

# Procesos.

## Planificación del Procesador.

### Sistemas Operativos.

#### Tema 2.

### Concepto de Proceso.

- Una definición sencilla: Programa en ejecución.

Entidad pasiva

Programa

**RECURSOS**

CPU  
Memoria  
Ficheros  
Dispositivos

+

Entidad activa

**Proceso**

Pila

↓

↑

Datos

Código

Otros recursos  
Ficheros abto.  
Disponible E/S, etc

← PC

- El programa necesita una serie de recursos para su ejecución:
  - Tiempo de la CPU.
  - Memoria.
    - Con el contenido del programa.
    - Pila para datos temporales.
    - Sección de datos con variables globales.
  - Acceso a archivos y dispositivos E/S.

## Concepto de Proceso.

- Para entender mejor el concepto de proceso:
  - Los procesos tienen un carácter secuencial:
    - Un proceso en su ejecución puede generar más de un proceso (llamada fork).
    - Dos procesos pueden asociarse al mismo programa.
- Proceso: Unidad de trabajo del sistema.
  - En general, habrá más de un proceso ejecutándose concurrentemente.
- Procesos de usuario y procesos del sistema.
- El sistema operativo se encargará de:
  - La creación y eliminación de procesos.
  - La planificación de procesos.
  - La sincronización, comunicación y manejo de bloqueos mutuos entre procesos.

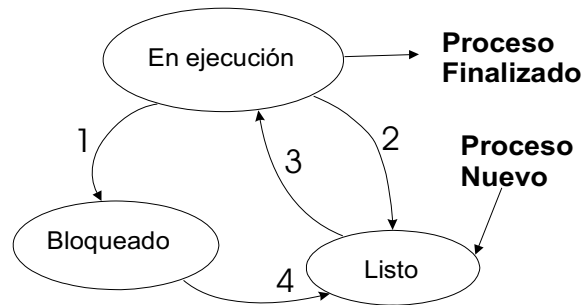
## Estado de un proceso.

- El estado de un proceso:
  - Relacionado con su actividad del proceso en un cierto momento.
  - Al ejecutarse irá cambiando de estado.
- Posibles estados de un proceso:
  - En ejecución: Está usando el procesador.
  - Bloqueado: No puede hacer nada porque está esperando un evento externo (esperando la conclusión de E/S).
  - Listo: Está en memoria esperando turno para ejecutarse en la CPU (espera asignación del procesador).

## Estado de un proceso.

- Diagrama de transición de estados:

- 1.- Pasa a esperar un suceso (E/S) y se bloquea.
- 2.- Expulsión de proceso de la CPU
- 3.- El planificador elige otro proceso.
- 4.- El suceso (E/S) que esperaba el proceso acaba.



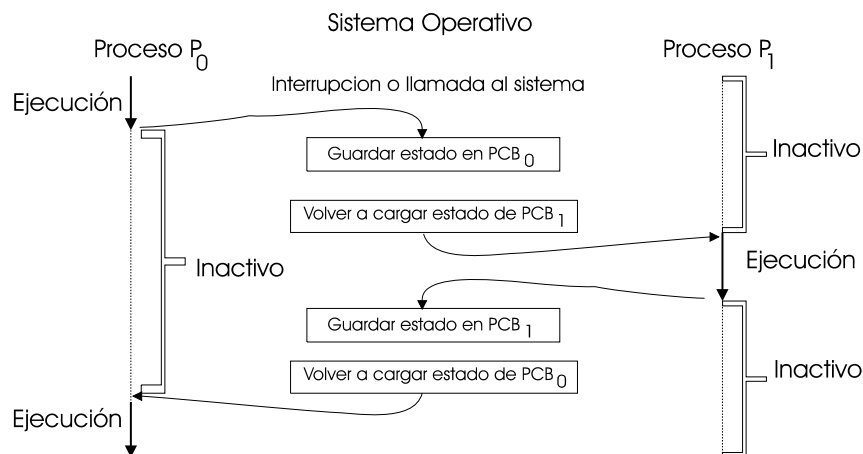
- En un instante: sólo un proceso en ejecución, los demás estarán listos o en espera.

## Bloque de control del proceso (PCB)

- En el S.O.:
  - Un proceso se representa por: Un Bloque de Control del Proceso (PCB, Process Control Block).
- Es un conjunto de registros que almacena información sobre el proceso:
  - Estado del proceso: Nuevo, Listo, en Ejecución, Bloqueado.
  - Contador del programa: Dirección siguiente instrucción a ejecutar.
  - Registros de la CPU: Contenidos al final de la ultima ejecución (contador de programa, puntero a pila, registros de datos, etc.).
  - Información planificación CPU: prioridad, apuntadores a las colas, algoritmo usado.
  - Información contable y de identificación: Número de proceso, tiempo real y de CPU utilizado.
  - Información estado E/S: Solicitudes E/S pendientes, lista archivos abiertos, etc.

## Bloque de control del proceso (PCB)

- Se utiliza para poder ejecutar procesos concurrentes: **hay un cambio de contexto** (se produce una interrupción que debe atender el sistema operativo)

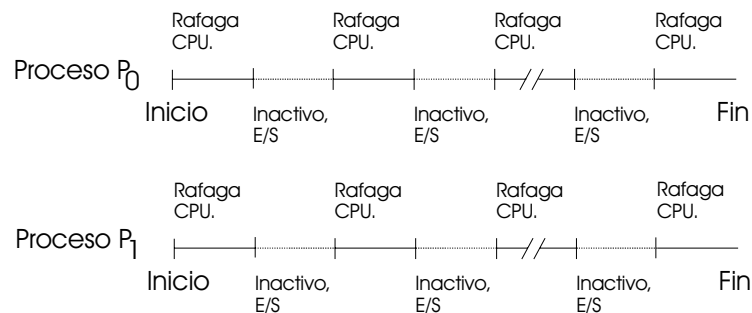


## Concepto de hilo de ejecución: thread.

- 1) Varios procesos pueden cooperar para resolver una misma tarea. Tendremos *ejecución concurrente entre procesos comunicados por memoria*
- 2) Un programa podría realizar actividades concurrentes (paralelismo dentro del proceso). Tendremos: *Ejecución concurrente de varios "hilos" dentro de un proceso.*
- Cada **hilo, thread o proceso ligero** tiene su propio:
  - Contador de programa, pila, registros y estado del proceso ligero
- Los procesos ligeros de un mismo proceso comparten la información del proceso:
  - Espacio de memoria, Variables globales, Archivos abiertos, Procesos hijos, Temporizadores, Señales y semáforos, Contabilidad

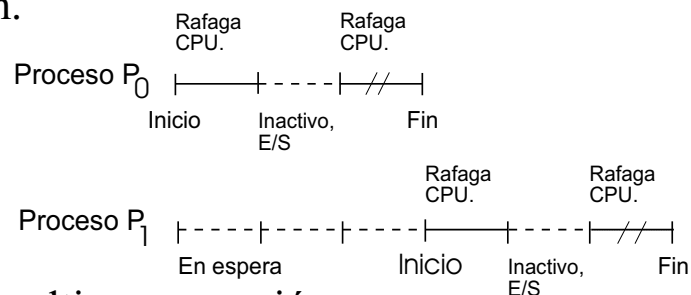
## Concepto de planificación.

- Si hay dos procesos listos para ejecución ...
  - ¿ Cual se ejecutará primero?
- El planificador (scheduler) del sistema operativo decide cual.
- El planificador utiliza  $\Rightarrow$  Algoritmo de planificación.
- Un ejemplo de planificación de procesos:  $P_0$  y  $P_1$  listos

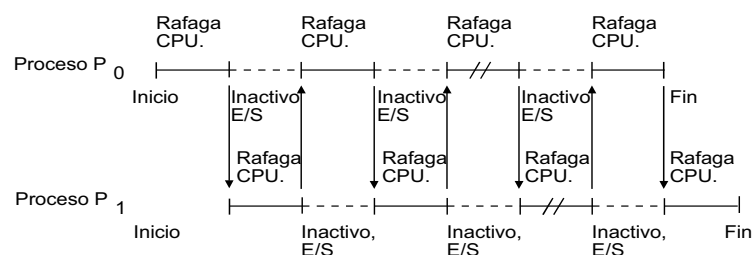


## Concepto de planificación.

- Una manera sencilla de planificación, sin multiprogramación.



- Planificación con multiprogramación.

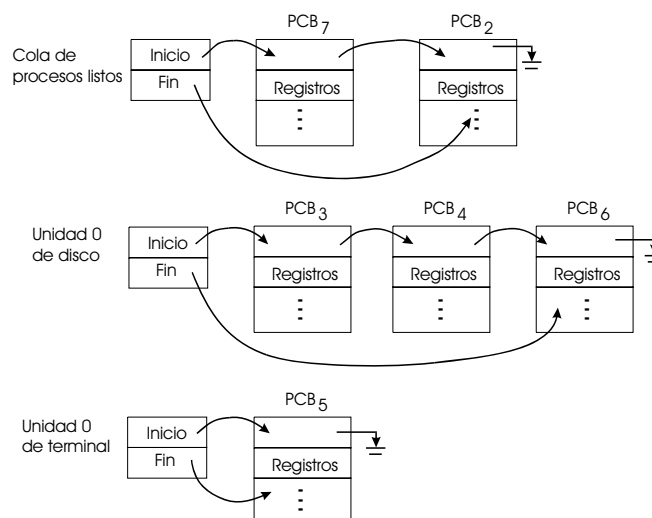


## Concepto de planificación.

- Colas de planificación:
  - El S.O. usa una serie de colas para planificar los recursos (Memoria, E/S, CPU etc.).
  - Cola de trabajos:
    - Procesos en almacenamiento secundario esperando memoria principal.
  - Cola de procesos listos:
    - Procesos en memoria principal, listo y esperando su ejecución (una lista ligada).
  - Cola de dispositivos:
    - Para cada dispositivo (disco, impresora, etc.) hay una cola de procesos esperando utilizarlo.

## Concepto de planificación.

- Se usan los PCB como elementos de las colas:



- ¿Puede haber un mismo PCB en más de una cola?

## Concepto de planificación.

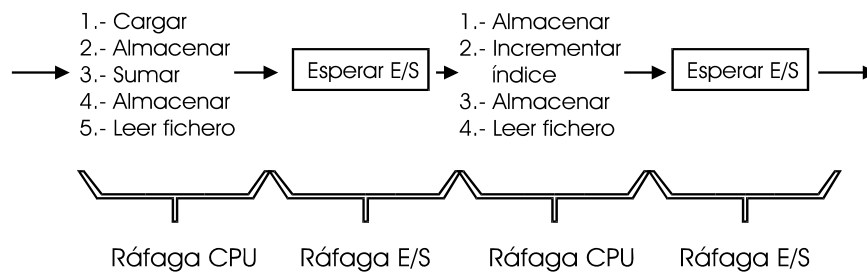
- Un proceso cambia de cola a lo largo de su ejecución.
- Planificador:
  - Elemento del sistema operativo que selecciona procesos en esas colas.
- En lo que a ejecución de procesos respecta:
  - Planificador a largo plazo (planificador de trabajos).
  - Planificador a corto plazo (planificador de la CPU).

## Concepto de planificación.

- Planificador de trabajos:
  - Necesidad:
    - Si hay muchos procesos ... algunos en almacenamiento secundario.
  - Cometido:
    - Se encarga del intercambio entre memoria y almacenamiento secundario. Controla el número de procesos en memoria (grado de multiprogramación).
  - Frecuencia:
    - Se ejecuta con menor frecuencia que el planificador CPU (cuando termina un proceso, etc.) ... puede ser más lento
  - Eficiencia:
    - Buena mezcla en memoria entre procesos limitados por la CPU y por E/S

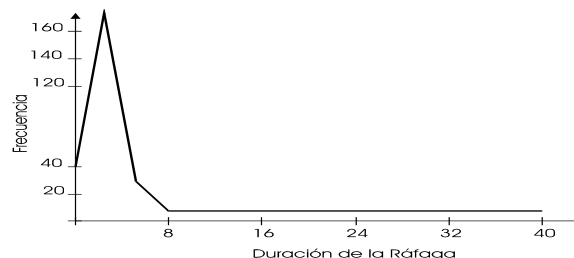
## Concepto de planificación.

- Ejecución de un proceso: Ciclo de ráfagas CPU y E/S:
  - En la ejecución de un proceso se alternan la ejecución en CPU y la espera de E/S.
    - Ráfaga CPU: Tiempo de ejecución en CPU entre dos E/S.
    - Ráfaga E/S: Tiempo entre solicitud y terminación de E/S.
- Ejemplo gráfico de ejecución de proceso:



## Concepto de planificación.

- Histograma típico de duración de ráfagas de CPU:



- Un proceso limitado por la E/S:
  - Predomina la duración de ráfagas de E/S, (normalmente, muchas ráfagas de CPU breves).
- Un proceso limitado por la CPU:
  - Predomina la duración de ráfagas de CPU, (normalmente, pocas ráfagas de CPU de larga duración).



## Concepto de planificación.

- Planificador de la CPU: (planificador a corto plazo)
  - **Cometido:** Selecciona un proceso listo y le asigna CPU, reparte el tiempo de CPU entre procesos.
  - **Frecuencia:** Mayor que la del planificador de trabajos ... debe ser más rápido.
- Pueden efectuarse decisiones de planificación de la CPU:
  - 1.- Proceso cambia de estado de ejecución a bloqueado (E/S).
  - 2.- Cuando termina un proceso.
  - 3.- Proceso cambia de estado de ejecución a listo (interrupción).
  - 4.- Un proceso cambia de estado bloqueado a listo (acaba E/S).
- En 1 y 2 se debe seleccionar un nuevo proceso para ejecución.
- En 3 y 4 puede o no hacerse:
  - Sí: Esquema de planificación expulsiva o apropiativa.
  - No: Esquema de planificación hasta terminación (no apropiativo).

## Concepto de planificación.

- Cambio de contexto (cambio de proceso):
  - Cambio del proceso en ejecución.
  - Requiere:
    - Guardar el estado del proceso que se estaba ejecutando (PCB).
    - Cargar el estado (PCB) guardado para el nuevo proceso que se ejecutará.
  - Su duración es un “gasto de tiempo” (típica de 1-100 microsg) y depende de:
    - La velocidad de la memoria.
    - El número de registros.
    - Existencia de instrucciones especiales (una sola instrucción para cargar o almacenar todos los registros), Etc.

## Algoritmos de planificación.

- Algoritmo de planificación:
  - Decide el proceso de la cola de procesos listos al que se le asigna CPU.
- Criterios para comparar algoritmos de planificación:
  - Equidad: procesos usan la CPU de forma equitativa.
  - Eficiencia (utilización de la CPU): 100% uso.
  - Tiempo de retorno (o de trabajo global): tiempo que tarda en ejecutarse un proceso concreto.
  - Tiempo de respuesta: minimizar el tiempo de respuesta para usuarios interactivos.
  - Tiempo de espera: tiempo que un proceso espera en la cola de procesos listos.
  - Rendimiento (productividad): número de trabajos procesados por unidad de tiempo.

## Algoritmos de planificación.

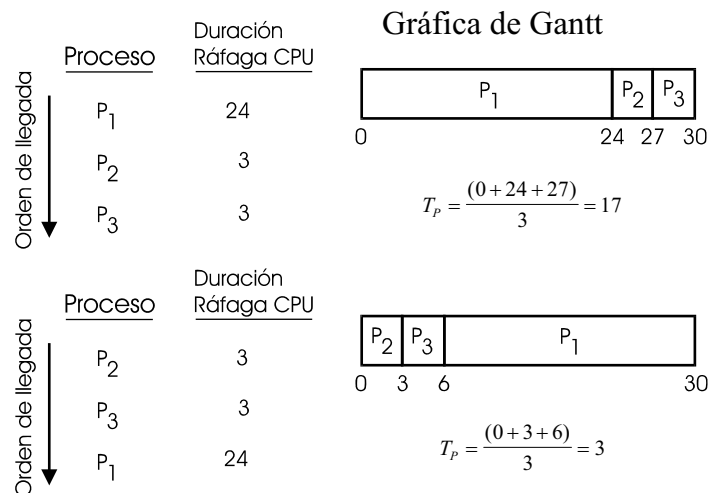
- Objetivos:
  - Maximizar eficiencia y rendimiento.
  - Minimizar tiempo de retorno, espera y respuesta.
- Se pueden intentar optimizar los valores promedio, máximos o mínimos.
- Para que todos los usuarios buen servicio podemos minimizar el tiempo de respuesta.
- En los algoritmos que veremos calcularemos el tiempo promedio de espera.

## Orden de llegada (FCFS).

- FCFS (first-come, first-served):
  - El primer proceso que entró en la cola de procesos listos es el primero al que se le asigna CPU.
  - Se implementa con una cola FIFO.
- El tiempo promedio de espera suele ser bastante largo:
  - Ejemplo (efecto convoy):
    - Un proceso A, limitado por CPU, se ejecuta y retiene la CPU.
    - Los demás acabarán su E/S y pasan a la cola de listos.
    - Cuando acaban, entra A y retiene de nuevo la CPU.
    - Todos vuelven a esperar otra vez (ráfagas cortas CPU).
- Es un algoritmo del tipo ejecución hasta terminación.

## Orden de llegada (FCFS).

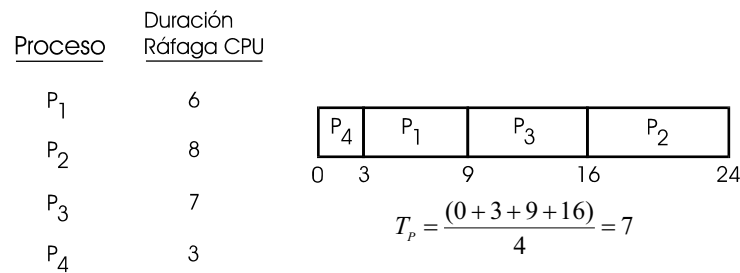
- Ejemplo de cálculo de tiempos promedio(en msg):



- T<sub>p</sub> se reduce (variable con orden llegada).

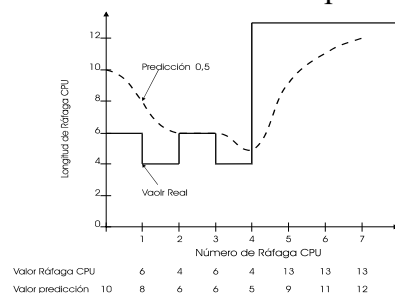
## Primero el trabajo más corto (SJF).

- SJF (shortest-job-first):
  - Se asocia a cada proceso la longitud de su siguiente ráfaga de CPU.
  - Si CPU disponible se le asigna al proceso de menor longitud de ráfaga. Si hay dos con igual longitud de ráfaga se usa FCFS.
- Ejemplo:



## Primero el trabajo más corto (SJF).

- Es óptimo con el criterio del tiempo promedio de espera. Se usa frecuentemente.
- Problema:
  - ¿Como conocer la longitud de la siguiente ráfaga de CPU?
  - Se intenta predecir longitud de siguiente ráfaga de CPU:
    - Un método sencillo: el valor de la última ráfaga de CPU.
    - Como Promedio exponencial de ráfagas anteriores:



$$T_{n+1} = \alpha \cdot t_n + (1 - \alpha) \cdot T_n$$

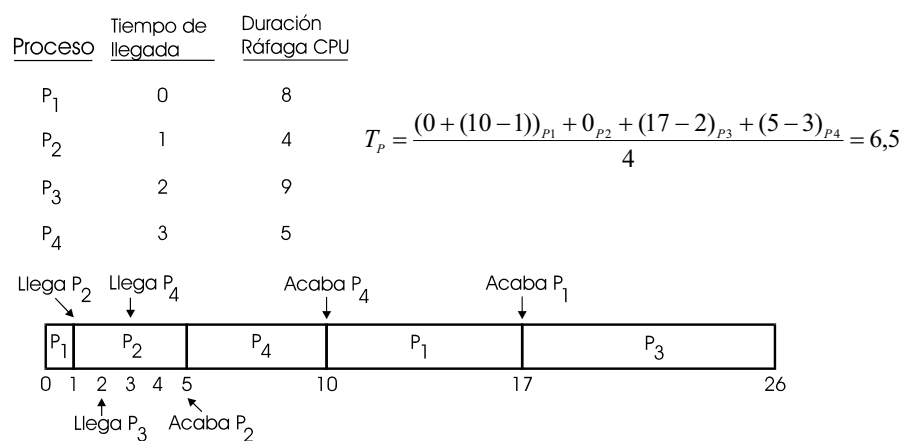
$T_n = \text{Valor previsto para } n_{\text{esima}} \text{ rafaga CPU}$   
 $t_n = \text{Valor } n_{\text{esima}} \text{ rafaga CPU}$   
 $0 \leq \alpha \leq 1$

## Primero el trabajo más corto (SJF).

- La planificación SJF puede ser hasta terminación o expulsiva:
  - ... Llega un proceso A a la cola p.l. con menor ráfaga CPU que tiempo de ejecución de ráfaga le queda a B, proceso en ejecución.
    - Si se ejecuta A: planificación expulsiva.
    - Si continua ejecución B: planificación hasta terminación.

## Primero el trabajo más corto (SJF).

- Ejemplo de planificación SJF expulsiva:



## Prioridades generales.

- Se asigna una prioridad a cada proceso:
  - El de menor prioridad se ejecuta en CPU (si hay dos de igual prioridad FCFS).
    - SJF es un caso particular:  $p=1/T$ .
    - Se asigna números a la prioridad.
- Ejemplo:

Proceso	Duración Ráfaga CPU	Prioridad
P <sub>1</sub>	10	3
P <sub>2</sub>	1	1
P <sub>3</sub>	2	3
P <sub>4</sub>	1	4
P <sub>5</sub>	5	2

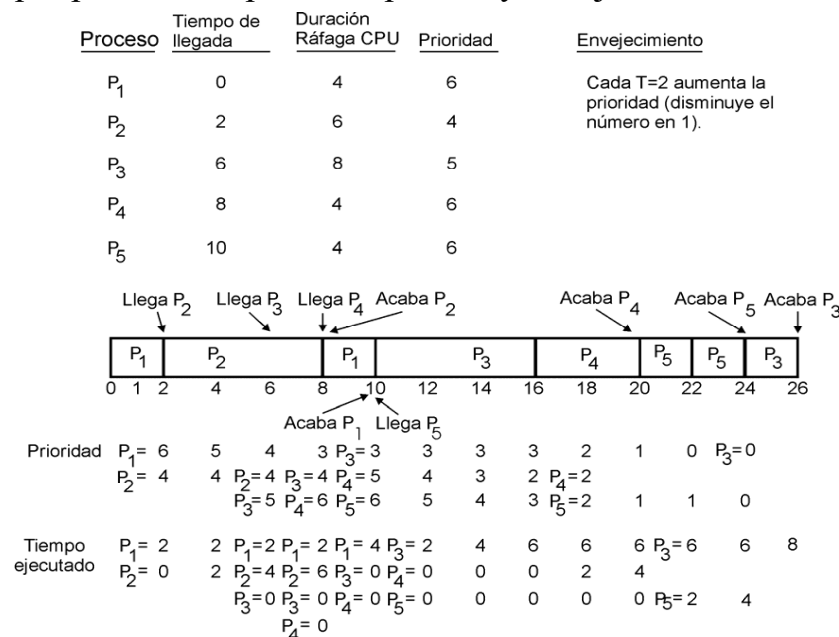
0 1 6 16 18 19

## Prioridades generales.

- Las prioridades pueden definirse:
  - **Factores externos al sistema:** Importancia del proceso, del usuario, etc.
  - **Factores internos al sistema:** Requisitos de memoria, límites de tiempo, número de archivos abiertos, etc.
- La planificación puede ser hasta terminación o expulsiva.
- Problema: Bloqueo indefinido o inanición.
  - Un proceso con muy baja prioridad puede llegar a no ejecutarse nunca.
  - **Solución:** Envejecimiento.
    - Se aumenta la prioridad al aumentar el tiempo de espera en la cola de procesos listos.

## Prioridades generales.

- Ejemplo prioridades planif. expulsiva y envejecimiento:



## Turno rotativo (Round Robin).

- Turno rotativo (RR, Round Robin):
  - La cola de procesos es circular (a nivel práctico se implementa con una FIFO).
  - El planificador la recorre y asigna un tiempo máximo de CPU (Q cuanto de tiempo) a cada proceso .
- Un proceso puede abandonar la CPU:
  - Libremente, si ráfaga de CPU < Q.
  - Después de una interrupción, si ráfaga de CPU > Q.
- Características:
  - Esquema de planificación expulsiva.
  - Si hay n procesos en cola, tiempo espera máximo entre dos ejecuciones (n-1)·Q.
  - Tiempo promedio de espera bastante grande.
  - Diseñado para sistemas de tiempo compartido (equidad).

## Turno rotativo (Round Robin).

- Ejemplo:

Proceso	Duración Ráfaga CPU	Cuanto de tiempo $Q = 4$	$T_p = \frac{(0+4+7+(10-4))}{3} = 5,66$
P <sub>1</sub>	24		
P <sub>2</sub>	3		
P <sub>3</sub>	3		

Diagram illustrating the execution of processes P<sub>1</sub>, P<sub>2</sub>, and P<sub>3</sub> under Round Robin scheduling with a quantum of 4. The timeline shows P<sub>1</sub> running from 0 to 4, 7 to 10, and 14 to 22. P<sub>2</sub> runs from 4 to 7. P<sub>3</sub> runs from 7 to 10.

- Rendimiento: Depende del tamaño del cuanto.
  - Si  $Q \gg$  se convierte en FCFS.
  - Si  $Q \ll$  se reparte el tiempo equitativamente (efecto cambio contexto: ineficiencia).
  - 80% ráfaga menores que  $Q$ .

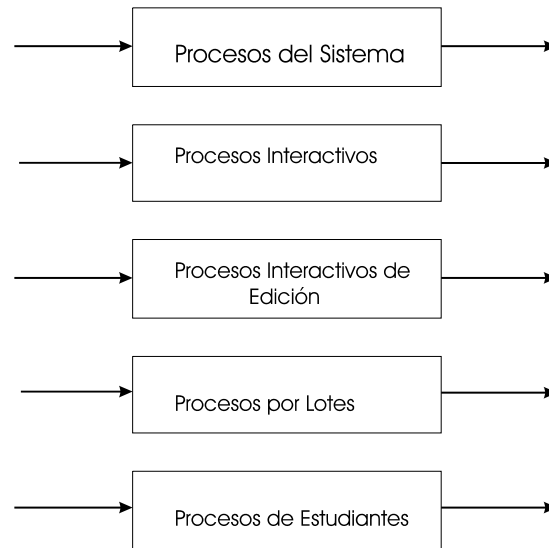
## Colas multinivel con y sin realimentación.

- Los procesos pueden clasificarse en grupos:
  - Primer plano (interactivos).
  - Segundo plano (por lotes).
- Podemos usar colas distintas:
  - Una para cada grupo, con prioridades distintas y algoritmos de planificación distintos.
  - Ejemplo: RR para primer plano, FCFS para segundo plano.
- Cada nuevo proceso se incluirá en una cola y permanecerá en ella.
- Debe existir una planificación entre las colas.
  - Ejemplo:
    - Prioridad absoluta de la cola de primer plano sobre la de segundo.
    - Reparto de la CPU: 80 % primer plano, 20% segundo plano.



## Colas multinivel con y sin realimentación.

- Ejemplo:

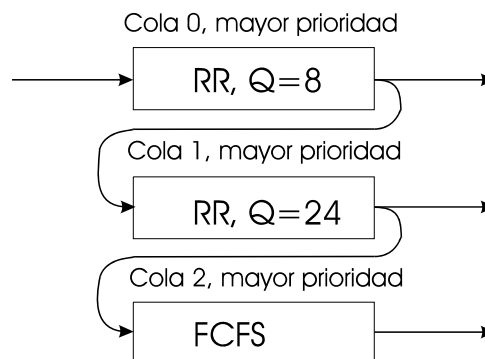


## Colas multinivel con y sin realimentación.

- Planificación de colas múltiples con realimentación:
  - Además de incluir varias colas podemos permitir que los procesos cambien de cola.
  - Se puede, por ejemplo:
    - Ir cambiando a los procesos interactivos o de ráfagas de CPU cortas a colas de mayor prioridad.
    - Cambiar los procesos de ráfagas CPU largas a las colas de menor prioridad.
    - Si un proceso espera demasiado en una cola se puede mover a otra de mayor prioridad (envejecimiento, bloqueo indefinido).

## Colas multinivel con y sin realimentación.

- Ejemplo:



- El planificador:

- Ejecuta primero los procesos cola 0, si vacía cola 1, si vacía cola 2.
- Proceso nuevo se coloca en cola 0.
- Si la duración de ráfaga de un proceso de la cola 0 es mayor que  $Q=8$ , se pasa a cola 1.
- Si la duración de ráfaga de un proceso de cola 1 es mayor que  $Q=24$ , se pasa a cola 2.

## Colas multinivel con y sin realimentación.

- Para definir un planificador de colas múltiples con realimentación necesitamos:

- El número de colas.
- El algoritmo de planificación para cada cola.
- El método para saber cuando cambiar un proceso a otra cola de mayor o menor prioridad.
- El método para determinar a que cola entra un proceso nuevo.

## Planificación de varias CPUs.

- El problema de planificación es más complejo.
- Según el tipo de procesadores:
  - Distintos (sistema heterogéneo):
    - Cada procesador tiene su propia cola y algoritmo de planificación.
  - Idénticos (sistema homogéneo): Pueden compartir cargas.
    - Una cola distinta para cada procesador:
      - Unas más llenas que otras?.
    - Una cola común:
      - Cada procesador se planifica a sí mismo (multiprocesamiento simétrico):
        - Mira la cola común y selecciona un proceso.
        - Problemas: Todos accediendo a la misma cola.
      - Un procesador planifica a los demás (multiprocesamiento asimétrico).

## Evaluación de Algoritmos.

- ¿Cómo seleccionar un algoritmo en un sistema?
  - Decidir un criterio o criterios a utilizar.
- Varios métodos de evaluación:
  - *Evaluación analítica*:
    - Obtención de una fórmula o un número que evalúe el rendimiento.
    - A partir de:
      - El criterio.
      - El algoritmo.
      - La carga de trabajo del sistema.
    - Modelado determinista:
      - Se tome una carga de trabajo determinada.
    - Modelado de colas:
      - Se usa una distribución de ráfagas de CPU y E/S.
      - Se usa otra de tiempos de llegada de procesos.
  - *Simulaciones*.