

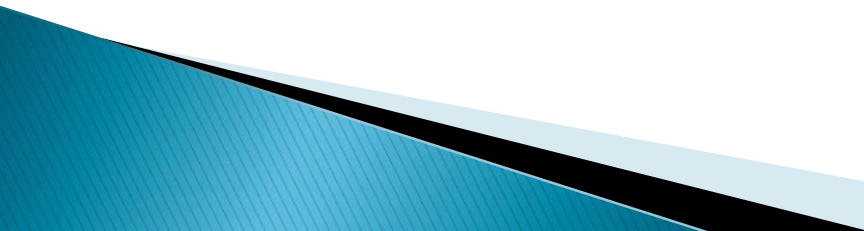
Arquitectura y sistemas operativos

UTN – Mar del Plata

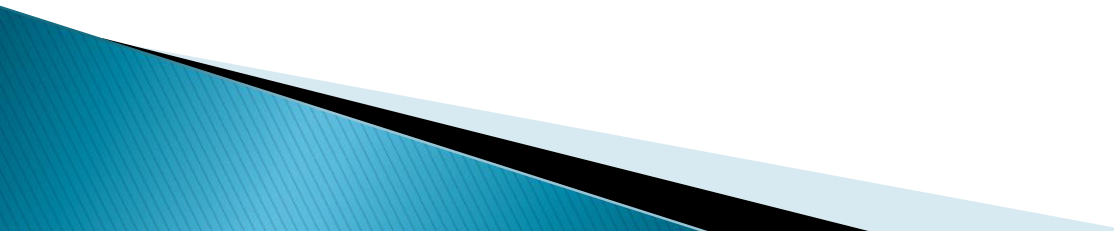
Planificación

- ▶ El planificador (scheduler) del sistema operativo decide que algoritmo de planificación utilizar (FCFS, SJF, Prioridades, Round Robin).
- ▶ Los algoritmos de planificación se utilizan cuando hay 2 o mas procesos listos para su ejecución.

Colas de planificación

- ▶ El sistema operativo usa una serie de colas para planificar los recursos. Estas pueden ser las siguientes:
 - ▶ Cola de trabajos: procesos de almacenamiento secundario esperando memoria principal.
 - ▶ Cola de procesos listos: procesos en memoria, listos y esperando su ejecución.
 - ▶ Cola de dispositivos: para cada dispositivo hay una cola de procesos esperando utilizarlo.
- 

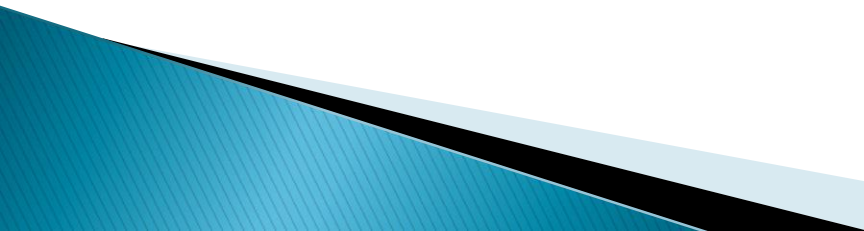
Funciones del planificador de procesos

- ▶ 1. Llevar control del estado de cada proceso.
 - ▶ 2. Decidir que proceso usa el procesador y durante cuanto tiempo, para ello empleara un cierto criterio en base al cual tomar las decisiones.
 - ▶ 3. Asignar el procesador al proceso.
 - ▶ 4. Quitar el procesador al proceso.
- 

Tipos de planificación

- ▶ **Largo plazo:** cuando se crea un proceso se puede decidir alguno de los criterios para su planificación, como por ejemplo la prioridad o quantum (tiempo máximo que se permite a un proceso el uso del procesador).
- ▶ **Corto plazo:** cada vez que un proceso abandona la CPU, toma la decisión de que proceso planificar en función de la política de planificación establecida y del valor de los parámetros planificados.
- ▶ **Mediano plazo:** otras partes del sistema operativo pueden intervenir en la planificación de forma indirecta (swap), al sacar un proceso de memoria, etc hace que este no sea planificable.

Evaluación de rendimiento de planificación

- ▶ **Equidad:** procesan la CPU de forma equitativa.
 - ▶ **Eficiencia:** utilización del CPU al 100%.
 - ▶ **Tiempo de Retorno/Finalización/Ejecución:** tiempo que tarda en ejecutarse un proceso en concreto.
 - ▶ **Tiempo de respuesta:** minimizar el tiempo de atención para usuarios interactivos.
 - ▶ **Tiempo de espera:** tiempo que un proceso espera en cola de procesos listos.
 - ▶ **Rendimiento (productividad):** numero de trabajos procesados por unidad de tiempo.
- 

Tipos de Planificación

- ▶ **No apropiativo:** el proceso en ejecución conserva el uso de la CPU mientras lo desee.

Ejemplo: FCFS, SJF y Prioridades

- ▶ **Apropiativo:** el sistema operativo puede expulsar a un proceso de la CPU.

Ejemplo: Prioridades, SRTF y Round Robin.

Algoritmo – FCFS

- ▶ FCFS – First come, first server
- ▶ El primer proceso que entro en la cola de procesos listos es el primero al que se le asigna CPU.
- ▶ Se implementa con una cola FIFO (first in, first out).
- ▶ Es un algoritmo del tipo ejecución hasta terminación.
- ▶ Sensible al orden de llegada de los procesos.

Algoritmo – FCFS Ejemplo

Proceso	Duración								
P1	9	0		9		13		15	
P2	4		P1			P2		P3	
P3	2								

Tiempo de espera: $P1=0, P2=9, P3=13$ / $Cent. P = 7,3 + .ESPERA MEDIO$

Tiempo de retorno: $P1=9, P2=13, P3=15$

Tiempo espera medio: $(0+9+13) / 3 = 7,3$

Productividad: 3 (cantidad de procesos) / 15 (tiempo finalización de todos los procesos)

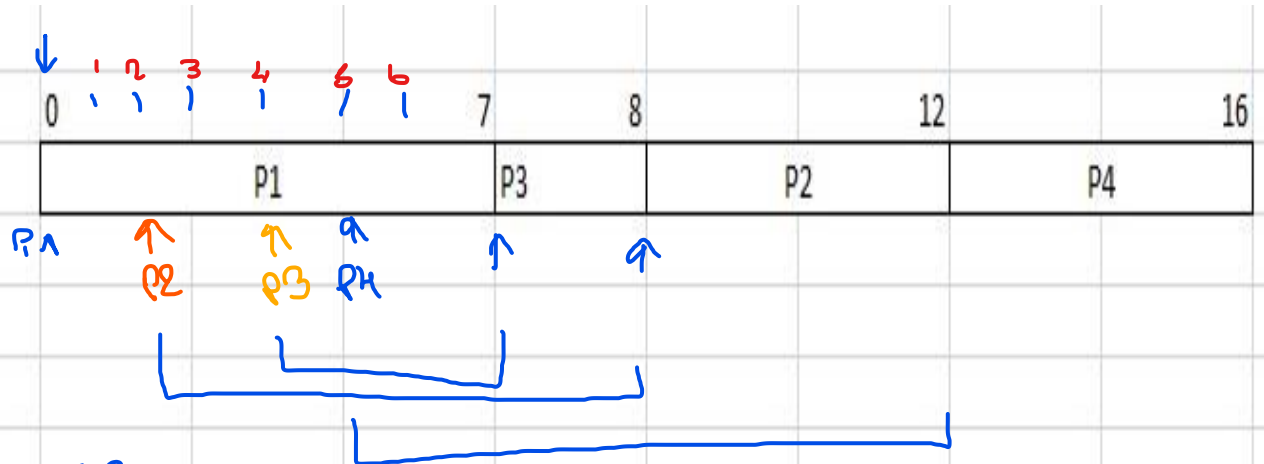
Como se puede mejorar el tiempo de espera medio??

Algoritmo – SJF

- ▶ SJF – **Short Job First** (Primero el trabajo corto).
- ▶ 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 2 con igual longitud de ráfaga se usa FCFS.**
- ▶ **Solamente se puede aplicar si se conoce de antemano la duración de cada trabajo.**
- ▶ **Posibilidad de inanición –> si continuamente llegan trabajos cortos, los trabajos largos nunca llegan a ejecutarse.**

Algoritmo – SJF Ejemplo

Proceso	Llegada	Duración
P1	0	7
P2	2	4
P3	4	1
P4	5	4



Tiempo espera:

$$P1 = (0 \text{ Inicio proceso} - 0 \text{ Llegada proceso}) = 0$$

$$P2 = (8 \text{ Inicio proceso} - 2 \text{ Llegada proceso}) = 6$$

$$P3 = (7 \text{ Inicio proceso} - 4 \text{ Llegada proceso}) = 3$$

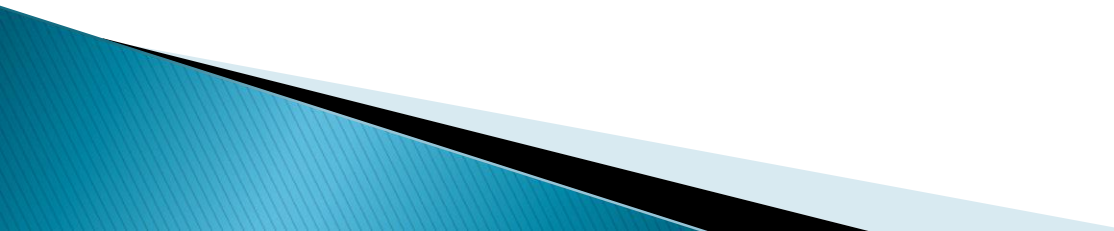
$$P4 = (12 \text{ Inicio proceso} - 5 \text{ Llegada proceso}) = 7$$

$$4 / 16 = 0,25 \rightarrow P$$

$$\text{Tiempo espera medio} = (0 + 6 + 3 + 7) / 4 = 4$$

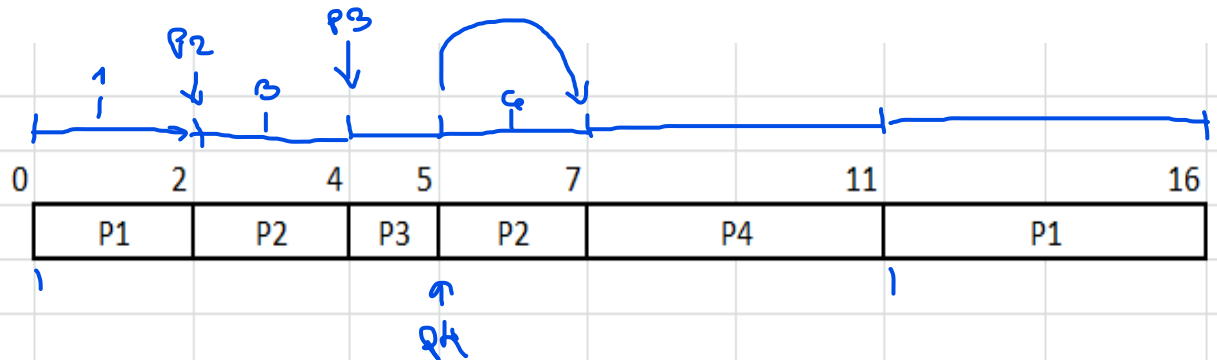
Costo . P

Algoritmo – SRTF

- ▶ SRTF (Shortest Remaining Time First) Primero el menor tiempo restante.
 - ▶ El planificador siempre escoge el proceso que tiene el menor tiempo de ejecución restante esperado.
 - ▶ El planificador podría expulsar al proceso actual cuando llega un nuevo proceso con menor ráfaga.
 - ▶ El planificador debe tener una estimación del tiempo de proceso para realizar la función seleccionada, y existe riesgo de inanición para los procesos más largos.
- 

Algoritmo – SRTF Ejemplo

Proceso	Llegada	Duración
P1	0	5
P2	2	4
P3	4	1
P4	5	4



Tiempo de respuesta:

$$P1 = (0 \text{ Inicio proceso} - 0 \text{ llegada proceso}) = 0$$

$$P2 = (2 \text{ Inicio proceso} - 2 \text{ llegada proceso}) = 0$$

$$P3 = (4 \text{ Inicio proceso} - 4 \text{ llegada proceso}) = 0$$

$$P4 = (7 \text{ Inicio de proceso} - 5 \text{ instante de llegada}) = 2$$

$$\text{Tiempo respuesta medio} = (0 + 0 + 0 + 6) / 4 = 1,5$$

P1 0
 P2 0
 P3 0
 P4 2

r.m.e
 0,5

Algoritmo – Prioridades

- ▶ Se asigna una prioridad a cada proceso.
- ▶ El de menor prioridad se ejecuta en CPU, si hay 2 procesos de igual prioridad se utiliza FCFS.
- ▶ Se asigna números a la prioridad.

Algoritmo – Prioridades ejemplo

10
20
30
40
50

Proceso	Duracion	Prioridad
P1	10	3
P2	1	1
P3	2	4
P4	1	5
P5	5	2

← 1

↓ P1

↓

0	1	6	16	18	19
P2	P5	P1	P3	P4	

Tiempo medio: **Respuesta**

↓

$$P1 = (16 \text{ fin del proceso} - 10 \text{ duracion}) = 6$$

$$P2 = (1 \text{ fin del proceso} - 1 \text{ duracion}) = 0$$

$$P3 = (18 \text{ fin del proceso} - 2 \text{ duracion}) = 16$$

$$P4 = (18 \text{ fin del proceso} - 1 \text{ duracion}) = 18$$

$$P5 = (6 \text{ fin del proceso} - 5 \text{ duracion}) = 1$$

T.E

6

0

6

18

1

T.R

16

1

18

19

6

T.M.E

$$\frac{6 + 0 + 16 + 18 + 1}{5} = 8,2$$

$$\text{Tiempo espera medio} = (6 + 0 + 16 + 18 + 1) / 5 = 8,2$$

$$\text{Tiempo medio retorno} = (16 + 1 + 18 + 19 + 6) / 5 = 12$$

Algoritmo – Round Robin

- ▶ También llamado Turno Rotatorio – Circular.
- ▶ Adecuado para implementar tiempo compartido.
- ▶ Corresponde a FCFS con expropiación.
- ▶ Cada proceso tiene un quantum (cuanto) de tiempo máximo. Si cuando expira el quantum de tiempo el proceso continua en CPU, el planificador lo desaloja y lo ingresa al final de la cola de listos.
- ▶ Un proceso puede abandonar la CPU: libremente (si ráfaga de CPU < quantum) o después de interrupción (si ráfaga de CPU > quantum).

Algoritmo – Round robin ejemplo

Proceso	Duración
P1	24
P2	3
P3	3

quantum = 4

CPU

6
1
4
I

L
P1
P2
P3
P1 20
P4 5
P1 16
P4 1
P1 12

0 4 7 10 30

P1 P2 P3 P1

Tiempo espera medio: $(0 + 4 + 7 + (10 - 4)) / 3 = 5,66$

P1 P2 P3
6
P1