COGNOMS:

NOM: DNI/NIE:

IMPORTANTE leer atentamente antes de empezar el examen: Escriba los apellidos, el nombre y el DNI/NIE antes de empezar el examen. Escriba un solo carácter por recuadro, en mayúsculas y lo más claramente posible. Es importante que no haya tachones ni borrones y que cada carácter quede enmarcado dentro de su recuadro sin llegar a tocar los bordes. Use un único cuadro en blanco para separar los apellidos y nombres compuestos si es el caso. No escriba fuera de los recuadros, todo lo que haya fuera de ellos es ignorado. La identificación del alumno se realiza de forma automática, no seguir correctamente estas instrucciones puede comportar no tener nota.

Problema 1. (5 puntos)

El departamento de producción de IBM quiere explorar los costes de sustituir los antiguos procesadores Power5 fabricados en tecnología de 130 nm por otros nuevos (Power5+) fabricados en tecnología de 90 nm. La siguiente tabla muestra algunas de las características de los dos procesadores relacionadas con los costes de fabricación:

chip	área del dado (en mm^2)	defectos por cm^2	complejidad (α)	diámetro de la oblea (en mm)
Power5 viejo	389	30%	4	300
Power5+ nuevo	186	70%	4	300

Recordatorio de formulas no usadas habitualmente (ten cuidado con las unidades):

$$\text{Dies per waffer} = \frac{\text{Area útil}}{\text{Die area}} = \frac{\pi \times (\text{diametro}/2)^2}{\text{Die area}} - \frac{\pi \times \text{diameter}}{\sqrt{2 \times \text{Die area}}}$$

$$\text{Die yield} = \text{Waffer yield} \times \left(1 + \frac{\text{defectos por unidad de area} \times \text{die area}}{\alpha}\right)^{-\alpha}$$

a) **Calcula** cuántos dados totales salen de una oblea para cada uno de esos dos procesadores.

b) **Calcula** el *die yield* (porcentaje de dados correctos) de una oblea para esos dos procesadores. Suponer un *waffer yield* igual a 1.

c) **¿Cuál** es el coste de fabricación de cada dado correcto para cada tipo de procesador?. Asume que el coste de fabricar una oblea es de 500 euros.

Suponed que los chips viejos del IBM Power5 se han estado vendiendo por un 40% más del coste de fabricación de un dado, y que los chips nuevos Power5+ se van a sacar a la venta al doble del precio de un chip viejo.

Durante la fase de empaquetado el 100% de los dados producen chips plenamente funcionales. El coste de empaquetado se considera despreciable respecto el coste del dado.

d) **¿Cuál** es el precio de venta del nuevo procesador Power5+ y el beneficio que obtendrá IBM por la venta de cada chip nuevo?.

Las ventas de procesadores Power5 viejos ha sido de 500000 chips por mes y las previsiones de mercado anticipan unas ventas por mes de tres veces más de los nuevos procesadores Power5+. Sabemos también que IBM ha gastado mil millones de euros para desarrollar el Power5+ nuevo en 90 nm.

e) **Calcula** durante cuanto tiempo (años) debe IBM mantener ese flujo de ventas para amortizar los costes de desarrollo del nuevo chip Power5+.

El departamento de promoción de ventas de IBM asegura que cambiar a los nuevos procesadores mejorará la parte de cálculo de nuestros programas en un 36%. Para valorar la efectividad del cambio estudiamos el comportamiento de uno de nuestros programas representativos que es intensivo en cálculo, a lo que dedica el 90% del tiempo. El 10% restante lo dedica a esperar por operaciones de Entrada/Salida, para la que los nuevos procesadores no incorporan ninguna mejora.

f) **Calcula** la ganancia de nuestro programa completo al ejecutarlo con el nuevo procesador.

Sabiendo que con el procesador viejo nuestro programa tarda 37.02 segundos en ejecutar las instrucciones de cálculo, y que el programa ejecuta 522 millones de instrucciones dinámicas.

g) **Calcula** los MIPS para los dos procesadores (viejo y nuevo).

COGNOMS:

NOM: DNI/NIE:

Problema 2. (5 puntos)

Dado el siguiente código escrito en C, que compilamos para un sistema linux de 32 bits:

```
typedef struct {
  char a;
  short b;
  char c;
  short d[4];
  } s1;
```

```
typedef struct {
  int e;
  short f;
  s1 g[6];
    } s2;
```

a) **Dibuja** cómo quedarían almacenadas en memoria las estructuras **s1** y **s2**, indicando claramente los desplazamientos respecto al inicio, el tamaño de todos los campos y el tamaño de los structs.

b) **Define** de forma CLARA Y CONCISA el concepto de localidad espacial y cómo se aprovecha este concepto en el funcionamiento de la jerarquía de memoria.

c) Se dispone de una Memoria cache de 4 KB, asociativa por conjuntos, con líneas de 32 bytes y 8 líneas por conjunto. Dada una dirección de memoria principal de 32 bits, A31..0, siendo A31 el bit de mayor peso y A0 el bit de menor peso. Indica qué bits de la dirección corresponden al desplazamiento dentro de la línea, qué bits corresponden al conjunto y qué bits corresponden a la etiqueta. Justifica la respuesta. Puedes poner un dibujo o indicarlo con palabras

Dado el siguiente código escrito en C, que compilamos para un sistema linux de 32 bits:

```
int examen(short d[2][2], char c, short b, short *a) {
  short e;
  char f;
  short g[2][2];
  short *h;
   . . .
  x=examen(g,f,e,h);
  . . .
}
```

d) **Dibuja** el bloque de activación de la rutina examen, indicando claramente los desplazamientos respecto a **%ebp** y el tamaño de todos los campos. Si el parámetro se pasa por referencia o es un puntero, pon una @ antes del nombre. En caso de chars y shorts que requieran alineamiento, tanto en los parámetros como en las variables locales, indica claramente su posición y los bytes que quedan "vacíos" para conseguir el alineamiento.

e) **Traduce** a ensamblador x86 la instrucción `d[1][0]=d[1][1],` que se encuentra en el interior de la subrutina examen, usando el mínimo número de instrucciones. Se valorará que se usen registros que puedan ser modificados por la subrutina examen sin necesidad de salvarlos previamente en el bloque de activación.