



| | | | | | |
|---|---|---|------|-------|------|
| 1 | 2 | 3 | test | extra | NOTA |
|---|---|---|------|-------|------|

Nombre y apellidos

DNI/NIE

| | |
|--|--|
| | |
|--|--|

DURACIÓN: Dispone de 105 minutos para realizar el examen.

En los ejercicios justifique TODAS sus respuestas. Las respuestas que sólo den resultados sin justificar se considerarán nulas.

Lea las instrucciones para el test en la hoja correspondiente.

1 (1,75 puntos) El sistema operativo ofrece funciones y servicios que persiguen mejorar aspectos bien distintos del sistema, tales como la seguridad, el rendimiento y la usabilidad. Por cada uno de esos tres aspectos, tiene usted que dar un ejemplo de un servicio del sistema operativo cuyo objetivo principal consista en mejorar ese aspecto particular. En cada caso, justifique por qué ha elegido ese servicio.

Estos son los tres aspectos para los que tiene que encontrar un servicio del S.O. que sirva de ejemplo:

- Mejorar la protección y seguridad del sistema.
- Aumentar el rendimiento del sistema (velocidad, aprovechamiento de los recursos, etc.).
- Mejorar la usabilidad del sistema (facilitar a los usuarios la interacción con el sistema).

Utilice entre 75 y 150 palabras para su exposición.

2 (1,25 puntos) En una máquina con un solo procesador tenemos un planificador de procesos que utiliza el algoritmo Round Robin con Q=1 milisegundo. El tiempo de cambio de contexto se considera despreciable. En la cola de preparados tenemos un conjunto de 20 procesos recién llegados, todos ellos con intención de ejecutar una ráfaga de 2 milisegundos de duración. Bajo estas condiciones, obtenga los siguientes datos:

- ¿Cuál será el tiempo de retorno del primer proceso en finalizar?
- ¿Cuánto tardará en vaciarse toda la cola?
- ¿Cuántos cambios de contexto serán necesarios para planificar toda esta carga?
- ¿Cuál será el tiempo medio de espera para los procesos de la cola?

3 (1 punto) Se propone el siguiente algoritmo para gestionar el acceso a una sección crítica para dos procesos. Analice si verifica todas las propiedades que debe cumplir una solución aceptable y por tanto puede considerarse un algoritmo correcto.

| | |
|--|--|
| <pre>// variables globales bool sc1 = false, sc2 = false; int turno = 1;</pre> | |
| <pre>Código del proceso 1 11 while (true) { 12 ... Sección no crítica ... 13 sc1 = true; 14 while (turno==2 && sc2) {} 15 ... SECCIÓN CRÍTICA ... 16 turno = 2; 17 sc1 = false; 18 }</pre> | <pre>Código del proceso 2 21 while (true) { 22 ... Sección no crítica ... 23 sc2 = true; 24 while (turno==1 && sc1) {} 25 ... SECCIÓN CRÍTICA ... 26 turno = 1; 27 sc2 = false; 28 }</pre> |