

# CERTAMEN #1 INF-250

## Pauta de Corrección

Wenceslao Palma <wenceslao.palma@ucv.cl>

- (4 ptos c/u) Responda las sgtes preguntas. Justifique cada una de sus respuestas.
  - Considerando la administración de procesos en Unix. Cuándo un proceso se encuentra en estado ready y almacenado en el disco? R.: Cuando no existe memoria suficiente para almacenar todos los procesos que se encuentran en el sistema. Un proceso puede pasar a dicho estado estando en memoria y en estado ready, cuando se encuentra durmiendo en memoria o bien cuando acaba de ingresar al sistema.
  - Al realizar una llamada al sistema, ocurre un cambio de contexto? R.: si. Una llamada al sistema es una invocación que realiza un proceso hacia para obtener un servicio otorgado por el kernel. En este caso el proceso que realiza la llamada al sistema pasa al estado bloqueado.
  - Por qué las primitivas wait y signal deben ser atómicas? R.: porque si no lo fueran es posible generar inconsistencias en el valor del semáforo.
  - Cuál es la diferencia entre la utilización de fork() y threads? R.: cuando se utiliza fork se realiza una copia del proceso padre (código+datos) lo cual no ocurre cuando se usan threads ya que un thread comparte el espacio de direcciones del proceso que lo crea. Luego la principal diferencia es el overhead.
- (20 ptos.) Considere los siguientes datos:

Proceso	Llegada	Tiempo de Servicio
A	0	4
B	2	6
C	4	3
D	10	6
E	13	6

Muestre como los algoritmos FCFS y RR( $q=1$ ). Realice una tabla con los valores de Tiempo de Finalización,  $T_{retorno}$  y  $T_{retorno}/T_{servicio}$ . Para cada algoritmo realice el gráfico correspondiente. Con todo lo realizado comente.

R.:

FCFS (9 ptos.)

```
      0..4....10..13....19.....25
A     XXXX
B       XXXXX
C        XXX
D         XXXXX
E          XXXXXX
```

	A	B	C	D	E	Media
Tfinalizacion	4	10	13	19	25	
Tretorno	4	8	9	9	12	8.4
Tr/Ts	1	1.3	3	1.5	2	1,76

RR (q=1) (9 ptos.)

```
      0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25
A     X X  X  X
B       X  X  X  X  X  X
C        X  X  X
D         X  X  X  X  X
E          X  X  X  X  X  X
```

	A	B	C	D	E	Media
Tfinalizacion	7	16	11	23	25	
Tretorno	7	14	7	13	12	10.6
Tr/Ts	1.75	2.33	2.33	2.16	2	2.11

comentario (2 ptos)

3. (a) (12 ptos.) Escriba un script usando bash que utilizando una función recursiva llamada **buscar** muestre todos los archivos \*.c que se encuentran a partir del directorio actual.

```
#!/usr/bin/bash

buscar () {
  for archivo in `ls`; do
    if [ -d "$archivo" ]; then
      cd "$archivo"
      buscar
      cd ..
    else
      if echo $archivo | grep -q *.c; then
        echo $PWD/"$archivo"
      fi
    fi
  done
}

buscar
```

- (b) (12 ptos.) El sgte código corresponde al proceso productor de una posible solución al problema del productor/consumidor. Es correcto dicho código? justifique.

```
semaforo s = 1;           // sección crítica
semaforo vacio = n;       // elementos disponibles en el buffer
semaforo lleno = 0;       // elementos ocupados en el buffer

while (true){
    wait(s);
    wait(vacio);
    producir();
    signal(lleno);
    signal(s);
}
```

R.: El código no es correcto.

El productor se bloqueará si ingresa en la sección crítica y no hay espacio en el buffer. Si el consumidor consume al menos un producto el productor se desbloquea pero esto no sucederá ya que el productor no ha salido de la sección crítica.

En este caso la solución es verificar si hay espacio en el buffer y luego ingresar en la sección crítica.