

Material permitido: Solo calculadora no programable	Aviso 1: Todas las respuestas deben estar debidamente razonadas.
Tiempo: 2 horas	Aviso 2: Escriba con buena letra y evite los tachones.
N2	Aviso 3: Solución del examen y fecha de revisión en http://www.uned.es/71902048/

1. Conteste **razonadamente** a los siguientes apartados:

- a) (1 p) ¿Qué información se suele almacenar de forma general en una entrada de un directorio?
 - b) (1 p) Describir la planificación de hilos en función del tipo de hilos soportado por el sistema operativo.
2. (2 p) Enumerar las acciones que debe realizar el sistema operativo para crear un proceso.
3. (2 p) Explicar **razonadamente** qué es el *buffering* y comentar los problemas que resuelve.
4. (2 p) En una oficina de Correos existen 3 ventanillas de atención al cliente. Cuando un cliente entra en la oficina para realizar alguna gestión debe guardar una única cola hasta que alguna ventanilla queda libre. Explicar **razonadamente** si el pseudocódigo del programa que se muestra en la Figura 1 coordina adecuadamente la actividad de los clientes en la oficina. En caso negativo modifique el programa para que funcione correctamente.
5. (2 p) El sistema operativo en colaboración con el hardware gestiona la memoria principal mediante paginación por demanda. La traducción de direcciones se realiza usando una MMU con un TLB. El tiempo medio de acceso al TLB es despreciable y su tasa de aciertos es del 95 %. Determinar el tiempo medio de despacho de una referencia a memoria si se tienen: una tasa de fallos de página del 8 %, un tiempo medio de acceso a memoria de 125 ns y un tiempo medio de gestión de un fallo de página de 40 ms. Despreciar la existencia de memoria caché.

Material permitido: Solo calculadora no programable	Aviso 1: Todas las respuestas deben estar debidamente razonadas.
Tiempo: 2 horas	Aviso 2: Escriba con buena letra y evite los tachones.
N2	Aviso 3: Solución del examen y fecha de revisión en http://www.uned.es/71902048/

```
int Z=0;
semáforo_binario X, Y;

void cliente()
{
    wait_sem(X);
    Z = Z + 1;
    if (Z > 3){
        signal_sem(X);
        wait_sem(Y);
    }

    signal_sem(X);
    realizar_gestión();
    wait_sem(X);
    Z = Z - 1;
    signal_sem(Y);
    signal_sem(X);
}

main()
{
    init_sem(X,1);
    init_sem(Y,0);
    ejecución_concurrente(cliente,...,cliente);
}
```

Figura 1