

Comparación de Planificadores

1. Objetivo

El objetivo de esta tarea es comparar el comportamiento y rendimiento de distintos algoritmos de planificación de procesos. Para ello, se implementaron los siguientes planificadores: FCFS, RR, SPN y FB. El programa ejecuta cinco rondas de simulación midiendo los tiempos de respuesta, espera y penalización.

2. Descripción general del programa

El programa está escrito en Python 3 y se compone de las siguientes partes:

Componente	Descripción
class Proceso	Representa un proceso con atributos como tiempo de llegada, ejecución, inicio, fin y métricas.
FCFS()	Planificación por orden de llegada.
RR()	Simula Round Robin con quantum configurable.
SPN()	Selecciona el proceso con menor tiempo de ejecución disponible.
FB()	Planificación por colas multinivel con retroalimentación.
imprimir_resultados()	Muestra los resultados y calcula T, E y P.
asignar_cargas_aleatorias()	Genera cargas aleatorias por ronda.
main()	Controla las 5 rondas y muestra los resultados.

Además, se emplean **colores ANSI** para visualizar gráficamente las ejecuciones, de modo que cada proceso se distinga fácilmente.

3. Fórmulas de las métricas

Para cada proceso se calculan las siguientes métricas:

$T = t_{fin} - t_{llegada}$ (Turnaround)

$E = T - t_{ejecución}$ (Tiempo de espera)

$P = T / t_{ejecución}$ (Proporción de penalización)

Los promedios se obtienen dividiendo entre el número total de procesos.

4. Ejemplo de ejecución

```
Ronda numero: 1
  Cargas asignadas a los procesos:
  A: 0, t=3 | B: 1, t=5 | C: 3, t=2 | D: 9, t=5 | E: 12, t=5 | (total:20)

  FCFS: AAABBBBBCCDDDDDEEEEE
  T=6.2, E=2.2, P=1.74

  RR1: AABABCBBCBDDDEDEEEE
  T=6.8, E=2.8, P=1.77

  RR4: AAABBBBBCCDDDDDEEEEE
  T=6.4, E=2.4, P=1.72

  SPN: AAACBBBBBDDDDDEEEEE
  T=5.6, E=1.6, P=1.32

  FB: ABAACBBCBDDDEEEDDEE
  T=7.0, E=3.0, P=1.81
```

Las cargas se generan aleatoriamente. Cada ronda muestra los nuevos tiempos de llegada y ejecución, así como los resultados de todos los algoritmos.

Esto permite comparar tendencias y observar cómo varía el rendimiento dependiendo de la carga.

5. Conclusión

Los algoritmos muestran comportamientos distintos según la carga. FCFS es simple pero injusto; RR mejora la equidad; SPN optimiza el tiempo de espera, y FB logra un equilibrio adaptativo. El programa cumple con los criterios de la tarea: visualización, métricas, cargas aleatorias y colas multinivel.