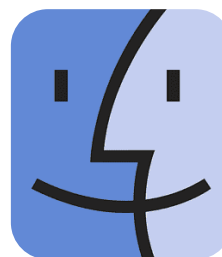
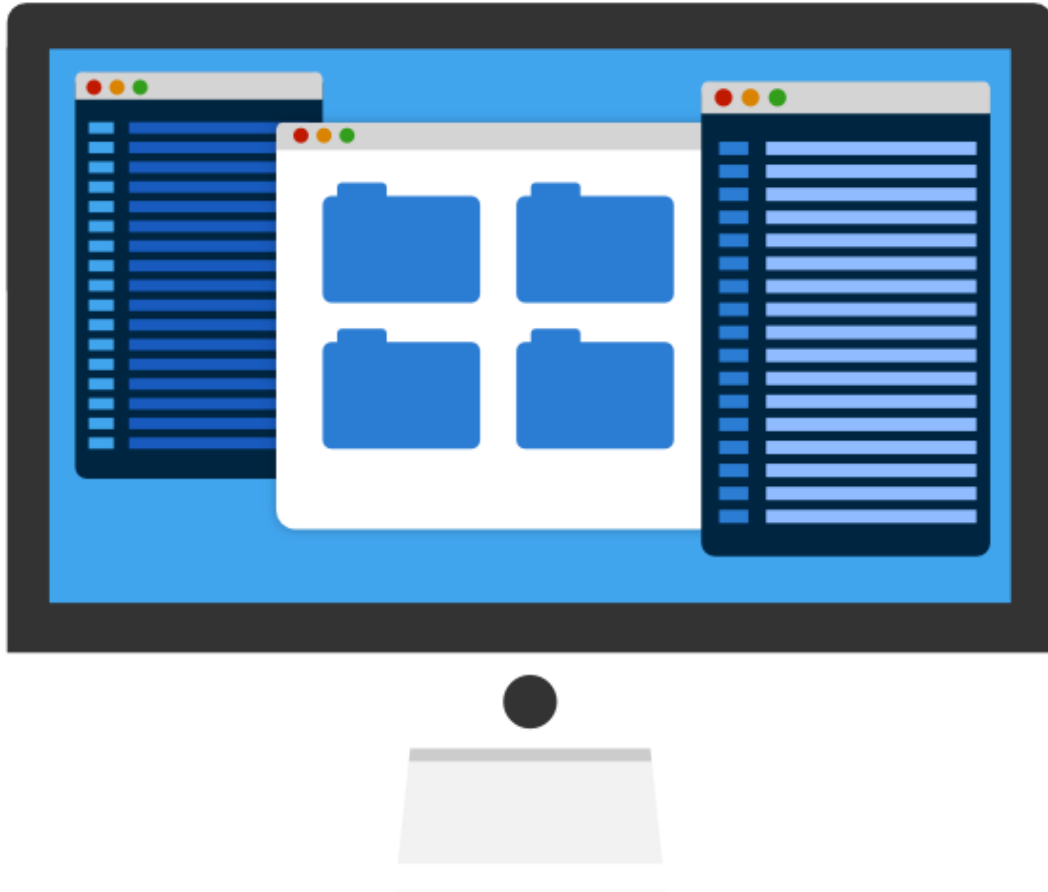




Actividad 10

Programa 5



Jessica Alexandra Magaña Salcedo

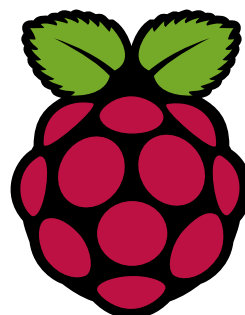
Sistemas Operativos

Maestra: Violeta Del Rocio Becerra Velazquez

Centro Universitario de Ciencias Exactas e
Ingenierías

2024A – D04

Fecha: 21/04/2024



Programa 5

Objetivo

El objetivo de esta actividad fue implementar una solución al problema clásico del productor-consumidor utilizando el lenguaje de programación Python. Se buscaba desarrollar un programa que simulara la producción y el consumo de elementos en un contenedor compartido por un productor y un consumidor, con el fin de comprender y aplicar conceptos de concurrencia y sincronización de hilos. La solución incluyó la creación de una interfaz gráfica para visualizar el estado del contenedor y el trabajo del productor y el consumidor en tiempo real.

El uso de Python se eligió por su sintaxis clara y legible, así como por su amplia gama de bibliotecas y herramientas disponibles para programación concurrente y desarrollo de interfaces gráficas de usuario (GUI). Se esperaba que esta actividad proporcionara una oportunidad para familiarizarse con el manejo de hilos en Python, así como para explorar la creación de aplicaciones con una representación visual que facilitara la comprensión del comportamiento del sistema.

La implementación de la solución implicó diseñar algoritmos para la producción y el consumo de elementos, utilizando tiempos de espera aleatorios para simular la aleatoriedad en las operaciones. Se utilizó un mutex para garantizar la exclusión mutua y evitar condiciones de carrera al acceder al contenedor compartido. Además, se creó una interfaz gráfica utilizando la biblioteca tkinter de Python, que mostraba el estado del contenedor y el trabajo del productor y el consumidor en tiempo real, mejorando la comprensión y la experiencia del usuario.

Reporte

La actividad consistió en diseñar e implementar una solución al problema clásico del productor-consumidor utilizando el lenguaje de programación Python. Este problema es un ejemplo común en el ámbito de la concurrencia y la sincronización de hilos, donde un productor agrega elementos a un contenedor compartido mientras que un consumidor los retira. El objetivo fue desarrollar un programa que simulara este escenario, prestando especial atención a la exclusión mutua y a la visualización del estado del sistema.

El lenguaje de programación Python se eligió debido a su popularidad, su sintaxis clara y legible, y su amplia disponibilidad de bibliotecas y herramientas para programación concurrente y desarrollo de interfaces gráficas de usuario (GUI). Python ofrece facilidades para trabajar con hilos y manejar la concurrencia de manera efectiva, lo que lo convierte en una elección adecuada para este tipo de problema.

La implementación de la solución se dividió en varias etapas. En primer lugar, se definieron las estructuras de datos necesarias, incluyendo el contenedor compartido y las variables de estado del productor y el consumidor. Se utilizó un mutex para garantizar la exclusión mutua al acceder al contenedor, evitando condiciones de carrera que podrían llevar a inconsistencias en los datos.

La siguiente etapa consistió en la implementación de las funciones del productor y el consumidor. Estas funciones se ejecutaron en hilos separados y simulaban la producción y el consumo de elementos en el contenedor. Se introdujeron tiempos de espera aleatorios para simular la variabilidad en las operaciones de producción y consumo, lo que añadió un grado de realismo al programa.

Una vez implementadas las funciones principales, se procedió a crear la interfaz gráfica utilizando la biblioteca tkinter de Python. La interfaz mostraba el estado del contenedor y el trabajo del productor y el consumidor en tiempo real. Se utilizaron etiquetas para representar visualmente el contenedor y las variables de estado del productor y el consumidor, proporcionando al usuario una representación clara del sistema en funcionamiento.

Finalmente, se agregaron mensajes que indicaban quién estaba trabajando en un momento dado, lo que mejoró la comprensión y la experiencia del usuario al interactuar con la aplicación. Estos mensajes proporcionaron retroalimentación en tiempo real sobre las actividades del productor y el consumidor, lo que facilitó el seguimiento del comportamiento del sistema.

Conclusión

La implementación de la solución al problema del productor-consumidor en Python fue un ejercicio valioso que permitió explorar conceptos fundamentales de concurrencia y sincronización de hilos. El uso de una interfaz gráfica proporcionó una representación visual que facilitó la comprensión del sistema, mientras que la exclusión mutua garantizó la integridad de los datos compartidos. La actividad proporcionó una oportunidad para familiarizarse con el manejo de hilos en Python y para desarrollar habilidades en programación concurrente y desarrollo de aplicaciones GUI.

Enlace para video

<https://drive.google.com/file/d/1wNIFtMfJNL1ceSszk3ijOlji9EfT3vbN/view?usp=sharing>