UNIVERSIDAD TECNOLÓGICA NACIONAL
FACULTAD REGIONAL MAR DEL PLATA
ARQUITECTURA DE SISTEMAS OPERATIVOS
1er Año – 1er Cuatrimestre

Trabajo Práctico:
Comunicación y Sincronización entre Procesos

1. ¿Qué se entiende por deadlock y livelock? De un ejemplo de cada uno fuera de los sistemas computacionales y analícelo.

2. En un sistema conviven 3 procesos y 2 recursos. Uno de los recursos (R2) es de uso exclusivo y el otro (R1) puede ser compartido por hasta dos procesos.
    a. ¿Puede haber deadlock o livelock?
    b. ¿Y si ahora R1 puede ser compartido por hasta 3 procesos?

3. Considere un sistema con 4 recursos del mismo tipo compartidos por 3 procesos. Cada recurso puede ser requerido a lo sumo por 2 procesos, y cada proceso puede requerir hasta 2 recursos. ¿En qué estado se encuentra el sistema? ¿Por qué?

4. En un sistema hay tres procesos (P1, P2 y P3) y tres recursos (R1, R2 y R3). Los tres recursos son de uso exclusivo. Se sabe que P1 requiere los tres recursos, P2 requiere de R1 y R2 y P3 solo requiere R3.
    a. ¿El sistema está libre de deadlock?
    b. ¿P3 influye en que el sistema esté o no libre deadlock?
    c. Si me aseguro que P2 no podrá pedir ningún recurso hasta que P1 haya liberado todos sus recursos. ¿El sistema está libre de deadlock? ¿Por qué?

5. Considere el siguiente programa. Note que el scheduler irá ejecutando estos procesos de manera concurrente, mezclando la ejecución de P1 y P2.

```
int x=10;
```

Proceso 1	Proceso 2

while (true) {
x--;
x++;
if (x != 10)
printf("x is %d",x);
}
	while (true) {
	x--;
	x++;
	if (x != 10)
	printf("x is %d",x);
	}

   a. Muestre una secuencia de ejecución (indicando el trace de programa/instrucción) en que imprima “x is 9”.
   b. Muestre una secuencia de ejecución (indicando el trace de programa/instrucción) en que imprima “x is 10”.

6. Considera que tiene un programa con dos procesos. Un proceso se encarga de imprimir y el otro se encarga de introducir ficheros para su impresión.

```
int peticiones = 0;
```

Proceso Impresora	Proceso Imprime Fichero

int R1;
While (true) {
If (peticiones > 0){
extraerColaPeticion();
R1 = peticiones;
R1 = R1 - 1;
peticiones = R1;
}
	int R2;
	While (true) {
	insertarFicheroCola();
	R2 = peticiones;
	R2 = R2 + 1;
	peticiones = R2;
	}

}	

Nota:

- La función extraerColaPeticion() extrae de la cola de peticiones un fichero y lo manda a imprimir. *(peticiones - 1).*
- La función insertarFicheroCola() inserta en la cola de peticiones un archivo para su impresión. *(peticiones + 1).*

a. ¿Presenta algún problema el código anterior si se ejecuta de manera concurrente? Analice el código.
b. Si la respuesta es negativa, explique el porqué. Si la respuesta es positiva, muestre la secuencia donde se genera el problema (Trace del programa).

7. Se tienen 3 procesos A, B y C. Implementar en máquina:
   a. El código con semáforos de manera tal que la secuencia sea ABC, ABC,ABC…
   b. ¿Y si quiero que la secuencia sea BCA, BCA, BCA…?

8. ¿Tiene algún problema el siguiente programa? ¿Por qué?

P1	**P2**

sem_wait(Sem1);
sem_wait(Sem2);
// Sección Crítica //
sem_post(Sem2);
sem_post(Sem1);
	sem_wait(Sem2);
	sem_wait(Sem1);
	// Sección Crítica //
	sem_post(Sem1);
	sem_post(Sem2);

9. Implementar en máquina la sincronización de los procesos A y B de tal manera que, siendo X una variable global:

A	B

x=199;
x=x+1;
Print(x);
	x=500;
	x=X/10;
	Print(x);

   a. Siempre el resultado de la ejecución sea 200 y 50.
   b. Siempre el resultado de la ejecución sea 50 y 200.

10. Dado el problema del Productor-Consumidor y los siguientes algoritmos:

sem_t espacioLibre, manipulaciónDelBuffer, elementoDisponible;	
Productor	**Consumidor**

While(true) {
producirUnElemento;
sem_wait(espacioLibre);
sem_wait(manipulaciónDelBuffer);
depositarElementoEnBuffer;
sem_post(manipulaciónDelBuffer);
sem_post(elementoDisponible);
}
	While(true) {
	sem_wait(manipulaciónDelBuffer);
	sem_wait(elementoDisponible);
	sacarElementoDelBuffer;
	sem_post(manipulaciónDelBuffer);
	sem_post(espacioLibre);
	consumirElemento;
	}

   a. Inicialice los semáforos según corresponda. ¿Es válida la solución anterior? ¿Por qué?