
Scheduling Ships

Calendarizador de Hilos

Fecha de asignación: 17, 18 de Setiembre, 2024
Grupos: 3-4 personas

Fecha de entrega: 10, 11 de Octubre, 2024
Profesor: Jason Leitón, Leonardo Araya

1. Objetivo

Crear una biblioteca de hilos, con el fin de ejecutar distintas tareas de manera simultánea y con ello calendarizarlas según se requiera.

2. Atributos a evaluar

- Trabajo individual y en equipo. Se requiere que funcione de manera efectiva como individuo y como miembro o líder en equipos diversos e inclusivos y en entornos multidisciplinarios, cara a cara, remotos y distribuidos.

3. Motivación

Los diferentes paralelismos que existen son campos de acción de un ingenier@ en computadores, es por esto que se debe de comprender los conceptos básicos relacionados a estos temas. Los procesos es el factor común de todos los tipos de paralelismo, por lo que la creación y manipulación de estos debe ser impecable por parte de un ingenier@ en computadores. Este proyecto busca implementar los hilos en espacio de usuario de tal manera que sea lo más eficiente posible de acuerdo con la funcionalidad solicitada. La abstracción de todo el ambiente que se debe de tener para controlar cualquier evento se vuelve esencial para un correcto funcionamiento del programa.

4. Descripción

4.1. CETHreads

Se deberá reimplementar la biblioteca Pthreads (utilizando las directivas de C, se aclara que un wrapper o interface no es reimplementación) con el nombre CEthreads, la cual incluye. al menos, las siguientes funciones:

- CETHread_create

- CThread_end
- CThread_join
- CMutex_init
- CMutex_destroy
- CMutex_unlock

4.2. Calendarizadores

Los calendarizadores se encargaran de ordenar la cola de “listos”, para que el hilo que se debe de ejecutar siempre esté de primero y los demás ordenados. El calendarizador deberá ser un parámetro.

Los algoritmos a implementar son:

- RR.
- Prioridad. Se necesitará parámetro extra para indicar la prioridad.
- SJF. Necesitará un parámetro extra para indicar el tiempo de cada barco.
- FCFS.
- Tiempo real. Necesitará el tiempo máximo que debe durar en pasar el canal.

4.3. El canal

El canal es el medio por donde viajan los barcos, para ir de un lado a otro, el cual es de ambos sentidos, pero solo puede transitar en un sentido a la vez de acuerdo con el algoritmo que se especifica más adelante.

En cada lado del canal se formará una fila de barcos que esperan su turno para pasar por el canal. Esta fila debe estar ordenada de acuerdo con el calendarizador elegido por el usuario. La cantidad de barcos a representar en cada lado del canal se debe ser especificada por el usuario. El canal tiene una longitud definida por el usuario, así como la velocidad de los barcos. La figura 1 muestra un ejemplo de lo que se espera tanto a nivel de hardware como de interfaz.

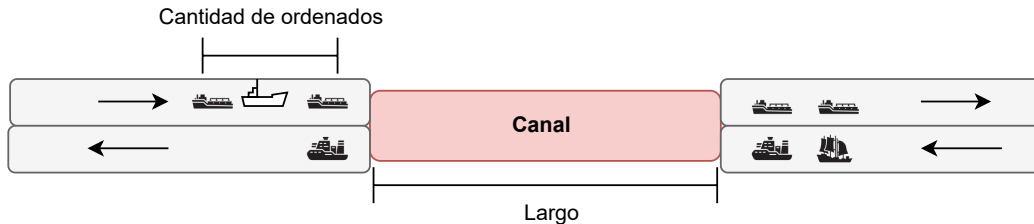


Figura 1: Ejemplo de Interfaz

El flujo de los barcos se controlará con alguno de los siguientes algoritmos:

- **Equidad:** consiste en establecer un parámetro W (indicado por el usuario) que indica cuantos barcos deben de pasar de cada lado, es decir, se inicia transportando W barcos de izquierda a derecha, después W barcos en sentido contrario. En caso de que alguno de los lados no haya embarcaciones se debe de garantizar el flujo de las mismas, en el lado en el que si hayan
- **Letrero:** consiste en un letrero (izquierda - derecha) que indica el sentido de vía del canal, cuando el letrero está en “izquierda” pasan los barcos del lado izquierdo, si estuviera en “derecha” entonces pasan los que están del lado derecho. El letrero cambia cada cierto tiempo, el cual es definido por el usuario al iniciar el programa.
- **Tico:** no hay control de flujo, sin embargo, no puede haber colisiones de barcos, también se debe de garantizar el flujo en caso de que hayan barcos solo de un lado.

En todos los casos las embarcaciones no pueden pasar, mientras otras estén en sentido contrario, con el fin de no ocasionar choques. Cabe destacar que por ningún motivo puede haber colisiones

Se debe crear un archivo de configuración para el canal, en el cual se especifique lo siguiente:

- Método de control de flujo.
- El largo del canal.
- La velocidad del barco.
- Cantidad de barcos ordenados en la cola de listos.
- El tiempo que el letrero cambie (en caso de que aplique).
- El parámetro W (en caso de que aplique).
- Cualquier otro que sea necesario pasar como parámetro.

4.4. Los barcos

Los barcos se generarán desde los extremos (océanos), y la función de todos es ir al otro océano pasando por el canal. Cada embarcación debe ser un hilo por lo que se generará un hilo nuevo cada vez que se cree un barco. Debe existir algún mecanismo sencillo, para indicar de cual océano se desea generar una embarcación.

Las embarcaciones se generan con velocidades ligeramente distintas (de acuerdo con el tipo).

Los barcos pueden ser de tres tipos:

- Normales: Sin tratamiento especial. (Es la más lenta)
- Pesqueras: Son más rápidas que las normales.
- Patrulla: Tienen tratamiento de urgencia por lo que son las más rápidas y se deben tratar como un sistemas de tiempo real *hard*.

La generación de barcos se puede hacer de dos tipos:

- Con carga definida: El usuario introduce previo a la ejecución un conjunto de barcos para cada lado del canal, indicando el tipo de barco.
- Por medio de tecla: Se inicia la ejecución y en cualquier momento se puede generar un nuevo barco de cualquier tipo en cualquiera de los lados.

4.5. Hardware

Se deberá construir un prototipo de hardware el cual muestre de alguna manera (creatividad de cada grupo) lo especificado anteriormente. Se debe mostrar la cola de barcos en ambos lados del canal, así como el letrero, y lo que sucede durante el transporte del canal. Se debe de mostrar el tipo de barco en cada caso. **Nota:** No se espera que haya movimiento de elementos físicos, pero que si se represente todos los elementos solicitados de manera codificada. Por ejemplo, el letrero puede ser representado por un led, en donde si está apagado el sentido del canal es de izquierda a derecha, en caso contrario, el sentido sería de derecha a izquierda.

4.6. Interfaz gráfica (Puntos extras)

Se deberá crear una interfaz gráfica que muestre lo de las secciones anteriores, se espera una pantalla similar a la que se observa en la figura 1. Cabe destacar que la interacción entre el programa y el usuario debe ser lo más sencillo posible. Además se calificará la fluidez de las animaciones, así como las restricciones de los movimiento. Cabe destacar que por ninguna circunstancia las embarcaciones pueden chocar y rebasar a otras, esto implica que dos barcos no pueden estar en el mismo punto a la misma vez. Todos los elementos deberán ser representados

gráficamente. El usuario puede cancelar el programa presionando la tecla w, se debe terminar de manera elegante.

5. Analogías con los recursos computacionales

- El canal juega el papel del CPU.
- Los barcos son los diferentes procesos (cada uno es un hilo).
- Las posiciones son recursos computacionales.
- Los calendarizadores son los algoritmos que se utilizan en un SO.
- Los algoritmos de flujo son análogos a las políticas de un SO.

5.1. Requerimientos técnicos

- La interfaz gráfica puede hacerse en cualquier lenguaje, el cual solo posea un **único hilo**.
- Este proyecto se debe realizar en el lenguaje de programación C. Pueden utilizar las bibliotecas que sean necesarias a excepción de la de hilos.
- Debe ser implementado en Linux y se debe proporcionar un makefile, que genere e instale lo necesario.
- No se permite soluciones “alambradas”.
- Se debe prestar especial atención a los errores de acceso a memoria o utilización de recursos. Es inaceptable el error *segmentation fault* o *Core dumped* se penaliza con 5 puntos cada error de este tipo.

6. Documentación- Estilo IEEE-Trans (máximo 5 páginas)

- Abstract: Síntesis del proyecto, Generalmente contiene entre 150-200 palabras, debe indicar en qué consiste el proyecto o tarea, cómo se realizó el proyecto o tarea y el principal hallazgo o conclusión.
- Introducción: Teoría necesaria, breve descripción del proyecto y qué es lo que se espera en el escrito.
- Ambiente de desarrollo: Se debe todo lo que se ocupa para ejecutar el proyecto.

- Atributos: Esta sección deben de describirse cuales atributos fueron reforzados durante el desarrollo del proyecto. Para el atributo de Trabajo individual y en equipo se debe especificar 7 puntos (Se debe colocar pregunta y respuesta), los cuales son los siguientes:
 - Indicar las estrategias para el trabajo individual y en equipo de forma equitativa e inclusiva en las etapas del proyecto (planificación, ejecución y evaluación).
 - Indicar la planificación del trabajo mediante la identificación de roles, metas y reglas.
 - Indicar cuales acciones promueven la colaboración entre los miembros del equipo durante el desarrollo del proyecto.
 - Indicar cómo se ejecutan las estrategias planificadas para el logro de los objetivos.
 - Indicar la evaluación para la el desempeño del trabajo individual y en equipo
 - Indicar la evaluación para las estrategias utilizadas de equidad e inclusión.
 - Indicar la evaluación para las acciones de colaboración entre los miembros del equipo
- Diseño, tanto del software como del hardware (en caso de que aplique): Diagramas UML, secuencia, arquitectura, imágenes, descripciones entre otros, todo lo que sea necesario para entender de una mejor manera el diseño y funcionamiento del proyecto. Se espera como mínimo el de arquitectura, componentes, UML y secuencia.
- Instrucciones de cómo se utiliza el proyecto.
- Conclusiones: se espera la generalización de los resultados, recordar que las conclusiones no son un resumen de lo que se realizó.
- Sugerencias y recomendaciones.
- Referencias

7. Entregables

- Código fuente con documentación interna.
- Documentación.
- Archivos necesarios para ejecutar el programa.

8. Evaluación

- Biblioteca de hilos 20 %
- Calendarizadores 15 %
- Integración 10 %
- Hardware 15 %
- Usuario y archivos 5 %
- Sincronización y Robustez 15 %
- Interfaz gráfica 10 %
- Documentación 20 %

9. Fecha de entrega

- 10,11 de Octubre. 23:55 por tecdigital.

10. Otros aspectos administrativos

- Para la revisión del proyecto se debe de entregar tanto la documentación como la implementación del software.
- No se reciben trabajos después de la hora indicada.
- En la revisión del proyecto pueden estar presentes el coordinador y asistente.
- Es responsabilidad del estudiante proveer los medios para poder revisar la funcionalidad del software, por ejemplo, si no se realiza la interfaz, se debe de proporcionar otro medio para la verificación, de lo contrario la nota será cero en los rubros correspondientes a la funcionalidad faltante.
- En caso de que alguna parte de la especificación esté confusa o ambigua, el grupo de trabajo puede agendar una cita con el profesor para aclarar cada aspecto.