

## Practica 1.

### Objetivo:

Afirmar de forma práctica los conceptos aprendidos en clase sobre, problemas y soluciones de la asignación y manejo de recursos en un sistema operativo.

### Descripción:

se trata de una aplicacion que simula de forma determinística los tiempos, estados y peticiones de los procesos y los recursos de un sistema. Por medio de una serie de comandos entrada se determinará la forma en que los procesos van solicitando y soltando los recursos del sistema.

### Comandos de entrada:

Se tendrá la siguiente sintaxis, puede ser por medio de algún archivo de texto o por medio de una consola de comandos:

T#([procesos]) -> indica una unidad de tiempo identificada con un número en la que se ejecutarán asignaciones y/o liberaciones de recursos ejemplo:

P#([recursos] | [operación]) -> indica que un proceso realizara cierta operación con los recursos indicados, las operaciones pueden ser S solicitud, L liberación.

```
T1(P1(R1|S), P2(R2|S), P3(R2,R3,R4|S))
T2(P4(R1,R6|S))
T5(P1(R6|S))
T7(P3(R3|L))
T8(P3(f))
```

### Log:

Deben tener un log donde se pueda ver el estado de los procesos y recursos en cada tiempo T para el ejemplo anterior sería:

```
T1-----
Se asigno R1 a P1
P1 se está ejecutando
Se asigno R2 a P2
P2 se está ejecutando
Se asignó R3 y R4 a P3
P3 en espera hasta que se libere R2
Fin T1-----
```

```
T2 -----  
P1 se está ejecutando  
P2 se está ejecutando  
P3 en espera hasta que se libere R2  
se le asigna R6 a P4  
P4 en espera hasta que se libere R1  
Fin t2 -----
```

Deberán implementar la negación de espera circular y la técnica de detección de ciclos de espera.

Se debe tener una opción para activar que se realice la negación de espera circular, otra opción para activar la detección de ciclos de espera y una por default que dejara que los procesos entren en un interbloqueo.

**Notas:**

- No se van a calificar errores semánticos ni sintácticos en el archivo y/o comandos.
- Todos los recursos son inapropiativos y exclusivos, y todos los procesos contendrán a sus recursos cuando estén en espera.
- Ya que no se trata de una aplicación del curso de compiladores o lenguajes formales, el archivo de entrada o comandos quedan a discreción del estudiante, el anterior es un ejemplo de como se podría hacer, el objetivo es que se puedan ingresar en unidades de tiempo las acciones de los procesos.
- El lenguaje es a elección del estudiante.

**Entregables:**

- Código fuente y/o ejecutable con la solución al problema que se solicita.
- Fecha de entrega 10 de agosto.