**UNIVERSIDAD DE GUADALAJARA**

Centro Universitario De Ciencias Exactas E Ingenierías

División de Tecnologías para la Integración Ciber-Humana

Seminario de Solución de Problemas de Sistemas Operativos I7030

Reporte de Actividad Práctica

**Actividad 7. Productor-Consumidor**

Alumnos:

Hernández Ledezma Brandon 215515031

Martínez Castillo Lisseth Abigail 218292645

Ramírez Plascencia Miguel Alejandro 215628413

Profesor: Quintanilla Moreno Francisco Javier

11 de septiembre de 2022

**Multiprogramación**

**Objetivo de la práctica**

El alumno identificará el problema del productor-consumidor, así como sus diferentes aplicaciones

**Antecedentes**

El problema de Productor-Consumidor es utilizado para la sincronización de multiproceso, pero para ello es importante tener en claro dos conceptos de antemano: Sincronización de procesos y Semáforo.

*Sincronización de procesos*

En los sistemas operativos existen muchos procesos presentes en un estado en específico. Al mismo tiempo contamos con recursos limitados, entonces estos recursos necesitan ser compartidos entre varios procesos, pero debemos asegurarnos de que dos procesos no estén usando los recursos al mismo tiempo porque esto podría generar inconsistencia en los datos, para ello los procesos deben sincronizarse.

Cuando los procesos comparten recursos se les llama procesos cooperativos y cuando no interfieren unos con otros se les llaman procesos independientes.

*Semáforo*

La sincronización entre procesos se puede alcanzar con la ayuda de un semáforo, una variable que indica el número de recursos que hay disponibles en el sistema en un tiempo en particular.

En general se representa con la letra “S”. Usa dos funciones **wait()** y **signal()**, ambas son usadas para cambiar el valor o estado del semáforo, pero éste puede ser cambiado solo por un proceso en un tiempo en específico y ningún otro proceso debería poder cambiarlo al mismo tiempo.

Existen dos tipos de semáforo:

* Binario: Su valor va a ser 0 o 1 dependiendo de la disponibilidad
* Contador: Su valor es inicializado y decrementa

*Productor-Consumidor*

Como su nombre lo indica, hay un productor, que produce algo y un consumidor que consume lo que el productor produjo. Los productores y consumidores comparten el mismo buffer de memoria de tamaño fijo.

El productor genera datos, los pone dentro del buffer y de nuevo comienza a generar datos. El trabajo del consumidor es consumir los datos del buffer.

Problemática:

* El productor debe producir datos mientras el buffer no esté lleno, si el buffer está lleno, el productor no puede hacer uso de la memoria.
* El consumidor solo puede consumir datos si el buffer no está vacío. Si el buffer está vacío el consumidor no debe poder tomar datos del buffer.
* El productor y consumidor no deben acceder al buffer al mismo tiempo

Solución:

La solución a la problemática se puede abordar con la ayuda de semáforos, que nos indiquen los diferentes estados:

* “S”: Usada para guardar la exclusión mutua entre procesos, en este estado no pueden acceder al buffer ni el productor ni el consumidor.
* “E”: Usada para definir el espacio vacío en el buffer, inicialmente se define al espacio entero vacío.
* “F”: Espacio que ha sido llenado por el productor, inicialmente es 0 porque no hay espacio que haya sido llenado por el productor

**Desarrollo**

Para el desarrollo de esta práctica se implementó una simulación donde se muestra cómo al iniciar la ejecución del programa los procesos se cargan en la memoria RAM, una vez cargados todos, el consumidor comienza a consumir los procesos cambiando el estatus del semáforo

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Interfaz de usuario gráfica, Gráfico

Descripción generada automáticamente

Imagen 2.1 y 2.2 Imágenes de la ejecución del programa, conforme se ejecutan los procesos se van liberando espacio en el buffer de la RAM.

Código

Texto

Descripción generada automáticamente

Imagen 2.3 Implementación del código de FCFS para encolar los procesos y el manejo del semáforo

Texto

Descripción generada automáticamente

Imagen 2.4 En estas funciones se encolan los procesos que utilizará el productor y después se definen las funciones que manejan la memoria RAM después de que el consumidor consume los procesos

**Conclusión:**

**Brandon Hernández Ledezma:**

La actividad de productor consumidor tuvo una complejidad un poco mayor a las anteriores y es que se tenían que mezclar los algoritmos que ya habíamos visto anteriormente con dos nuevos conceptos que son el "productor" y el "consumidor" personalmente cuando se tratan como entidades ajenas al sistema operativo es bastante confuso pero al final leyendo un poco de la teoría pude entender realmente el funcionamiento y las ventajas de usarlos, el proceso de crear la interfaz también fue un reto ya que se tenía que representar de manera gráfica la entrada en ram y al cpu mediante la representación de ambos con celdas y barras de carga.

**Lisseth Abigail Martínez Castillo:**

Entender el funcionamiento previo de los algoritmos de planificación y de los semáforos es importante pues de esta forma tenemos contexto de cómo funciona el algoritmo de productor consumidor. Es un algoritmo complejo pues implementa algoritmos previos. Comprender la teoría detrás de hilos también es necesario pues cada proceso hará uso de un hilo dedicado esto para evitar que haya problemas con el uso de los recursos del sistema.

**Miguel Alejandro Ramírez Plascencia:**

La creación del algoritmo resulto compleja ya que se necesitaban integrar lo que ya habíamos visto a manera de control dentro toda la estructura productor-consumidor, el manejar semáforos fue de vital importancia ya que con el podíamos regular la entrada a cpu desde ram pero sobre todo el poder mostrar su funcionamiento de manera visual sin duda ayuda mucho al entendimiento de que pasaba con el productor, el consumidor y los procesos.

**Fuentes:**

* (2019). *The producer-consumer problem in Operating System*  de AfterAcademy Sitio Web:[The Producer-Consumer problem in Operating System (afteracademy.com)](https://afteracademy.com/blog/the-producer-consumer-problem-in-operating-system)
* (2019).*What is Process Synchronization in Operating System?* de AfterAcademy Sitio Web: [What is Process Synchronization in Operating System? (afteracademy.com)](https://afteracademy.com/blog/what-is-process-synchronization-in-operating-system)
* (2019)*What is semaphore and what are its types?* de AfterAcademy Sitio Web: [What is semaphore and what are its types? (afteracademy.com)](https://afteracademy.com/blog/what-is-semaphore-and-what-are-its-types)