

Nombre:	
DNI:	Grupo:

Examen de Problemas (3.0p)

1. Bucles for (0.5 puntos). Una función, fun_c, tiene la siguiente estructura general:

```
long fun_c(unsigned long x) {
    long val = 0;
    int i;
    for ( _____ ; _____ ; _____ ; ) {
        _____;
    }
    _____;
    return _____;
}
```

El compilador C GCC genera el siguiente código ensamblador:

```
# x en %rdi
fun_c:
    movl    $0, %ecx
    movl    $0, %edx
    movabsq $72340172838076673, %rsi
    # Mover un valor de 64b

.L2:
    movq    %rdi, %rax
    andq    %rsi, %rax
    addq    %rax, %rcx
    shrq    %rdi
    # Desplazar a derecha 1
    addl    $1, %edx

    cmpl    $8, %edx
    jne     .L2
    movq    %rcx, %rax
    sarq    $32, %rax
    addq    %rcx, %rax
    movq    %rax, %rdx
    sarq    $16, %rdx
    addq    %rax, %rdx
    movq    %rdx, %rax
    sarq    $8, %rax
    addq    %rdx, %rax
    andl    $255, %eax
    ret
```

Analizar el funcionamiento de este código y responder a las siguientes preguntas.
Resultará útil convertir la constante decimal de la línea 4 (`movabsq`) a hexadecimal.

- A. Usar la versión ensamblador para rellenar las partes que faltan del código C.
B. Describir en castellano qué calcula esta función.

2. Representación y acceso a estructuras (0.5 puntos). Para la declaración de estructura

```
struct {
    char      *a;
    short     b;
    double    c;
    char      d;
    float     e;
    char      f;
    long long g;
    void      *h;
} foo;
```

suponer que fue compilada en una máquina Windows (32b), donde cada tipo de datos primitivo de K bytes debe tener un desplazamiento múltiplo de K.

A. ¿Cuál es el desplazamiento en bytes de cada campo de la estructura?

B. ¿Cuál es el tamaño total de la estructura?

C. Reordenar los campos de la estructura para minimizar el espacio desperdiciado, y entonces mostrar los desplazamientos en bytes y tamaño total para la estructura reordenada.

Responder mediante una tabla con el siguiente formato:

(A)	nombre	n.campo2	total (B)
tamaño	tamaño campo1	t.campo2	tamaño
desplaz	desplaz. campo1	d.campo2	total
(C)	nombre	n.campo2	total
tamaño	tamaño campo1	t.campo2	tamaño
desplaz	desplaz. campo1	d.campo2	total

- Entrada/Salida** (0.5 puntos). Disponemos de un microprocesador de 8 bits (bus de datos de 8 bits y bus de direcciones de 16 bits) con E/S independiente. Diseñe un sistema de E/S que permita acceder a los siguientes puertos: puerto 0x0220 de salida y 0x0221 de entrada. Utilice lógica de decodificación distribuida. No emplee decodificadores.
- Diseño del sistema de memoria** (0.5 puntos). Diseñe un sistema de memoria para un procesador de 32 bits (cada dirección de memoria corresponde a una palabra de 32 bits) a partir de módulos SRAM 4K×8 y ROM 8K×16. La memoria SRAM debe ocupar las direcciones 0x0000 a 0x1FFF y la ROM las direcciones 0xC000 a 0xFFFF.
- Memoria cache** (0.5 puntos). Los parámetros que definen la memoria de un computador son los siguientes:
 - Tamaño de la memoria principal: 4 GB
 - Tamaño de la memoria cache: 64 KB
 - Tamaño de bloque: 256 B

Determine el tamaño de los distintos campos de una dirección en las siguientes condiciones:

- Correspondencia completamente asociativa
- Correspondencia directa
- Correspondencia asociativa por conjuntos con 16 bloques por conjunto

6. **Unidad de control** (0.5 puntos). La figura muestra el camino de datos de un procesador de 32 bits que direcciona la memoria por bytes y en el que cada instrucción y cada dato ocupa una palabra completa (4 bytes). El multiplexor MUX2 selecciona o bien la salida del registro Y o un valor constante igual a 4 utilizado para incrementar el contenido del contador de programa. Las operaciones de lectura o escritura en memoria consumen un ciclo de reloj. Se desean implementar tres instrucciones de suma:

<i>Instrucción</i>	<i>Direccionamiento del operando fuente</i>	<i>Formato</i>
addr Rdst,Rsrc	registro	una palabra
addi Rdst,[Rsrc]	indirecto a memoria a través de registro	una palabra
addx Rdst,[Rsrc+desp]	indexado	dos palabras, la segunda contiene el desplazamiento

Escriba (en lenguaje de transferencia de registros o de alto nivel) un microprograma que incluya la fase de captación de instrucción y la fase de ejecución de cada una de esas tres instrucciones.

