

Simulador de planificación de procesos

Jorge Steven Reyes 1667619-3743
Universidad del Valle Sede Tulua
reyes.jorge@correounivalle.edu.co

Katherin Paz 1759649-3743
Universidad del Valle Sede Tulua
paz.katherin@correounivalle.edu.co

Ronald Esteban Mejía 1958725-2711
Universidad del Valle Sede Tulua
Mejia.ronald@correounivalle.edu.co

Resumen

El presente informe tiene como propósito dar a mostrar los conceptos aprendidos sobre planificación de procesos en la asignatura de Sistemas Operativos, aplicándolos en la elaboración de un programa que simula dicha planificación teniendo en cuenta los siguientes aspectos:

Se generan al azar los datos necesarios para la simulación.

Se le permite al usuario digitar desde el teclado los datos necesarios para la simulación.

Se implementaron dos colas donde la primera (de mayor prioridad) tiene una política apropiativa (RR), la segunda (de menor prioridad) tiene una política no apropiativa con tres algoritmos que el usuario puede escoger (FCFS, SJN, HRN).

Se implementaron criterios de penalización en la primera cola: El proceso sera penalizado cuando acabe su quantum o ejecute toda su ráfaga de cpu. Se implementaron criterios de recompensa en la segunda cola: El proceso sera recompensado cuando ejecute toda su ráfaga.

Se presentan los resultados (Tiempo de espera, tiempo de servicio, instante de tiempo final, índice de servicio, así como los tiempo promedio de espera y servicio.) de forma tabulada.

Este proyecto se realizo utilizando el lenguaje de programación Java y a Netbeans como IDE, a esto se deben ciertas características en la organización de los archivos como se muestra en la figura 1.

Se hizo uso de la librería Jfreechart que permite la visualización de distintos tipos de gráficos, este caso el diagrama de Gantt.

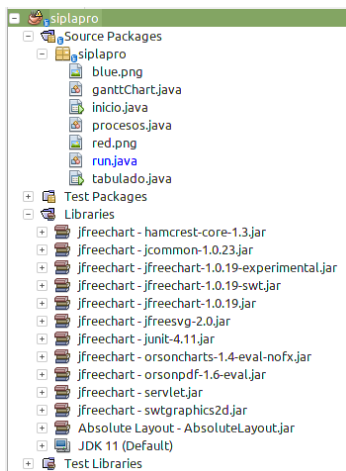


FIGURA 1: Archivos del proyecto

Funcionamiento

El archivo "inicio" es un JFrame que sirve como interfaz para recoger los datos necesarios del usuario, tales como: Nombre del proceso, tiempo de llegada, ráfagas de cpu, ráfagas de E/S, quantum y el algoritmo que funcionara en la segunda cola; Se va llenando la tabla con los datos de cada proceso agregado. Cuando el usuario inserte los procesos deseados y escoja el quantum podrá empezar la simulación, internamente se crea un arreglo de procesos y se ejecuta la función "Planner" de la clase "Run" con el array de procesos y el dato de quantum. (ver figura 2)

La clase "procesos" es la plantilla para la instanciación de los objetos procesos con sus debidas características. (ver en Figura 3)

La clase Run contiene las funciones necesarias para la planificación. La función principal "Planner" da valores a todas las variables y ejecuta ejecuta un ciclo while en el que cada iteración es un instante de tiempo, en cada instante de tiempo interactúan diferentes funciones auxiliares las cuales encolan los procesos que van llegando, encolan los procesos después de pasar por e/s, entre otras funciones. En el mismo while dependiendo del estado de las colas y del tipo de algoritmo que escogió el usuario, se ejecutan los procesos de estas colas ya sea en la cola1 con algoritmo RR o en la cola2 con algoritmos FCFS o SJN o HRN, al entrar en estas colas se van ejecutando los proceso (restándole unidades de ráfaga de cpu) y ejecutando los criterios de penalización y recompensa.

FIGURA 2: Inicio

```
public class procesos {
    public String Name, pCola;
    public int Tlleg, Cpu1, Es, Cpu2, pEs, Tfinal, Num;
    public float Tser, Tesp, IndSer;

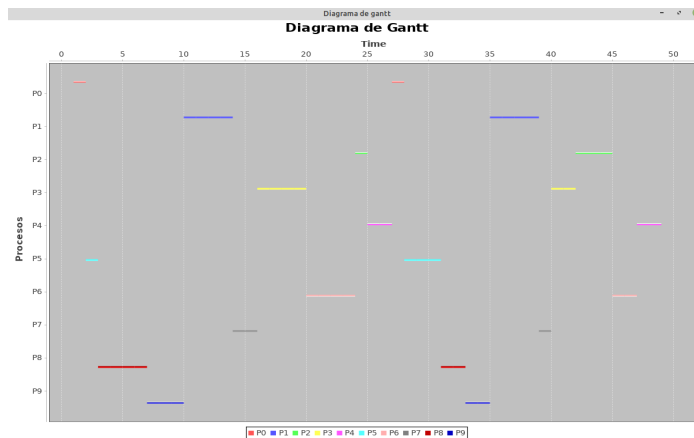
    procesos(String name, int tlleg,int cpu1,int es, int cpu2){
        Name = name;
        Tlleg = tlleg;
        Cpu1 = cpu1;
        Es = es;
        pEs = 0;
        Cpu2 = cpu2;
        pCola = null;
        Tfinal = -1;
        Tser = -1;
        Tesp = -1;
        IndSer = -1;
    }
}
```

FIGURA 3: Procesos

Cada que se ejecuta un proceso, su nombre es almacenado en un array que servirá como historial y mas adelante se utilizara para crear la gráfica de Gantt.

Se instanciara la ventana de tabulado y ejecutando una función de dicha clase para cargar la información necesaria para generar el tabulado. (Ver en figura 4)

Al final este imprime en consola lo que paso en cada instante de tiempo (Proceso, cpu, estado de las colas, si un proceso se encolo después de e/s). El tabulado se generara en el JFrame mostrando los resultados, este JFrame tiene un botón que ejecuta la función de la clase "ganttchart" y abre una ventana mostrando dicho gráfico. (ver en Figura 5)

FIGURA 4: *Tabulado*FIGURA 5: *Diagrama de Gantt*

Prueba y resultados

1 Prueba

1.1 *Se ingresaron 10 procesos random. (Figura 6)*

1.2 *Resultados. (Figura 7)*

1.3 *Historial en consola*

Debido a que el termino de los procesos es en el instante 49 solo se mostrara el estado en el tiempo inicial (0, 1) y en el instante final (49).)

| Proceso | T. Llegada | Refaga CPU | E/S | Refaga CPU |
|---------|------------|------------|-----|------------|
| P0 | 3 | 2 | 3 | 4 |
| P1 | 3 | 3 | 2 | 2 |
| P2 | 4 | 2 | 1 | 1 |
| P3 | 3 | 1 | 3 | 1 |
| P4 | 1 | 3 | 1 | 1 |
| P5 | 1 | 4 | 1 | 2 |
| P6 | 1 | 1 | 4 | 3 |
| P7 | 1 | 2 | 1 | 3 |
| P8 | 2 | 3 | 4 | 4 |
| P9 | 2 | 3 | 4 | 3 |

Cola 1: RR quantum: 1 Penalizacion cola 1: acabar quantum o acabar rafaga.

Cola 2: FCFS Recompensa cola 2: acabar rafaga.

Ejecutar

FIGURA 6: Datos

| Proceso | T. Llegada | Refaga CPU | E/S | Refaga CPU | T. Final | T. Servicio | T. Espera | Indice de Servi... |
|---------|------------|------------|-----|------------|----------|-------------|-----------|--------------------|
| P0 | 3 | 2 | 3 | 4 | 46 | 40.0 | 34.0 | 0.15 |
| P1 | 3 | 3 | 2 | 2 | 49 | 44.0 | 39.0 | 0.11363637 |
| P2 | 4 | 2 | 1 | 1 | 37 | 34.0 | 31.0 | 0.0882353 |
| P3 | 3 | 1 | 3 | 1 | 35 | 33.0 | 31.0 | 0.060606062 |
| P4 | 1 | 3 | 1 | 1 | 17 | 13.0 | 9.0 | 0.30769232 |
| P5 | 1 | 4 | 1 | 2 | 38 | 32.0 | 26.0 | 0.1875 |
| P6 | 1 | 1 | 4 | 3 | 29 | 25.0 | 21.0 | 0.16 |
| P7 | 1 | 2 | 1 | 3 | 40 | 35.0 | 30.0 | 0.14285715 |
| P8 | 2 | 3 | 4 | 4 | 43 | 36.0 | 29.0 | 0.19444445 |
| P9 | 2 | 3 | 4 | 3 | 48 | 42.0 | 36.0 | 0.14285715 |

Tiempo medio de servicio: 33.4 Tiempo medio de espera: 28.6 Indice de servicio: 0.15478288

FIGURA 7: Resultados tabulado

Tiempo: 0

Estado Cola1:

Estado Cola2:

Tiempo: 1

RR

Cola1: P4. cpu1: 2

Estado Cola1:

P5

P6

P7

Estado Cola2:

P4

...

...

Tiempo: 48

FCFS

Cola2: P1. cpu: 0

Estado Cola1:

Estado Cola2:

El proceso P0, termino en: 46

El proceso P1, termino en: 49

El proceso P2, termino en: 37

El proceso P3, termino en: 35

El proceso P4, termino en: 17

El proceso P5, termino en: 38

El proceso P6, termino en: 29

El proceso P7, termino en: 40

El proceso P8, termino en: 43

El proceso P9, termino en: 48

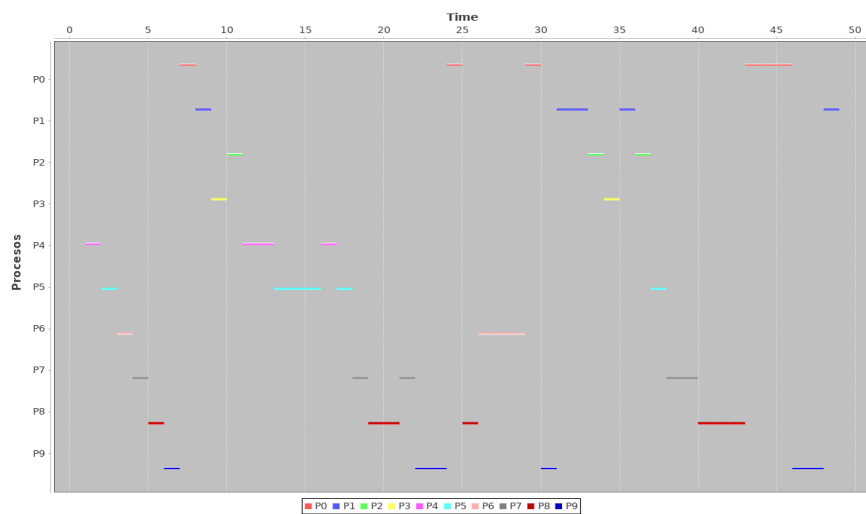


FIGURA 8: Diagrama de Gantt