

INFORME SOBRE USO Y PRUEBAS DE LA APLICACIÓN
REALIZADA PARA EL PROYECTO:
Algoritmo de Planificación de Colas Múltiples con Retroalimentación.

-Jonatan Obando Saa - 1664001
-Jacobó Hernández Echeverry - 1759585
-Luis Eduardo Henao Padilla - 1667483

Requisitos e Información para usar el programa:

- 1) Para ejecutar el archivo que se encuentra en la carpeta aplicación es requerido tener instalado Java SE Platform (cualquier versión).
- 2) Si desea acceder al código del programa, este se encuentra en la carpeta: PlanificacionProcesos > src > planificacionprocesos, y ahí encontrará los archivos .java; para la creación de esta aplicación usamos el IDE netBeans 8.2, pero el código se puede visualizar con otras aplicaciones.

Aclaraciones:

Cuando ejecutas la aplicación (ejecutable.jar), se muestra la siguiente ventana:

Simulador de Algoritmo de Planificación de Colas Múltiples con Retroalimentación

1) Para la siguiente simulación puedes ingresar los siguientes datos (**Opción b.**) o generarlos de forma aleatoria (**Opción a.**):

a) Si desea generar la información de forma aleatoria tenga en cuenta que el número de procesos será 10 en cualquier caso.

b) *Número de Procesos:

*Información del proceso: (digite los datos de un proceso, ingresalos y repite hasta haber registrado los de tu caso).

Instante de Llegada: Rafaga de CPU Inicial:

Rafaga de E/S: Rafaga de CPU Final: Prioridad:

2) Nuestra aplicación tiene 2 Colas, la **primera** cola de mayor prioridad tiene una **política apropiativa**, mientras que para la **segunda** cola, se requiere que escoga entre los siguientes algoritmos de **política no apropiativa**:

Algoritmo First Come First Served *Cola 1: Round Robin, Quantum:

3) Aplicar Penalización y Recompensa

*Las rafagas de CPU Iniciales entrarán a la Cola 1 y
*las rafagas de CPU finales entrarán a la Cola 2

Penalización: Para procesos que estén en la Cola 1 y ejecuten una unidad de tiempo (no de Quantum) se les pregunta si su Prioridad es menor a la Prioridad del proceso anteriormente ejecutado (En la Cola 1); En caso de ser cierto, estos descienden a la Cola 2.

Recompensa: Para procesos que estén en la Cola 2 se les pregunta antes de ejecutarse si su tiempo de espera actual es mayor o igual a la suma de sus rafagas de CPU; En caso de ser cierto, estos ascienden a la Cola 1.

Donde cabe aclarar:

- 1) El programa (En su última Versión) solo recibe datos numéricos como parámetros de entrada, por lo que si se ingresa algún carácter fuera del Dominio el programa no funcionará como debe ser.
- 2) Para el ítem 3 el programa tiene por defecto aplicar los criterios de penalización y recompensa, pero como recomendación le sugerimos desactivarlos para las primeras pruebas.
- 3) En cualquier caso, las rafagas de CPU iniciales entran a la Cola 1 y las rafagas de CPU finales a la Cola 2.
- 4) (Leer criterio de penalización del programa antes de seguir) Hay casos para la penalización de procesos que no están especificados, como por ejemplo el caso de que no haya proceso anterior para comparar O que el proceso anterior sea él mismo (esto ocurre cuando se maneja más de una unidad de quantum). Estos casos se obvian por que no cumplen la condición, pero si están presentes y son “controlados” en la ejecución.

Prueba:

Para la siguiente simulación ingresamos la siguiente información en el programa:

Proceso	Instante llegada	Ráfaga cpu inicial	Ráfaga de e/s	Ráfaga de cpu final	Prioridad
P1	0	1	1	1	1
P2	0	2	1	2	5
P3	3	2	1	3	8
P4	3	5	1	1	0
P5	7	1	1	1	1
P6	4	2	2	2	2
P7	6	1	1	2	5
P8	7	2	1	1	1
P9	7	4	2	1	10
P10	8	1	1	1	0

Cola 1: RR, Quantum: 1

Cola 2: FCFS

Criterios: Si

Después de ingresar la información anterior, damos click al botón Empezar Simulación

Y como resultado obtuvimos esta tabla:

RESULTADOS

Proceso	Tiempo Servicio(T)	Tiempo Espera(E)	Indice Servicio(I)	Tiempo Final(tf)
P1	17	14	0.1176470588235...	17
P2	19	14	0.2105263157894...	19
P3	21	15	0.2380952380952...	24
P4	34	27	0.1764705882352...	37
P5	19	16	0.1052631578947...	26
P6	28	22	0.1428571428571...	32
P7	27	23	0.1111111111111...	33
P8	21	17	0.1428571428571...	28
P9	22	15	0.2272727272727...	29
P10	19	16	0.1052631578947...	27

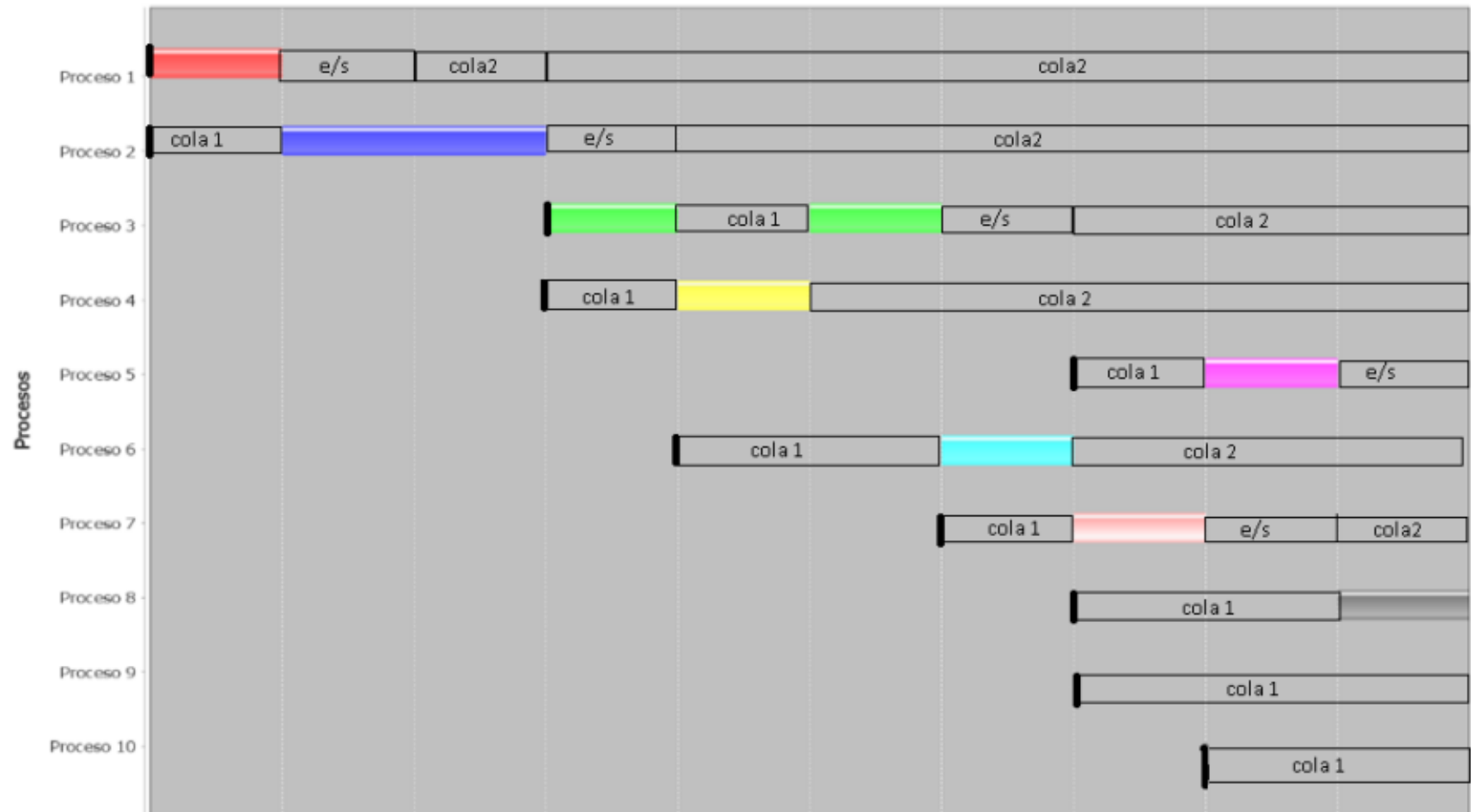
Cola 1: RR, Quantum: 1

Cola 2: FCFS

Criterios: Si

	Tiempo Servicio(T)	Tiempo Espera(E)	Indice Servicio(I)
PROMEDIO:	22	17	0.1577363640831133

El programa también nos permite graficar un diagrama de Gantt con el procedimiento de la simulación: desde el instante 0 al instante 10 obtuvimos:



Análisis del intervalo anterior:

En la figura anterior podemos observar que el programa se ejecuta bastante bien, le hicimos la prueba de escritorio y corresponden exactamente los datos arrojados por el programa que con los que debería arrojar. por ejemplo, comenzamos y vemos que en el tiempo cero llegan el proceso 1 y 2 pero solo se ejecuta el proceso 1 y el proceso 2 queda en espera en la cola uno; También observamos que para el proceso 1 en el momento en que ejecuta su nanosegundo en CPU entra al estado de e/s y cuando hace su proceso de entrada y salida se encola en cola 2. Y así vamos observando proceso por proceso nanosegundo a nanosegundo como el programa nos muestra el estado de cada proceso y así mismo corroborar de que toda nuestra información es correcta.

Aquí una foto del diagrama completo:

