



Universidad Nacional Autónoma de México



Facultad de Ingeniería

Semestre 2025-1

Sistemas Operativos

Tarea 2 - Comparación de planificadores

Profesor: Gunnar Eyal Wolf Iszaevich

Alumno: León Pérez Aarón Rodrigo

Fecha de entrega: 19/Noviembre/2024

Descripción del Problema

El problema consiste en comparar diferentes algoritmos de planificación de procesos en un sistema operativo. Cada algoritmo organiza cómo los procesos reciben tiempo de CPU, afectando métricas clave como tiempo de turnaround, tiempo de espera y proporción de penalización. Este problema tiene aplicaciones en sistemas reales, donde una planificación eficiente puede maximizar el rendimiento y reducir los tiempos de espera de los usuarios.

El programa debe:

- Generar múltiples procesos con tiempos de llegada y duraciones aleatorias.
- Simular los algoritmos de planificación: FCFS, Round Robin (RR) y Shortest Process Next (SPN).
- Evaluar las métricas clave de cada algoritmo.
- Mostrar visualizaciones que indiquen cómo cada algoritmo asigna el tiempo de CPU a los procesos.

Algoritmos Implementados

1. FCFS (First Come, First Served):
 - Planifica los procesos en el orden en que llegan.
 - Métrica clave: simplicidad en la implementación.
 - Problema: Puede sufrir de inanición para procesos con tiempos largos.
2. Round Robin (RR):
 - Asigna tiempos de CPU en ciclos fijos (quantum).
 - Los procesos que no terminan en un ciclo regresan a la cola.
 - Beneficio: Justo y equitativo para todos los procesos.
 - Implementación: Se incluye RR1 (quantum=1) y RR4 (quantum=4).
3. SPN (Shortest Process Next):

- Prioriza los procesos con menor tiempo de ejecución.
- Beneficio: Minimiza el tiempo promedio de espera.
- Problema: Puede causar inanición para procesos más largos.

Métricas Calculadas

- **Tiempo de Turnaround (T):**
 - Tiempo total que un proceso toma desde su llegada hasta su finalización.
 - Fórmula: $T = \text{tiempo_final} - \text{tiempo_llegada}$.
- **Tiempo de Espera (E):**
 - Tiempo que un proceso pasa esperando para ser ejecutado.
 - Fórmula: $E = \text{tiempo_turnaround} - \text{duracion_proceso}$.
- **Proporción de Penalización (P):**
 - Relación entre el tiempo de turnaround y la duración del proceso.
 - Fórmula: $P = \text{tiempo_turnaround} / \text{duracion_proceso}$.

Estructura del Código

1. **Clases y Funciones Principales:**
 - Proceso: Clase que representa a un proceso con atributos como nombre, llegada, duración, y tiempo restante.
 - generar_procesos(): Genera procesos con tiempos aleatorios.
 - Algoritmos (fcfs, rr, spn): Simulan cada estrategia de planificación.
 - calcular_metricas(): Calcula métricas para evaluar cada algoritmo.
 - mostrar_esquema_visual(): Muestra la asignación de CPU a los procesos.
2. **Flujo del Programa:**
 - **Paso 1:** Genera procesos.

- **Paso 2:** Ejecuta cada algoritmo.
- **Paso 3:** Calcula y muestra métricas.
- **Paso 4:** Repite para 5 rondas.

3. Parámetros Configurables:

- Número de procesos.
- Tiempo máximo de llegada y duración.
- Quantum para Round Robin.

Ejemplo de Salida

Ronda 1: A: llegada=0, duración=3 B: llegada=1, duración=4 C: llegada=5, duración=7 D: llegada=7, duración=2 E: llegada=11, duración=3	Ronda 5: A: llegada=0, duración=4 B: llegada=2, duración=2 C: llegada=3, duración=7 D: llegada=6, duración=7 E: llegada=10, duración=6
FCFS: AAABBBBCCCCCDEEE T=7.00, E=3.20, P=3.82	FCFS: AAAABBBCCCCCDDDDDEEEEEE T=9.60, E=4.40, P=2.61
RR1: AABABBCBCDCDCECECEC T=7.40, E=3.60, P=3.61	RR1: AAABACBCDCDCEDCEDCEDCEDE T=12.60, E=7.40, P=3.13
RR4: T=0.00, E=-3.80, P=1.63	RR4: T=0.00, E=-5.20, P=0.79
SPN: AAABBBBDCCCCCCEEE T=6.00, E=2.20, P=3.67	SPN: AAAABBBCCCCCCEEEEEEDDDDDDD T=9.40, E=4.20, P=2.58

Conclusión

Después de realizar la práctica, comprendí cómo los diferentes algoritmos de planificación de procesos afectan el desempeño de un sistema operativo. FCFS es simple y directo, pero puede provocar largos tiempos de espera para algunos procesos. RR demostró ser más equitativo al distribuir el tiempo de CPU, aunque depende mucho del quantum elegido. SPN optimiza el tiempo promedio, pero puede dejar a los procesos largos esperando indefinidamente. Al implementar y comparar estos algoritmos, noté cómo pequeñas decisiones en la planificación pueden tener un impacto significativo en las métricas de desempeño. Sin duda, fue una experiencia enriquecedora que me hizo reflexionar sobre la importancia de la planificación adecuada en sistemas reales.