

MLFQ (Multilevel FeedbackQueue) Scheduling

Opcional No. 1

Descripción general

A continuación se definen los lineamientos de un actividad práctica para los estudiantes del curso sistemas operativos, en particular, deben crear un video explicativo, implementar un simulador usando el paradigma Orientado Objetos y leer varios archivos de entrada.

Objetivos

Durante el desarrollo de de las actividades se logra:

- Evidencia los problemas de la ejecución concurrente y resume los posibles mecanismos que pueden ser empleados a nivel del sistema operativo para mitigar la aparición de los problemas evidenciados
- Compara y contrasta los algoritmos comúnmente usados para planificación expropiativa y no-expropiativa de procesos en sistemas operativos
- Identifica las maneras como la lógica alrededor de los algoritmos de planificación son aplicables a otros dominios, agendamiento en redes, planificación de proyectos y otros problemas más allá de la computación

Implementación

Realice la implementación del algoritmo MLFQ usando el paradigma orientado a objetos. Este algoritmo agrupa los procesos en diferentes colas donde cada proceso tiene un identificador, la cantidad de tiempo en CPU que requiere, el tiempo de llegada a la cola de procesos listos y otros atributos. El algoritmo se encarga de gestionar estas colas y despachar los procesos en el orden adecuado, respetando la prioridad de cada cola y utilizando diferentes políticas de planificación para su ejecución.

En la implementación se deben considerar tres colas de los siguientes esquemas

- RR(1), RR(3), RR(4), SJF.
- RR(2), RR(3), RR(4), STCF.
- RR(3), RR(5), RR(6), RR(20).

Entradas y Salidas

Las entradas consisten en archivos de texto que contienen una lista de procesos, donde cada línea representa un proceso con la siguiente estructura:

```
# Archivo: mlq001.txt
# etiqueta; burst time (BT); arrival time (AT);Queue (Q);Priority(5>1)
A;6; 0; 1; 5
B;9; 0; 1; 4
C;10; 0; 2; 3
D;15; 0; 2; 3
E;8;0; 3; 2
```

Cada campo corresponde a:

- **Etiqueta:** Identificación del proceso (p1, p2, etc.).
- **Burst Time (BT):** Tiempo total que el proceso necesita para ejecutarse.
- **Arrival Time (AT):** Momento en el que el proceso llega al sistema.
- **Queue (Q):** Cola a la que pertenece el proceso (nivel de prioridad en el algoritmo MLQ).
- **Prioridad:** Prioridad del proceso dentro de su cola, siendo 5 la más alta y 1 la más baja.

Las **salidas** deben ser generadas en un archivo de texto donde se detalle el orden de ejecución de los procesos, con la siguiente estructura:

```
# archivo: mlq001.txt
# etiqueta; BT; AT; Q; Pr; WT; CT; RT; TAT
A;6; 0; 1; 5; 3; 9; 0; 9
B;9; 0; 1; 4; 6; 15; 3; 15
C;10; 0; 2; 3; 20; 30; 15; 30
D;15; 0; 2; 3; 25; 40; 20; 40
E;8;0; 3; 2; 40; 48; 40; 48

WT=23.5; CT=35.5; RT=19.5; TAT=35.5;
```

Este archivo mostrará el tiempo de espera (WT), tiempo de completado (CT), tiempo de respuesta (RT) y *TurnAround Time* (TAT). Adicional, los valores promedio de estas métricas.

Entrega: Video Explicativo + Informe

4. Debe crear un informe en PDF donde presente una breve descripción del funcionamiento del algoritmo MLQ.
 - a. Cree dos entradas y presente el resultado que se espera de su implementación
 - b. Considere los siguientes archivos (<https://drive.google.com/drive/folders/1-26Y7gvqbwSzjsPBJ24vfsvA5B09FMY?usp=sharing>)
5. Debe subir el código fuente a través de un repositorio como Github con acceso público.
6. Grabe un video de máximo 5 minutos en el que expliquen su implementación.
 - a. El video debe incluir una explicación del código, con énfasis en la lógica utilizada para la planificación de las colas y la asignación de procesos.
 - b. El video debe incluir la ejecución del código con al menos tres de los archivos de entrada y análisis de los resultados obtenidos en el archivo de salida.

Criterios de Evaluación:

- Implementación del Algoritmo: 40%
- Calidad del Código (estructura, comentarios, buenas prácticas): 20%
- Formato y Estructura de Archivos: 10%
- Calidad y Claridad del Video: 30%

Letra Chica

- La entrega debe realizarse en las fechas establecidas a través del campus virtual.
- Tenga en cuenta los criterios de evaluación
- Puede realizar la implementación utilizando lenguajes de programación que permitan el paradigma Orientado a Objetos