

Nombre:

DNI:

Grupo:

Sobre 10, cada respuesta vale 2 si es correcta, 0 si está en blanco o claramente tachada, y -2/3 si es errónea.
Anotar las respuestas (**a**, **b**, **c** o **d**) en la siguiente tabla.

1	2	3	4	5
a	d	c	b	b

- En los modos de direccionamiento del tipo Desplazamiento(Base,Indice,FactorEscala), puede usarse como... **Tema2.2, tr.4**
 - desplazamiento, cualquier constante de 1, 2 o 4 bytes (incluso el nombre de una variable, por su dirección)
 - base, cualquiera de los 8 registros enteros salvo %esp **sería cualquiera**
 - índice, también cualquiera salvo %ebp **sería %esp**
 - factor de escala, cualquier constante de 1, 2, 4 u 8 bytes **sin bytes**
- ¿Cuál de las siguientes secuencias de instrucciones calcula $a=b-a$, suponiendo que %eax es a y %ebx es b?
 - subl %eax, %ebx **b=b-a**
 - subl %ebx, %eax **a=a-b**
 - notl %eax
addl %ebx, %eax **sería negl**
 - Varias o ninguna de las respuestas anteriores son correctas, no se puede marcar una y sólo una **Tema2.2, tr.9**
- La diferencia entre las instrucciones test y cmp consiste en que...
 - test realiza una operación and lógico, mientras que cmp realiza una resta
 - test modifica sólo los flags lógicos (ZF,SF) mientras que cmp modifica los aritmético-lógicos (ZF,SF,CF,OF)
 - ambas respuestas son correctas **Tema2.2, tr.21-22**
 - ambas respuestas son incorrectas
- Para traducir una construcción if-then-else de lenguaje C a lenguaje ensamblador, gcc utiliza generalmente...
 - Un salto condicional, según la condición expresada en el código C
 - Un salto condicional, según la condición opuesta a la del código C, y otro salto incondicional **Tema2.2, tr.28-32**
 - Dos saltos condicionales (uno para la parte if y otro para la parte else)
 - Dos saltos condicionales y dos saltos incondicionales
- Para traducir una construcción “do-while” de lenguaje C a lenguaje ensamblador, gcc utiliza generalmente ...
 - un salto condicional hacia adelante, según la misma condición que en lenguaje C
 - un salto condicional hacia atrás, según la misma condición que en lenguaje C **Tema2.2, tr.38-40**
 - un salto condicional hacia adelante, según la condición opuesta a la de lenguaje C
 - un salto condicional hacia atrás, según la condición opuesta a la de lenguaje C