



# Universidad Nacional Autónoma de México

## FACULTAD DE CIENCIAS

### Computación Concurrente

### **Práctica 03**

Implementación de Algoritmo de Peterson y Filtro modificado

### **Profesor:**

Salvador González Arellano

### **Integrantes:**

Contreras Ibarra Francisco

Marín Parra José Guadalupe de Jesús

Ortega González José Ethan

Ramírez Gallegos Leslie

Ramírez López Alvaro

## 1. Teoría:

1. Proponer 4 problemas donde se pueda utilizar el algoritmo de Peterson para su solución.

**Solución:**

- Gestionar el uso de una computadora para dos roomies.
- Controlar las horas de salida de dos personas.
- Administrar el uso de un hangar para dos avionetas en un pequeño aeropuerto.
- Dar prioridad a uno de los dos procesos para un recurso que gestiona las llegadas de entrada en una empresa.

2. Proponer 2 problemas donde se pueda utilizar el Algoritmo del Filtro

**Solución:**

- Dar prioridad al préstamo de libros a usuarios con suscripción en una librería.
- Priorizar  $n$  procesos sobre algunos recursos en un sistema operativo.

Responde las siguientes preguntas, justificando tu respuesta:

- ¿Los algoritmos cumplen con No Deadlock?

**Solución:** Sí cumplen porque los procesos primero ingresarán a la sección crítica con seguridad, por lo tanto, si un proceso se adelanta entonces éste ingresaría a la sección crítica.

- ¿El Algoritmo de Peterson cumple con la propiedad de Justicia?

**Solución:** Si cumple la propiedad de justicia, ya que en el algoritmo de Peterson cuando un proceso quiere entrar a su región crítica este mediante las banderas dirán que quiere entrar a su región para después ceden su turno y se quedará en una espera ocupada donde estará preguntando si es el turno del otro proceso y si el otro proceso quiere usar la región crítica y esto sucederá hasta que el otro proceso deje de usar la región crítica donde después de ocupar la región crítica se dirá mediante la bandera que no se quiere entrar a la región crítica por lo que el que el proceso que estaba en la espera ocupada saldrá de la espera ocupada para así entrar a su región crítica por lo que en algún momento cuando el proceso que esté en su región crítica salga entonces el otro podrá entrar a su región crítica y si un proceso está en su región no crítica esto no afectará al que quiera acceder a su región crítica ya que mediante la bandera del proceso que está en su región esta indicara que no quiere entrar a su región crítica por lo que el que quiere entrar a su región crítica saldrá de la espera ocupada para entrar a su región crítica.

- ¿Cuál de estos algoritmos cumple con la propiedad Libre de Hambruna?

**Solución:**

- El algoritmo de Peterson
- El algoritmo del Filtro

- ¿Cumplen con Exclusión Mutua?

**Solución:** Sí cumplen ya que se encargan de gestionar los procesos para que usen los recursos de manera organizada y con la finalidad que no se produzcan problemas.

## 2. Referencias:

- OS Paterson Solution - javatpoint. (s.f.). [www.javatpoint.com](http://www.javatpoint.com). Recuperado 23 de octubre de 2022, de <https://www.javatpoint.com/os-paterson-solution>
- (S/f). Utexas.edu. Recuperado el 23 de octubre de 2022, de <http://users.ece.utexas.edu/~garg/dist/jbkv2/Peterson-Proof.pdf>
- Exclusión mutua - Win32 apps. (2022, 24 septiembre). Microsoft Learn. Recuperado 23 de octubre de 2022, de <https://learn.microsoft.com/es-es/windows/win32/wmformat/mutual-exclusion>
- seguridad y vivacidad/viveza - PDF Free Download. (s. f.). Recuperado 29 de octubre de 2022, de <https://docplayer.es/203743104-Seguridad-y-vivacidad-viveza.html>
- Algoritmo de Peterson - frwiki.wiki. (s. f.). Recuperado 29 de octubre de 2022, de [https://es.frwiki.wiki/wiki/Algorithme\\_de\\_Peterson](https://es.frwiki.wiki/wiki/Algorithme_de_Peterson)