UTN – 1º Rec 1° Parcial	Sistemas Operativos	23/05/2015

Nombre y Apellido:..... Curso: ….....

TEORÍA					PRÁCTICA			NOTA
1	2	3	4	5	1	2	3	

TEORÍA: Responda brevemente las siguientes preguntas. Justifique.

1. **Explique detalladamente las operaciones implicadas en un process switch ocurrido al terminar un quantum**
2. **Compare los algoritmos Virtual Round Robin y Round Robin usando los siguientes criterios: atención de procesos I/O bound, inanición y tiempo de respuesta promedio.**
3. **Dé un ejemplo en pseudocódigo donde sea necesario utilizar el stack, el heap y la sección de datos del proceso. Indique la relación entre estos elementos y el código que escribió.**
4. **Elija dos condiciones necesarias para que ocurra deadlock e indique dos formas para prevenirlas (una para cada una). ¿En cuál situación sería preferible utilizar técnicas de Prevención en lugar de técnicas de Evasión?**
5. **Escriba el pseudocódigo de las operaciones wait y signal (bloqueantes), indicando en dónde podría ocurrir una condición de carrera y resolviéndola a través de algún mecanismo que sea idóneo para sistemas multiprocesador.**

Bonus

Dada la función presentada a la derecha, explique detalladamente:

a. **Las ventajas de declarar 'str' de dicha forma en vez de usar un puntero combinado con malloc()**
b. **¿Qué problema común podría tener con una función que sea recursiva?**

```
fopen2(char *dir, char *filename, char *mode)
  {
  char str[strlen(dir) + strlen(filename) + 1];
  strcpy (str, dir);
  strcat (str, filename);
  return fopen (str, mode);
  }
```

PRÁCTICA: Resuelva los siguientes ejercicios justificando las conclusiones obtenidas.

Ejercicio 1

	R1	R2	R3	R4		R1	R2	R3	R4
P1	1	1	0	0	P1	2	0	0	1
P2	1	0	2	1	P2	0	0	2	0
P3	0	0	1	0	P3	1	0	1	0
P4	1	0	0	1	P4	0	0	0	1
Recursos asignados					Peticiones actuales				

a. Teniendo en cuenta las matrices de **Recursos Asignados** y **Peticiones Actuales** indique cuál debería ser el vector **Recursos Totales** mínimo para que no haya deadlock (la suma de las instancias de todos los recursos debe ser la mínima).
b. Utilizando la matriz de **Recursos asignados.** interpretando en esta ocasión a la matriz de **Peticiones Actuales** como **Necesidad** (recursos que todavía podría pedir) y su vector de **Recursos Totales** obtenido en a) proponer una petición de recursos válida por parte de un proceso que no sea aprobada por estado inseguro.

Ejercicio 2

Un SO planifica sus procesos utilizando 3 colas de Ready.

	Arribo	CPU	IO	CPU	IO	CPU
A	0	5	1	2	-	-
B	2	1	2	1	1	1
C	5	2	2	2	-	-
D	6	6	-	-	-	-

- Los procesos nuevos ingresan a una cola de prioridad intermedia que utiliza RR con Q = 3. Cuando los mismos son desalojados de dicha cola por fin de Q se mueven a la cola de menor prioridad que utiliza FIFO. Los mismos son promovidos a la cola intermedia luego de volver de una IO.
- Cuando un proceso no consume todo su Q, al volver de IO se colocará en una cola de mayor prioridad en donde se le dará un Q igual a lo que no ejecutó de su Q anterior (su remanente). En caso de ser desalojado por fin de Q en dicha cola, se lo llevará al final de la cola RR de prioridad media.
- La cola FIFO será desalojada si un proceso llega a una cola de mayor prioridad.(el proceso desalojado se colocará al principio de la cola FIFO). Sin embargo, las colas RR son con desalojo entre ellas únicamente con respecto al quantum.

Grafique la ejecución de los siguientes procesos en un diagrama de Gantt.

Ejercicio 3

Homero es un buen padre que juega con sus hijos alternadamente para que no se pongan celosos. Estableció que primero juega con Maggie, luego con Lisa y luego con Bart (para luego volver con maggie, y repetir el ciclo). Cada tanto sus hijos deciden usar el sofá, para el cual solo hay dos lugares. Finalmente, si la cantidad de veces que Homero jugó es mayor a 100, los hijos se compadecen y deciden irse a dormir.
Dado el siguiente pseudo-código:

Bart	Lisa	Maggie	Homero

`while(true){`
`jugar_con_homero();`
`sentarse_en_sofa();`
`if (juegos>100) {`
`exit();`
`}`
`}`
	`while(true){`
	`jugar_con_homero();`
	`sentarse_en_sofa();`
	`if (juegos>100) {`
	`exit();`
	`}`
	`}`
		`while(true) {`
		`jugar_con_homero();`
		`sentarse_en_sofa();`
		`if (juegos>100) {`
		`exit();`
		`}`
		`}`
			`while(true){`
			`jugar_con_hijo();`
			`juegos++;`
			`}`

Sincronice los hilos usando exclusivamente semáforos exclusivamente para respetar el enunciado, sin que se produzca deadlock ni starvation y utilizando la menor cantidad de semáforos posibles.

Condiciones de aprobación: 3 preguntas correctamente respondidas y 1,5 ejercicios correctamente resueltos. La pregunta bonus es opcional, pero suma en caso de ser respondida.