5 puntos, 7 instrucciones máquina)
<pre> .text .globl mystrlen mystrlen: pushl %ebp movl %esp, %ebp xorl %eax, %eax ; len = 0 movl 8(%ebp), %edx ; edx = &s[0] cmpb \$0, (%edx) ; s[0] == '\0'? je .L2 ; == '\0' ==> end .L3: addl \$1, %eax ; len++ cmpb \$0, (%edx,%eax) ; s[len] == '\0'? jne .L3 ; != '\0' ==> loop .L2: popl %ebp ret </pre>	<pre> .text .globl mystrlen mystrlen: xorl %eax, %eax ; len = 0 cmpb \$0, (%rdi) ; s[0] == '\0'? je .L4 ; == '\0' ==> end .L3: addq \$1, %rax ; len++ cmpb \$0, (%rdi,%rax) ; s[len] == '\0'? jne .L3 ; != '\0' ==> loop .L4: ret </pre>

2. Unidad de Control (0.6 puntos).

8 micro-instrucciones, 12 micro-pseudo-ops (incluyendo goto fetch).

Se puntúa **0.05p** por micro-pseudo-op (total **0.05 x 12 = 0.60p**)

```

pop:    MAR := SP[27:0]
        SP := SP + (+1); MBR := M[MAR]      // Leer M[SP++]
        MAR := IR[27:0]
        M[MAR] := MBR; goto fetch           // Escribir M[dir]

push:   MAR := IR[27:0]
        SP := SP + (-1); MBR := M[MAR];     // Leer M[dir]
        MAR := SP[27:0]
        M[MAR] := MBR; goto fetch           // Escribir M[--SP]

```

3. Entrada/salida (0.4 puntos).

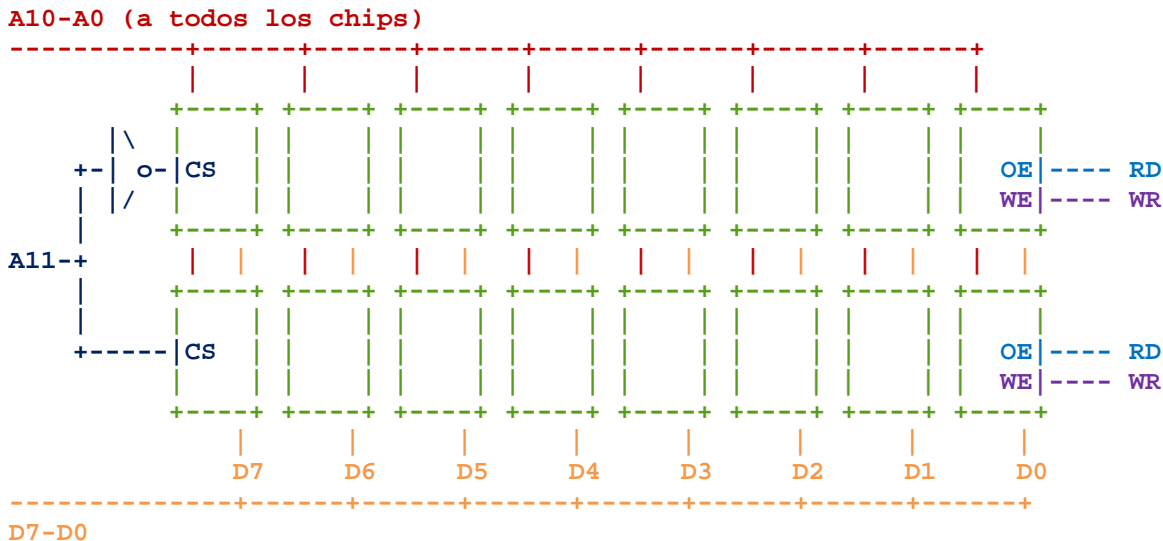
Se puntúa **0.1p** por operación indicada, **0.1p** por valor correcto, total **2 x 0.1 + 2 x 0.1 = 0.4p**

$30 \text{ pollings/s} * 400 \text{ ciclos/polling} = 12000 \text{ ciclos/s}$

$\text{Porcentaje} = 100 * 12\,000 \text{ ciclos/s} / 2\,000\,000\,000 \text{ ciclos/s} = 12 / 20000 = 0,0006\%$

4. Configuración de memoria (0.5 puntos).

Aproximadamente **0.1p** por zona (**CS/A₁₁**, **WE/WR**, **OE/RD**, **A₁₀₋₀**, **D₇₋₀**)



5. Memoria cache (0.5 puntos).

Solución:

a) Indique la primera y la última direcciones virtuales en hexadecimal:

0.1p

0x00 0000 0000

0x7F FFFF FFFF

b) (0,1) Indique la primera y la última direcciones físicas en hexadecimal:

0.1p

0x00 0000 0000

0x3F FFFF FFFF

c) (0,3) Indique los nombres y tamaños de los campos en los que se divide una dirección de memoria física de memoria desde el punto de vista de una cache L1.

0.3p

37363534333231302928272625242322212019181716151413121110 9 8 7 6 5 4 3 2 1 0

(26) (6) (6)

etiqueta conjunto byte

Explicación:

2^{15} B
 ----- = 2^9 líneas
 2^6 B/línea

2^9 líneas
 ----- = 2^6 conjuntos
 2^3 líneas/conjunto

$38 - 12 = 26$ bits cada etiqueta

Referencias:

<https://static.dev.sifive.com/U54-MC-RVCoreIP.pdf>

<https://riscv.org/specifications/privileged-isa/>