



UNIVERSIDAD NACIONAL AUTÓNOMA DE
MÉXICO

FACULTAD DE INGENIERÍA



Tarea 2

Comparación de planificadores

Integrantes:

Gómez Guzmán Anikey Andrea
Leon Gallardo Ian Yael

Materia

Sistemas Operativos

Grupo

6

Profesor

Gunnar Eyal Wolf Iszaevich

Semestre

2025-1

Fecha de entrega

7 de noviembre de 2024

Introducción

La intención de este código es simular una comparación entre planificadores de procesos, en este se incluye el , First-Served (FCFS), Round Robin (RR), Shortest Process Next (SPN) y Feedback (FB); estas comparaciones son evaluados con base a unos parámetros de rendimiento como el tiempo de espera y de retorno. Esto se hace por medio de rondas de rondas en las que se generan simulaciones aleatorias y los resultados se tabulan para compararlos.

Hilo de ejecución

1. Cuando se ejecuta el programa, se generan 5 rondas (elegidas por nosotros) de simulación
2. En cada una de las rondas generadas:
 - Se generan los 5 procesos con los tiempos de llegada y la duración, esta es elegida de forma aleatoria
 - Se ejecutan los algoritmos programados: FCFS, RR (quantum 1 y 4), SPN y FB.
 - Los cronogramas y métricas que salen como resultado se tabulan.
3. En cada tabla se plasman las métricas importantes para la comparación; es decir, el tiempo de retorno promedio, el tipo de espera promedio y la penalización promedio.

Descripción de las funciones

1. Función: `generaProceso(no_procesos=5, max_arrival=10, max_duration=10)`

Descripción: Genera procesos con tiempos de llegada y duración aleatorias; este retorna una lista con el nombre, llegada y duración, estos salen ordenados por tiempo de llegada

Parámetros:

- `no_procesos`: variable de tipo entero que genera los procesos, en nuestro programa usamos 5
- `max_arrival`: variable que tiene el tope de tiempo máximo de llegada de procesos
- `max_duration`: variable que da la duración máxima del proceso

2. Función: `fcfs(procesos)`

Descripción: esta función tiene la implementación de algoritmo First-Come, First-Served (FCFS) y ejecuta los procesos conforme van llegando. Retorna dos listas una con el cronograma de ejecución (`schedule (list)`) y otra con los tiempos de finalización (`compl_times (dict)`)

Parámetros:

- `procesos`: es una variable de tipo lista que tiene los procesos de ejecución

3. Función: rr(procesos, quantum)

Descripción: esta función implementa el algoritmo Round Robin (RR), es decir que asigna un quantum de tipo fijo para la ejecución de procesos. Esta función retorna una lista con el cronograma de ejecución schedule (list) y un diccionario con los tiempos de finalización compl_times (dict).

Parámetros:

- procesos: es una variable de tipo lista que tiene los procesos de ejecución
- quantum: variable de tipo entero con el tiempo de ejecución asignado por turno

4. Función: spn(procesos)

Descripción: esta función implementa el algoritmo Shortest Process Next (SPN), este selecciona el proceso más corto disponible de cada uno de los momentos. Esta retorna el cronograma de procesos generados (schedule (list)) y los tiempos de finalización de cada proceso (compl_times (dict)).

Parámetros:

- procesos: una variable de tipo lista con los procesos generados.

5. Función: fb(procesos, num_queues=4)

Descripción: implementa el algoritmo Feedback (FB), este utiliza varias colas con prioridades diferentes. Esta función retorna un cronograma de ejecución y los tiempos de finalización de cada proceso. En este caso nosotros escogimos 4 como el número de colas de prioridad.

Parámetros:

- procesos: variable de tipo lista con los procesos generados.
- num_queues: variable de tipo lista con la cola de prioridades.

6. Funcion: calculate_metrics(procesos, compl_times)

Descripción: esta función calcula las métricas de rendimiento (tiempo de retorno promedio, tiempo de espera promedio y penalización promedio) para que los algoritmos puedan realizar la planificación y eso mismo es lo que retorna en forma de diccionario.

Parámetros:

- procesos: variable de tipo lista con los procesos generados
- compl_times: variable de tipo diccionario con los tiempos de finalización.

7. Funcion: create_results_table(round_num, procesos, results)

Parámetros:

- procesos: variable de tipo lista que tiene el número de rondas de simulación
- round_num: variable de tipo lista que tiene una lista de procesos generados
- results: variable de tipo diccionario que tiene los resultados de los algoritmos

8. Función: run()

Descripción: es la función principal que ejecuta las demás funciones y por ende los algoritmos de planificación.

Ejemplo de ejecución

*Se generan 5 rondas, pero para explicarla solo se mostrara la primera

1. Procesos generados

Aqui se representan los procesos que han sido creados, donde cada proceso tiene:

- **Proceso:** Nombre del proceso
- **Llegada:** Tiempo de llegada del proceso al sistema.
- **Duración:** Tiempo que el proceso necesita para ejecutarse.

===== Ronda 1 =====		
Procesos generados:		
Proceso	Llegada	Duración
E	5	2
D	7	7
C	9	2
A	10	1
B	12	3

2. Resultados de los algoritmos

Esta tabla muestra cómo diferentes algoritmos de planificación gestionan los procesos. Cada algoritmo tiene los siguientes indicadores:

- **T (Tiempo promedio):** Tiempo promedio que los procesos pasan en el sistema (desde la llegada hasta la finalización).
- **E (Tiempo de espera):** Tiempo promedio que los procesos pasan esperando en la cola antes de ejecutarse.
- **P (Penalización):** Relación entre el tiempo total en el sistema y el tiempo de ejecución; mide qué tan bien se respetan los tiempos de ejecución ideales.
- **Cronograma:** Representación visual del orden de ejecución de los procesos en el tiempo.

Esto con el propósito de que los procesos se ejecutan en el orden en que llegan, y este es el tiempo acumulado que tardan los procesos.

Resultados de los algoritmos:				
Algoritmo	T (Tiempo promedio)	E (Tiempo espera)	P (Penalización)	Cronograma
FCFS	6.2	3.2	3.03	_____EEDDDDDDDCCABBB
RR1	5.6	2.6	1.84	_____EEDDCDACDBDBDBD
RR4	5.6	2.6	2.22	_____EEDDDDDCCADDDBBB
SPN	6	3	2.73	_____EEDDDDDDDACCBBB
FB	5.2	2.2	1.74	_____EEDDDCABCBDDDD

3. Estadísticas comparativas

Esta tabla resume los resultados promedio de todos los algoritmos:

- **T promedio:** Tiempo promedio acumulado en el sistema.
- **E promedio:** Tiempo promedio de espera.
- **P promedio:** Penalización promedio.

Estadísticas comparativas:		
T promedio	E promedio	P promedio
5.72	2.72	2.312
===== Ronda 2 =====		