



---

# Sistemas Operativos I

## [Entrega final]

---

2º INGENIERÍA INFORMÁTICA. GRUPO A



**Autores:**

José Javier Atienzar González

Iván Larios López

José Ramón Martínez Riveiro

Víctor Pérez Piqueras

**Trabajo en grupo:**

Simulador de un planificador a corto plazo

**Entrega Final: Memoria**

## Índice:

1. Introducción.....	1
2. El problema de la planificación.....	1
3. Parámetros de evaluación del planificador.....	2
4. Objetivos del planificador .....	3
5. Niveles de planificación.....	3
6. Algoritmos de Planificación no Expulsivos .....	4
6.1 Primero en llegar primero en ejecutar (FCFS).....	4
6.2 Primero el trabajo más corto (SJF) .....	5
6.3 Planificación basada en prioridades.....	5
7. Algoritmos de Planificación Expulsivos .....	6
7.1 Turno rotatorio (round-robin).....	6
7.2 Primero el de menor tiempo restante (SRTF) .....	6
7.3 Planificación expulsiva basada en prioridades.....	7
7.4 Colas multinivel .....	7
8. Código del planificador.....	9
8.1 Descripción general del código. ....	9
8.2 Formato de Entrada .....	10
8.3 Estructuras de Datos .....	10
8.4 Resultado de la ejecución: .....	11
8.5 Tratamiento de errores: .....	13
9. Conclusión .....	13
10. Bibliografía .....	13

## Tabla de ilustraciones:

Figura 1.....	1
Figura 2.....	4
Figura 3.....	5
Figura 4.....	6
Figura 5.....	8
Figura 6.....	11
Figura 7.....	11
Figura 8.....	12
Figura 9.....	12
Figura 10.....	12

## 1. Introducción.

La planificación es la gestión del procesador realizada por el sistema operativo a través de distintas políticas y mecanismos. Como objetivo tiene dar un buen servicio a los procesos y como función el reparto de tiempo del procesador para los procesos.

En este trabajo sobre la planificación de recursos, trataremos temas como las diferencias entre una planificación expulsiva y una no expulsiva, los parámetros de evaluación del planificador desde un punto de vista de usuario y de sistema y para concluir hemos tratado el tema de los niveles de planificación existentes situándolos a estos en un diagrama de estados de procesos.

## 2. El problema de la planificación

La planificación de recursos es necesaria cuando múltiples usuarios necesitan usar de forma exclusiva un determinado recurso, que puede constar de uno o más ejemplares. La planificación determina en cada instante qué ejemplar se le asigna a cada uno de los usuarios que requiera utilizar el recurso en ese momento. Cuando existe más de un recurso la planificación lleva a cabo una **multiplexación espacial** de los recursos, puesto que determina que recurso específico se asigna a cada usuario. Además, la planificación lleva a cabo una **multiplexación temporal** al determinar en qué instante se le asigna un recurso a un usuario.

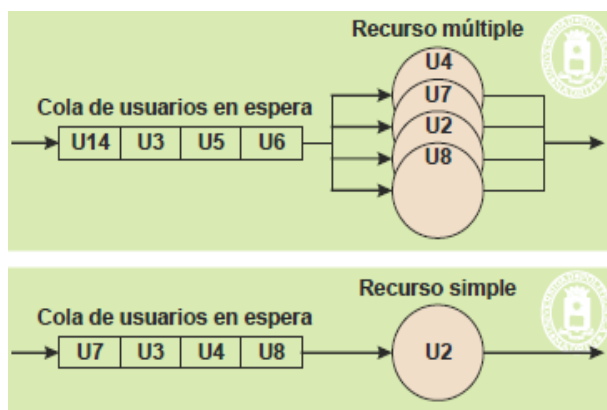


Figura 1. Planificación en un sistema con múltiples ejemplares de un recurso y con recurso simple.

Como muestra la ilustración 1, las solicitudes de los usuarios se pueden acumular, formándose una cola o varias colas de espera.

Hay dos tipos principales de algoritmos de planificación de recursos. Estos son:

- **Planificación no expulsiva o no apropiativa.**

Su funcionamiento consiste en asignar un recurso a un usuario y mantenerlo hasta que este termine, ya sea por bloqueo (inicio E/S), espera para otro proceso o terminación voluntaria. Este esquema solo se activará cuando quede un ejemplar libre y haya usuarios en estado de espera.

- **Planificación expulsiva o apropiativa.**

A diferencia de la no expulsiva, esta planificación es capaz de reasignar un recurso de uso por un usuario a otro, siendo el recurso por tanto expropiable. En este esquema se activará tanto si queda un recurso libre como si llega otra petición o el tiempo de uso de recurso asignado al usuario actual se termina.

Por tanto, la principal diferencia entre estos dos esquemas es la capacidad o no de reasignar un recurso que está siendo usado por un usuario a otro.

Existe una gran variedad de recursos planificables podemos destacar:

- **El procesador:** Recurso básico del computador. La cola de procesos listos se corresponde con la de espera, de tipo expropiable y que tiene la propiedad de la afinidad.
- **La memoria:** En sistemas con memoria virtual se produce multiplicación espacial y temporal de los marcos de página.
- **El disco:** Las peticiones de acceso al disco por parte de procesos se tienen que planificar para determinar su orden.

**Los mecanismos de sincronización:** Hay que planificar a cuál de los procesos que están esperando se les asigna ese mutex o cerrojo que es liberado. [1]

### 3. Parámetros de evaluación del planificador

El objetivo de la planificación es optimizar el comportamiento del sistema informático. Al ser el comportamiento informático muy complejo, el objetivo de la planificación se centrará en la faceta del comportamiento en la que se esté interesado en cada situación

Parámetros de evaluación del planificador

Se pueden usar dos tipos de parámetros:

- **De sistemas**, los cuales se centran principalmente en el uso eficiente del procesador.
- **De usuario** (ya sean de proceso o threads), basados en el comportamiento del sistema tal y como lo perciben los usuarios y procesos.

Relativos al tiempo que toman las distintas acciones de los procesos, lo cual es perceptible para el usuario o el proceso.

- **Tiempo de ejecución ( $T_e$ ):** Es el tiempo que tarda en ejecutarse un proceso o *thread* desde que es creado hasta que termina totalmente. Incluye todo el tiempo en que el proceso está listo para ejecutar, en ejecución y en estado bloqueado (por sincronización o entrada/salida).
- **Tiempo de espera ( $T_w$ ):** Reúne el tiempo que pasa un proceso en la cola de procesos listos para ejecutar. Si el proceso no se bloquea nunca, es el tiempo que espera el proceso o *thread* en estado listo para ejecutar antes de que pase al estado de ejecución.
- **Tiempo de respuesta ( $T_a$ ):** Tiempo que pasa entre el momento en que se crea el proceso y se pone listo para ejecutar y la primera vez que el proceso responde al usuario. Es fundamental en los sistemas interactivos, ya que un sistema de

planificación se puede configurar para responder muy rápido al usuario, aunque luego el proceso o *thread* tenga un tiempo de ejecución largo.

Desde el punto de vista de la planificación, un sistema se caracteriza con dos parámetros principales:

- **Uso del procesador(C):** Se basa en expresar porcentajes de uso útil del procesador por unidad de tiempo, lo cual implica una gran variación muy dependiente de las características de cada sistema. Un ejemplo sería un el procesador de un desktop, el cual generalmente no suele superar un 15% de uso. Por otro lado, un servidor con bastante tráfico rondará un uso del 95%. Este parámetro es importante por lo general, pero su importancia destaca en sistemas de tiempo real y sistemas con calidad de servicio.
- **Tasa de trabajos completados (P):** Este parámetro indica el número de procesos o threads ejecutados completamente por unidad de tiempo. Se puede calcular como la inversa del tiempo medio de ejecución ( $1/\text{Media}(T_e)$ ). Este parámetro es para indicar el número de procesos o threads que se ejecutan completamente por unidad de tiempo. Por tanto, para obtener esta tasa se calculará la inversa del tiempo medio de ejecución mediante la fórmula. [1]

$$P = \frac{1}{\text{Tiempo de ejec. medio}}$$

## 4. Objetivos del planificador

- **Propósito general:**
  - Maximizar el uso del procesador (C) y el número de procesos ejecutados por unidad de tiempo (P).
  - Minimizar el tiempo de ejecución de un proceso, tiempo de respuesta y espera ( $T_e, T_w, T_a$ ).
- **Interactivo:**
  - Maximizar la reducción del tiempo de respuesta ( $T_w$ )
- **Batch** (o por lotes):
  - Maximizar C y P para explotar al máximo el procesador. [1]

## 5. Niveles de planificación

### Niveles de planificación de procesos

Los sistemas pueden incluir varios niveles de planificación de procesos. La figura 2 muestra el caso de tres niveles: **corto, medio y largo plazo**.

El objetivo de la planificación a **largo plazo** es añadir nuevos procesos al sistema, tomándolos de la lista de espera. Estos procesos son de tipo batch, en los cuales no importa el instante preciso en el que se ejecuten (siempre que se cumplan ciertos límites de espera). [2]

La planificación a **medio plazo** trata la suspensión de procesos. Es la que decide qué procesos pasan a suspendido y cuáles dejan de estar suspendidos. Añade o elimina

procesos de memoria principal modificando el grado de multi-programación. La planificación a corto plazo, por tanto, se encarga de seleccionar el proceso en estado de

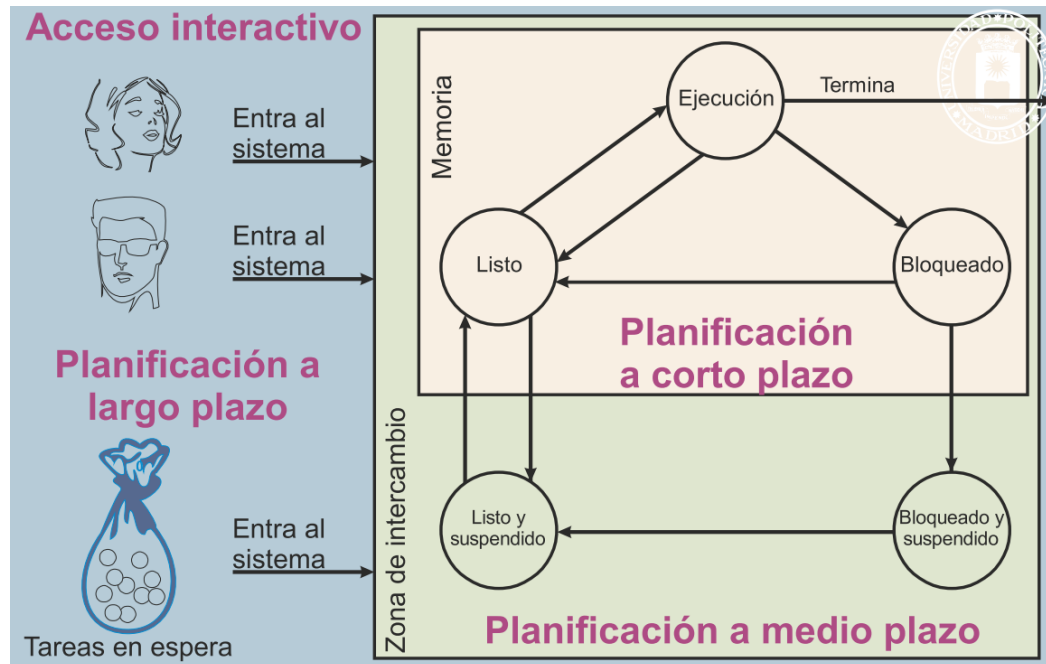


Figura 2. Tipos de planificación.

listo que pasa a estado de ejecución. Es, por tanto, la que asigna el procesador. Debido a la frecuencia con que se activa (cerca de un centenar de milisegundos en un sistema de turno rotatorio), el planificador a corto plazo debe ser rápido y generar poca carga para el procesador. [1]

## 6. Algoritmos de Planificación no Expulsivos

### 6.1 Primero en llegar primero en ejecutar (FCFS)

El algoritmo FCFS (First Come First Served) selecciona al usuario que lleva más tiempo esperando en la cola de listos.

Es un algoritmo sencillo, que introduce muy poca sobrecarga en el sistema, lo que permite obtener el máximo rendimiento del procesador. Como no es expulsivo los usuarios mantienen el recurso hasta que dejan de necesitarlo.

Por ejemplo, un proceso que recibe el procesador la mantiene hasta que requiere una operación de entrada/salida o de sincronización. Por ello, este algoritmo beneficia a los procesos intensivos en procesamiento frente a los intensivos en entrada/salida. Puede producir largos tiempos de espera, por lo no es adecuado para los sistemas interactivos, estando limitado su posible uso a sistemas de lotes. Sin embargo, todo proceso llega a ejecutar, por lo que no produce inanición. [1]

## 6.2 Primero el trabajo más corto (SJF)

El algoritmo SJF (Shortest Job First) elige el proceso con ráfaga de procesamiento más corta de la cola de procesos listos.

Es necesario pues conocer la duración de estas ráfagas, cosa imposible, a no ser que se trate de procesos por lotes repetitivos, en los cuales si se conoce la duración de las ráfagas de ejecución. Una alternativa es calcular estas ráfagas extrapolando los datos de las anteriores en cada proceso.

Minimiza tiempo de espera medio  $T_w$ , penalizando trabajos largos y tiene el riesgo de inanición de usuarios de larga duración.

Puede producir carga del sistema puesto que hay que recorrer la cola de procesos listos para buscar el corto o tenerla ordenada además de estimar las ráfagas.

Más adelante estudiaremos la versión expulsiva de este algoritmo (SRTF). [1]

## 6.3 Planificación basada en prioridades

Este planificador selecciona al usuario con la mayor prioridad. Si existen varios usuarios (por ejemplo, procesos listos) con igual prioridad se utiliza otro de los algoritmos, por ejemplo, el FCFS, para seleccionar al agraciado. Tiene la ventaja de proporcionar grados de urgencia. El sistema mantiene una cola de usuarios por prioridad, como se muestra en la figura 3. [1]

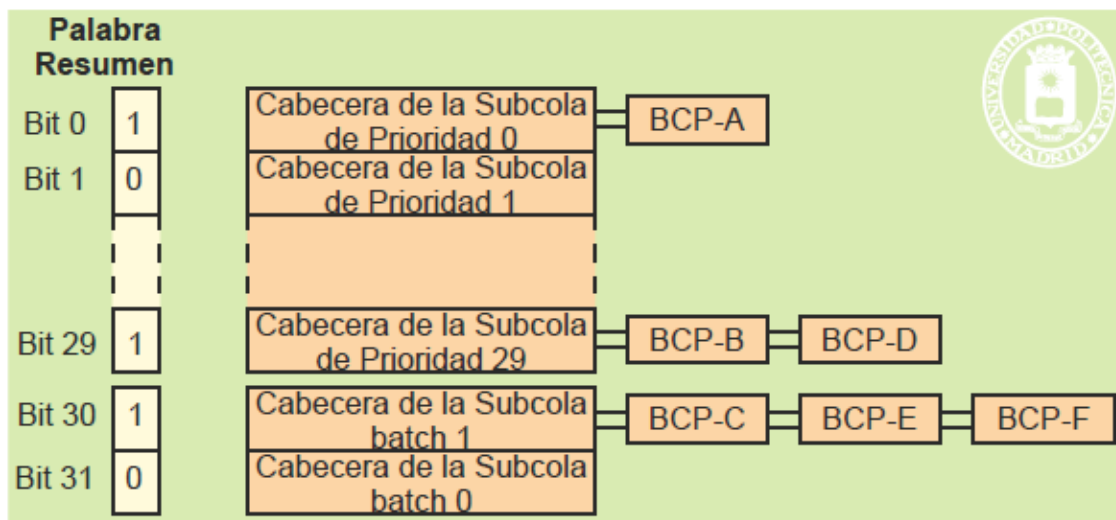


Figura 3. Ejemplo de colas de procesos organizadas por prioridad. La palabra resumen permite acelerar la búsqueda, puesto que solamente hay que analizar las niveles que tienen el resumen a 1.

## 7. Algoritmos de Planificación Expulsivos

### 7.1 Turno rotatorio (round-robin)

Como se ve en el apartado 3.14.4 del libro “Sistemas Operativos” de la UPM, un algoritmo antiguo y sencillo utilizado para repartir de forma equitativa el procesador entre los procesos listos que se encuentran en una cola, proporcionando un tiempo respuesta ( $T_a$ ) acotado. Tiene la ventaja de ofrecer reparto equitativo, lo que lo hace muy útil en un entorno de trabajo compartido. Este algoritmo fue ideado principalmente para sistemas de tiempo compartido.

Es una variación del algoritmo FCFS, puesto que se concede el procesador por un tiempo máximo acotado, denominado rodaja o cuanto. Un proceso estará en ejecución y cuando su cuanto termine será expulsado y puesto en la lista de espera. Otra alternativa es que el proceso entre en bloqueo, siendo también expulsado y esperando a que estuviera en listo para volver a lista de espera. [1]

Es un algoritmo expulsivo, que solamente expulsa el proceso si ha consumido enteramente su rodaja. En la figura 4 podemos observar como en el primer esquema sale en ejecución un proceso pero, al pasar un ciclo, este es desplazado a la última posición dejando paso al siguiente en la cola. [3]

Un aspecto importante es el tamaño del cuanto, ya que un cuanto grande proporciona un algoritmo muy similar al FCFS, mientras que un cuanto muy pequeño hará que la mayor parte del tiempo sea invertida en las conmutaciones del procesador. En el diseño de los cuantos existen varias alternativas, como las siguientes:

- Rodaja igual para todos los procesos, donde si los procesos tienen tiempos de ejecución parecidos el rendimiento será muy alto.
- Rodaja distinta según el tipo de proceso, útil para sistemas con procesos de diferentes tamaños.
- Rodaja dinámica, cuyo valor se ajusta según sea el comportamiento del proceso o el comportamiento global del sistema. [1]

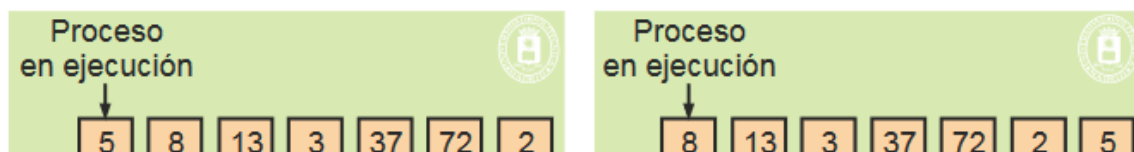


Figura 4. Paso de un proceso a la última posición tras terminar su cuanto, cediendo paso al siguiente. [1]

### 7.2 Primero el de menor tiempo restante (SRTF)

Es similar al SJF, con la diferencia de que si un nuevo proceso pasa a listo se activa el dispatcher para ver si es más corto que lo que queda por ejecutar del proceso en ejecución. Si es así, el proceso en ejecución pasa a listo y su tiempo de estimación se decrementa con el tiempo que ha estado ejecutándose. [4]



En SRTF se penaliza a las ráfagas largas (como en SJF). Un punto débil de este algoritmo se evidencia cuando una ráfaga muy corta suspende a otra un poco más larga, siendo más larga la ejecución en este orden al ser preciso un cambio adicional de proceso y la ejecución del código del planificador. Trabajos cortos.

#### **Características:**

- De los procesos que están esperando para usar la CPU, SRTF lleva a ejecución el proceso al que le reste menos tiempo para terminar.
- Los empates se dirimen mediante FIFO / FCFS

#### **Funcionamiento:**

- Los procesos llegan a la cola y solicitan un intervalo de CPU
- Si dicho intervalo es inferior al que le falta al proceso en ejecución para abandonar la CPU, el nuevo proceso pasa a la CPU y el que se ejecutaba a la cola de preparados.

#### **Inconvenientes:**

- El intervalo de CPU es difícil de predecir
- Posibilidad de inanición: los trabajos largos no se ejecutarán mientras hayan trabajos cortos. [1]

### 7.3 Planificación expulsiva basada en prioridades

Es una versión expulsiva del algoritmo de planificación basada en prioridades, de forma que, si llega un usuario más prioritario que el que está ejecutando, se expulsa a este para dar el recurso al más prioritario. Relacionada con la planificación basada en prioridades está la expulsiva basada en prioridades.

Sólo se diferencia de esta en que en esta versión si llega un proceso más prioritario que el que se está ejecutando, el ejecutado es expulsado para dar recurso al prioritario. [1]

### 7.4 Colas multinivel

Los procesos se organizan en distintas colas y se aplica un algoritmo de planificación distinto a cada cola. En las colas de prioridad se puede aplicar un round-robin entre los procesos de cada una, mientras que en las colas batch se puede aplicar un SJF con expulsión si aparece un proceso con prioridad.

El esquema de colas multinivel se caracteriza por los siguientes parámetros:

- El número de niveles existentes, es decir, el número de clases de procesos que se distinguen.
- El algoritmo de planificación de cada nivel.
- El esquema de planificación que se usa para repartir el procesador entre los distintos niveles.

Para evitar la falta de trabajo se puede añadir un mecanismo que, pasado un tiempo, mueva un proceso que ha estado en cola al siguiente nivel.

En la figura 5 se muestra un ejemplo con tres niveles sin realimentación, tal que en cada nivel se usa un round-robin con diferentes tamaños de cuanto. En cuanto al mecanismo de planificación entre niveles, se usa un esquema de prioridad tal que sólo se ejecutarán

procesos de un determinado nivel si en los niveles de mayor prioridad no hay ningún proceso listo.

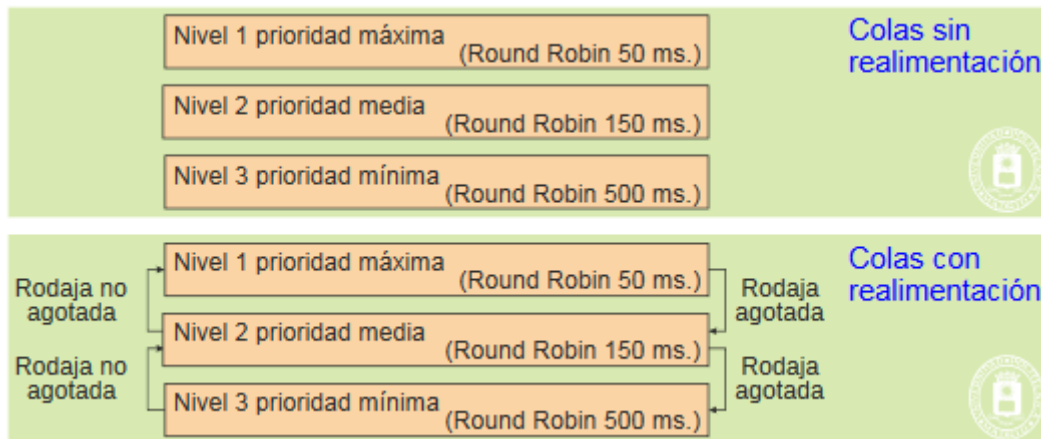


Figura 5. En ella se muestran colas multinivel con y sin realimentación. [1]

En el modelo de colas con realimentación, el sistema operativo puede cambiar de nivel a un proceso dependiendo de su evolución.

En este nuevo modelo existe un parámetro adicional: la política de cambio de nivel, que establece en qué circunstancias un proceso incluido en un determinado nivel pasa a formar parte de otro nivel. En el caso de la figura 5 se puede establecer que un proceso que no agota su cuanto se le sube de nivel, mientras que un proceso que agota su cuanto se baja de nivel. Con esta estrategia, los procesos con un mayor uso de la entrada/salida, estarán en mayor nivel de prioridad, mientras que los intensivos en el uso del procesador se situarán los niveles de menor prioridad, quedando en el nivel intermedio los procesos con un perfil mixto. [1]

## 8. Código del planificador

### 8.1 Descripción general del código.

El código en C es el resultado de la implementación del planificador que originalmente se pide realizar en el proyecto. Utilizaremos los algoritmos SJF y RR.

- Al ejecutar el programa, lo primero que pide son los datos del fichero de entrada, con los que se realizará la posterior ejecución.
- Si en algún momento de la lectura de la entrada el programa detecta algún error, por ejemplo, el número de parámetros es incorrecto, mostrará un mensaje de error para notificarnos.
- Si todo está correcto, leerá el quantum (para Round Robin), el número de procesos, y para cada proceso su nombre, especificado con números, por ejemplo, y separados por espacios, el tiempo de llegada y el tiempo de duración en este orden.
- Tras la lectura se almacenarán los datos en las estructuras de datos correspondientes y se realizará el algoritmo **SJF**.
- Se ordenarán los procesos por orden de llegada de acuerdo con los ciclos totales necesarios y los tiempos de uso del procesador. Cuando haya varios disponibles seleccionará el de menor tiempo de ejecución.
- Después calculará los tiempos de espera, permanencia y respuesta correspondientes a cada proceso. El tiempo de respuesta es igual al tiempo de espera, pues en el SJF los procesos entran una única vez a CPU y sólo salen cuando han sido completamente ejecutados.
- Almacenará el uso de la CPU en porcentaje y el reparto temporal de la CPU.
- Al acabar el algoritmo se almacenarán todos estos datos en un fichero de texto en el que se mostrará además un diagrama de Gantt de los procesos.
- Seguidamente se realizará el algoritmo **Round Robin**.
- Este algoritmo creará una cola con los procesos que se encuentren en espera (Proceso colaDentro), la variable del procesador (Procesos cpup) y una serie de variables auxiliares que llevan a cabo el completo funcionamiento del algoritmo.
- El algoritmo hará una simulación ciclo por ciclo.
- Cada ciclo se comprobará comparando el tiempo de llegada con el ciclo de procesador si algún proceso ha llegado.
- El procesador se mantiene inactivo hasta que llega el primer proceso, el cual entra a procesador en el ciclo que entra.
- Cada ciclo, el proceso que está en procesador disminuirá su tiempo de ejecución restante un ciclo y el contador de ciclos en procesador aumentará un ciclo.
- Cuando el contador de ciclos en procesador detecte que el cuanto ha terminado, el proceso en procesador se pondrá al final de la cola y el primer proceso en cola entrará a procesador.
- Cuando un proceso termine completamente su ejecución, se disminuirá el tamaño de la cola y se introducirá el primer proceso en cola en procesador.

## 8.2 Formato de Entrada

El nombre del fichero de entrada debe ser estrictamente: **fichero.txt**

Se especifica el quantum con la siguiente sentencia: `quantum:2`

La siguiente línea contendrá el número de procesos: `nproces:5`

La siguiente línea es: `proceso: llegada duracion`

Por último, una lista de los procesos de la forma:

**NOMBREPROCESO: TIEMPOLLEGADA TIEMPODURACIÓN**

```
0: 3 9
1: 5 3
2: 7 2
3: 8 5
4: 11 1
```

Estos pasos deben ser seguidos literalmente, de otro modo, el programa detectará un error en la introducción de los parámetros del fichero de entrada.

## 8.3 Estructuras de Datos

La información de entrada será recogida en las siguientes estructuras de datos:

- Número de procesos.
- Quantum(RR).
- Uso total de la CPU.
- Uso real de la CPU.
- Porcentaje de uso de la CPU.
- Estructura de datos para los procesos:
  - Nombre del proceso.
  - Tiempo de llegada.
  - Tiempo de duración.
  - Tiempo de permanencia.
  - Tiempo de respuesta.
  - Tiempo de espera.
  - Tiempo restante.

Para el algoritmo Round Robin se crearán dos colas, una para guardar los procesos que se encuentran en cola y otra auxiliar, y dos procesos auxiliares para poder llevar a cabo el algoritmo.

## 8.4 Resultado de la ejecución:

## Resultados ejercicio 4:

## a) SJF

Proceso	Llegada	Duracion	Permanencia	Espera
P0	3	9	9	0
P4	11	1	2	1
P2	7	2	8	6
P1	5	3	13	10
P3	8	5	15	10

El tiempo de respuesta en los algoritmos SJF coincide con el tiempo de espera, por eso no esta incluido  
El uso del CPU es un 86.956520 % 20/23 Tiempo

```

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23
|P0_|P4_|P2_|P1_|P3_|

```

Figura 6. Fichero de salida del algoritmo SJF

## b) RR q=2

Nombre	Tejecucion	Tpermanencia	Trespuesta	Tespera
P0	9	18	0	9
P1	3	4	1	1
P2	2	7	5	5
P3	5	15	6	10
P4	1	7	6	6

Uso CPU: 86.956519%

```

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23
|P0_|P1_|P0_|P2_|P3_|P4_|P0_|P3_|

```

Figura 7. Fichero de salida del algoritmo RR con q=2

c) RR q=3

			P0		P0		P1		P0		P2		P3		P1		P0		P4		P3		P0		P3				
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23						
			Nombre		Tejecucion		Tpermanencia		Trespuesta		Tespera																		
			P0		9		19		0		10																		
			P1		3		11		2		8																		
			P2		2		6		4		4																		
			P3		5		15		5		10																		
			P4		1		8		7		7																		
Uso CPU: 86.956519%																													

Figura 8. Fichero de salida del algoritmo RR con q=3

Resultado del ejercicio de 7 procesos:

a) SJF

	Proceso	Llegada	Duracion	Permanencia	Espera																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																					
	P0	4	8	8	0																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																					
	P1	5	2	9	7																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																					
	P4	10	2	6	4																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																					
	P2	6	3	13	10																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																					
	P5	11	3	11	8																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																					
	P3	7	5	20	15																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																					
	P6	6	7	28	21																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																					
El tiempo de respuesta en los algoritmos SJF coincide con el tiempo de espera, por eso no esta incluido																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																										
El uso del CPU es un 88.235291 % 30/34 Tiempo																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																										
	P0			P1	P4	P2	P5	P3	P6																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																	</

Figura 9. Fichero de salida del algoritmo SJF

b) RR q=2

0	1	2	3	P0	4	5	P1	6	7	8	P0	9	10	P2	11	12	P6	13	P3	14	15	P0	16	17	P4	18	19	P5	20	P2	21	22	P6	23	24	P3	25	P0	26	27	P5	28	29	P6	30	P3	31	P6	32	33	34
				Nombre	Tejecucion							Tpermanencia					Trespuesta					Tespera																													
				P0	8							25					0					17																													
				P1	2							3					1					1																													
				P2	3							17					4					14																													
				P3	5							26					7					21																													
				P4	2							10					8					8																													
				P5	3							19					9					16																													
				P6	7							28					6					21																													
Uso CPU: 88.235295%																																																			

Figura 10. Fichero de salida del algoritmo RR con q=2

## 8.5 Tratamiento de errores:

- **Si falta el quantum:**

```
El formato de quantum es incorrecto. Escribalo con el siguiente formato: quantum:n
Press [Enter] to close the terminal ...
```

- **Si el número de procesos indicados no coincide con el número de procesos en el fichero de entrada:**

```
No se ha escrito informacion del proceso en la linea 8. No deje lineas en blanco.
No se ha especificado informacion del numero de procesos ingresado.
```

- **Si falta el tiempo de llegada o el de duración:**

```
Para el proceso en la linea 0 no se ha especificado tiempo de llegada, tiempo de duracion o el formato es incorrecto.
Press [Enter] to close the terminal ...
```

## 9. Conclusión

En este trabajo sobre la planificación de recursos, hemos tratado temas como las diferencias entre una planificación expulsiva y una no expulsiva, los parámetros de evaluación del planificador desde un punto de vista de usuario y de sistema y para concluir hemos tratado el tema de los niveles de planificación existentes situándolos a estos en un diagrama de estados de procesos. También hemos visto los diferentes tipos de algoritmos de planificación, viendo así su funcionamiento básico, usos, ventajas e inconvenientes y ejemplos. Hemos observado que es algo muy importante y necesario para maximizar la eficiencia de nuestro procesador.

## 10. Bibliografía

- [1] "Sistemas Operativos". Pedro de Miguel Anasagasti y Fernando Pérez Costoya. Escuela Técnica Superior de Informática. Universidad Politécnica de Madrid. (18/05/2016).
- [2] <http://www.sc.ehu.es/acwlaroa/SO2/Apuntes/Cap3.pdf>
- [3] <http://algoritmosplanificacion.blogspot.com.es/2012/08/srtf-short-remaining-time-first.html>
- [4] <http://algoritmosplanificacion.blogspot.com.es/2012/08/srtf-short-remaining-time-first.html>