

Gerência do Processador



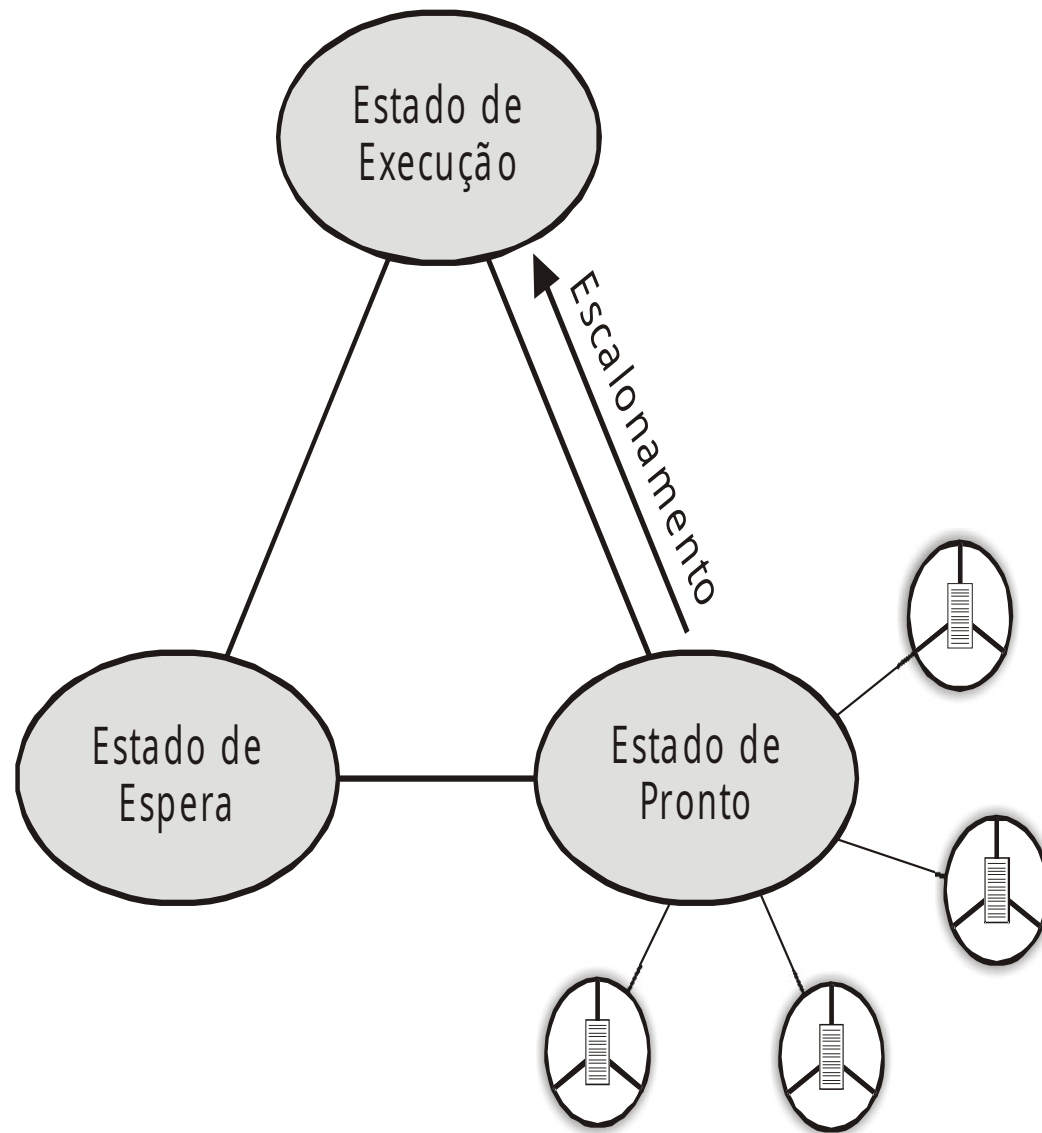
UNIFEI

Universidade Federal de Itajubá

IMC – Instituto de Matemática e Computação

Prof. Carlos Minoru Tamaki

Escalonamento



Funções Básicas

- Manter o processador ocupado
- Balancear o uso da CPU entre processos
- Privilegiar execução de aplicações críticas
- Maximizar o throughput do sistema
- Oferecer tempo de resposta razoável para usuário interativos
- Cada sistema possui sua política de escalonamento e é denominada escalonador (***scheduler***)
- Outra rotina de gerência é conhecida por ***dispatcher*** → responsável pela troca de contexto. O tempo gasto na troca de contexto é denominado ***latência do dispatcher***.

Critérios de Escalonamento

- Utilização do processador
 - 30% - baixa carga de processamento
 - 90% - muito carregado
- **Throughput**
 - Número de processos em um determinado intervalo de tempo
- **Tempo de processador ou Tempo de CPU**
 - Tempo que um processo leva no estado de execução

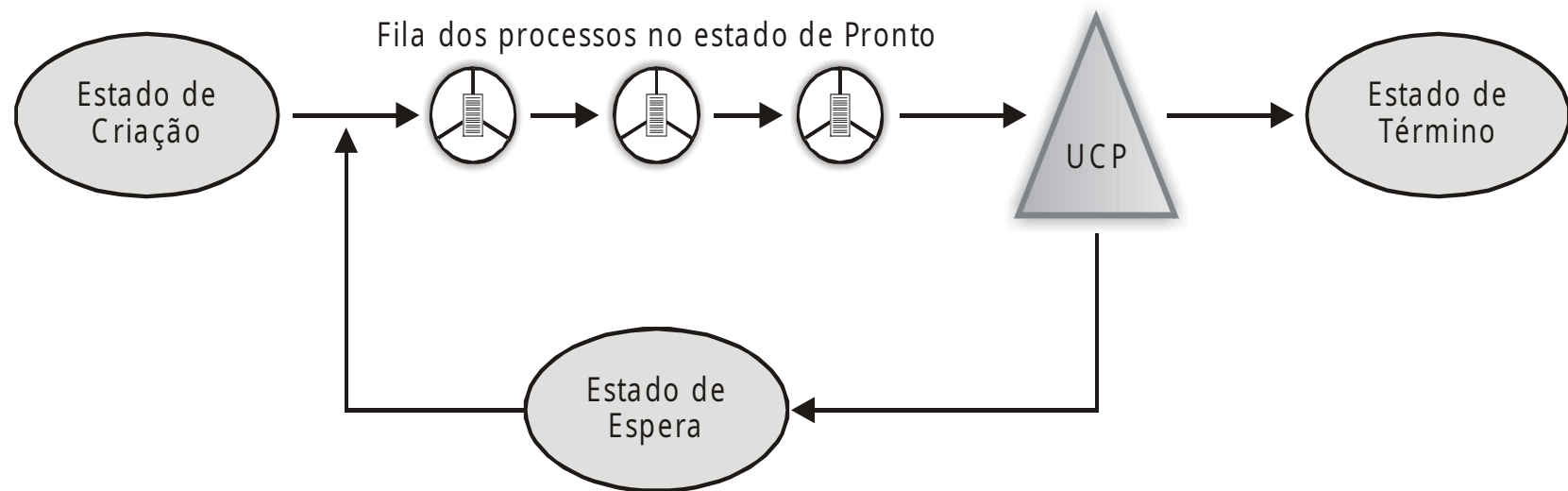
- **Tempo de espera**
 - Tempo total que o processo permanece na fila de pronto durante seu processamento
- **Tempo de Turnaround**
 - Tempo que um processo leva desde a sua criação até seu término
- **Tempo de resposta**
 - Tempo decorrido entre uma requisição ao sistema ou à aplicação e o instante em que a resposta é exibida

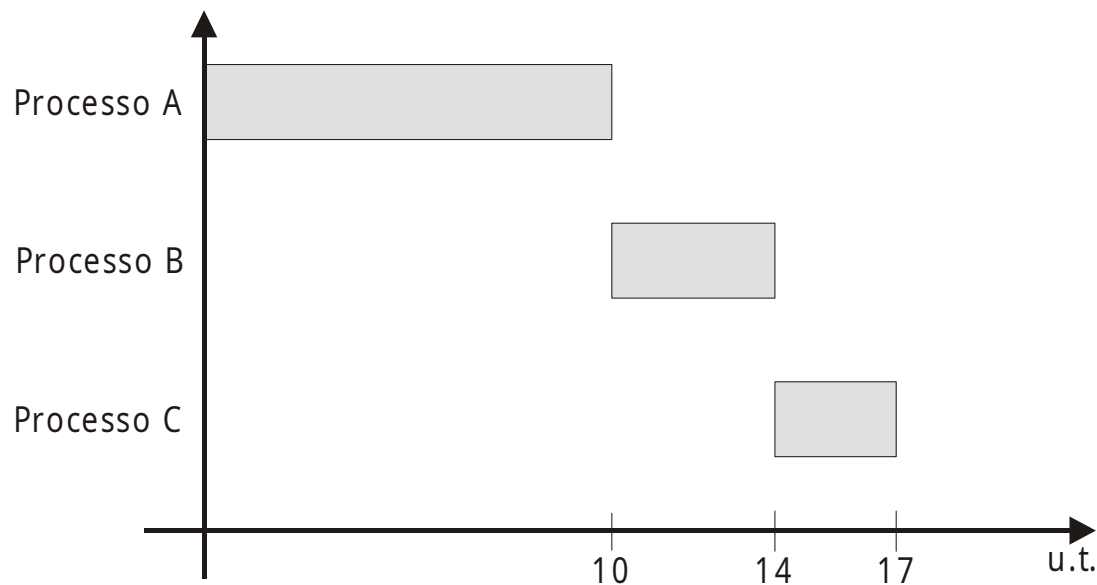
Escalonamentos Não-Preemptivos e Preemptivos

- **Escalonamento não-preemptivo:** predomina no processamento batch → quando um processo está em execução nenhum evento externo pode ocasionar a perda do processador.
- **Escalonamento preemptivo:** o sistema operacional pode interromper um processo em execução e passá-lo para o estado de pronto com o objetivo de alocar outro processo.

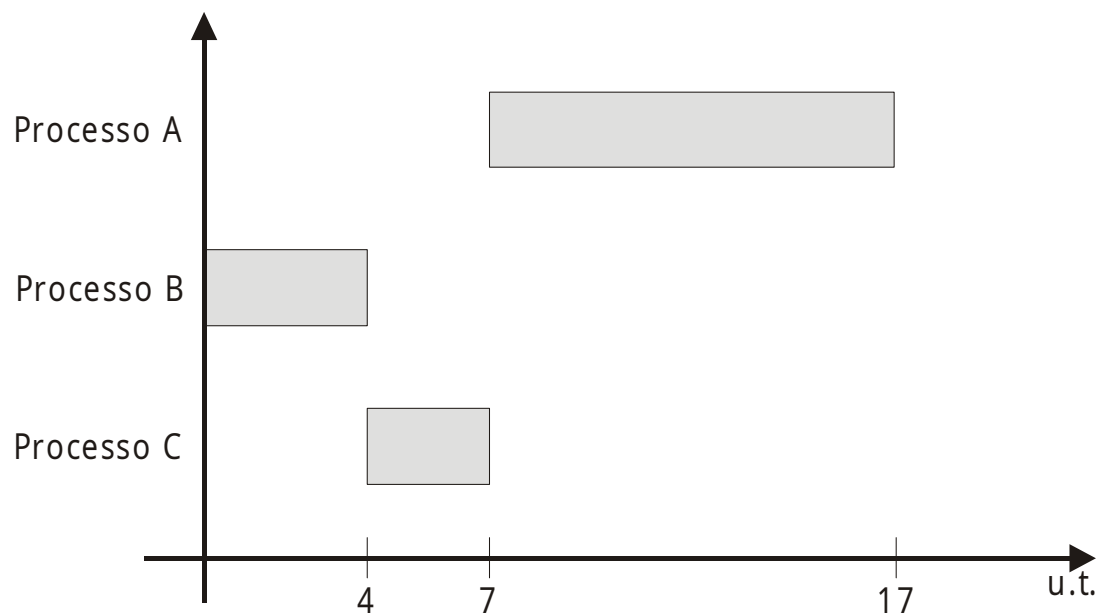
Escalonamento FIFO

- **FIFO scheduling ou FCFS scheduling**
(First Come First Served)





- Tempo médio de espera:**
 $(0+10+14)/3 = 8 \text{ u.t.}$



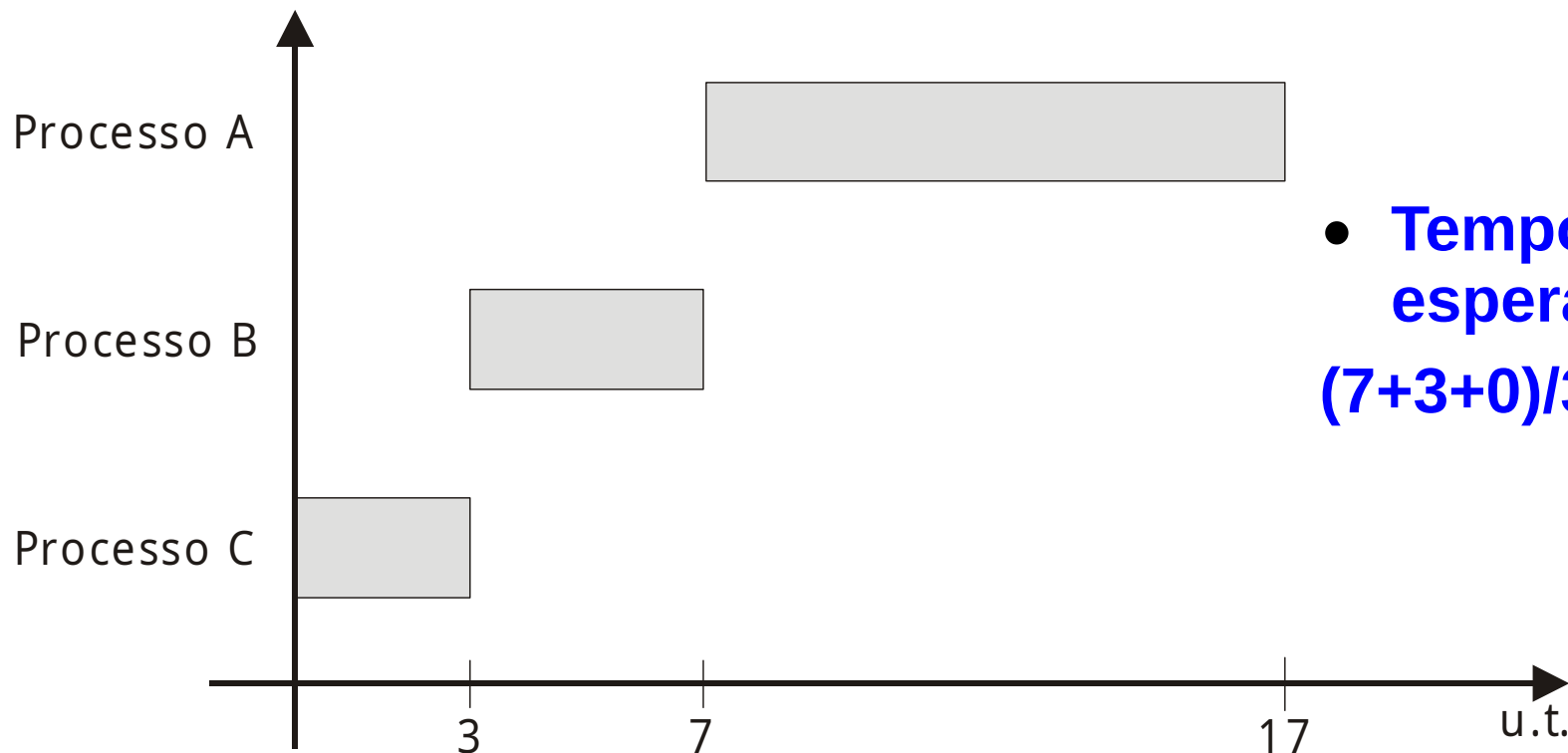
Processo	Tempo de processador (u.t.)
A	10
B	4
C	3

- Tempo médio de espera:**
 $(7+0+4)/3 = 3,7 \text{ u.t.}$

- No escalonamento FIFO não há diferença entre os processos **CPU-Bound** e **I/O-Bound**, então se existir um processo **I/O-Bound** mais importante, não vai ser possível tratar esta diferença.
- Escalonamento FIFO é do tipo não-preemptivo → sistemas mono programáveis com processamento **batch**.

Escalonamento SJF

- Escalonamento **Shortest-job-first** (SJF scheduler) ou **Shortest-Process-Next** (SPN scheduler)



- **Tempo médio de espera:**
 $(7+3+0)/3 = 3,3 \text{ u.t.}$

- Não é possível ao SO saber quanto tempo um processo irá permanecer utilizando a CPU
- O tempo pode ser estimado com base na média exponencial dos tempos passados:

$$\delta_{n+1} = \alpha \cdot t_n + (1 - \alpha) \cdot \delta_n$$

- Onde: t_n é informação mais recente, δ_n é o valor histórico e α é uma constante entre 0 e 1, 0.5 considera valores passados e presentes com mesmo peso.

- O escalonamento SJF é um escalonamento não-preemptivo.
- Sua vantagem sobre o FIFO é o tempo médio de ***turnaround*** dos processos
- É possível ocorrer ***starvation*** dos processos com alto tempo de CPU ou do tipo ***CPU-bound***
- Pode ser implementado com preempção e é denominado escalonamento ***Shortest Remaining Time (SRT scheduling)***

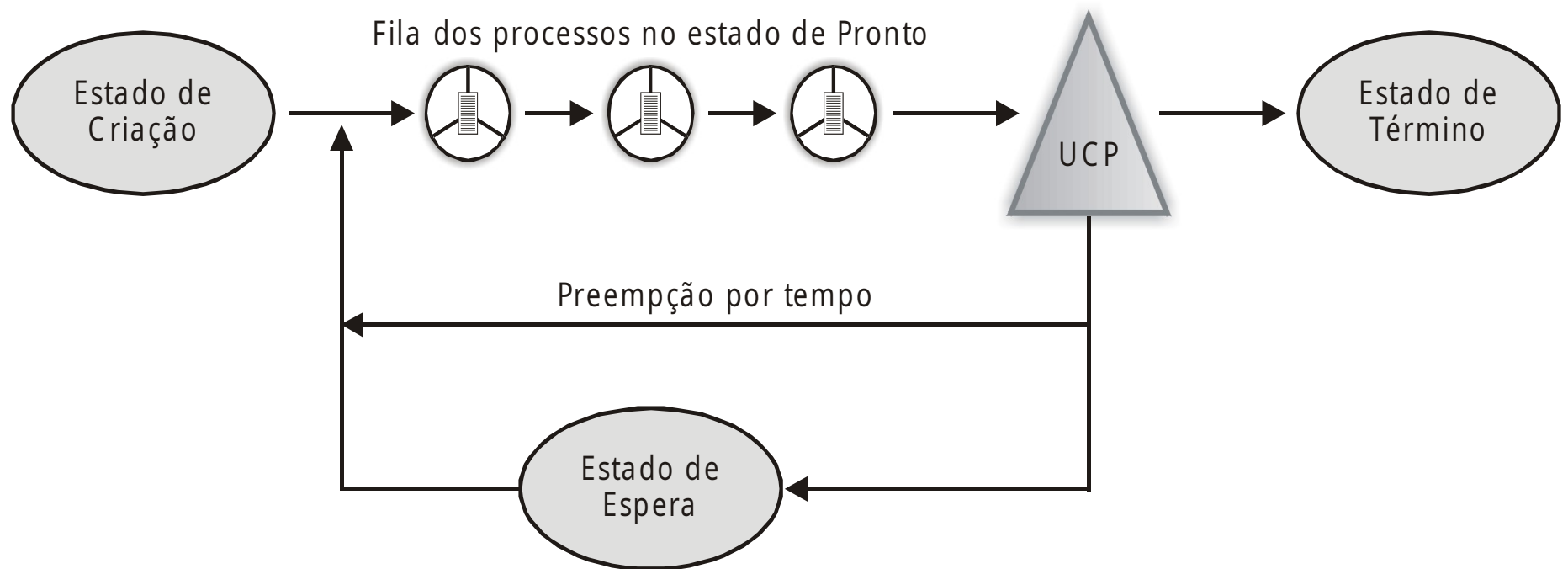
Escalonamento Cooperativo

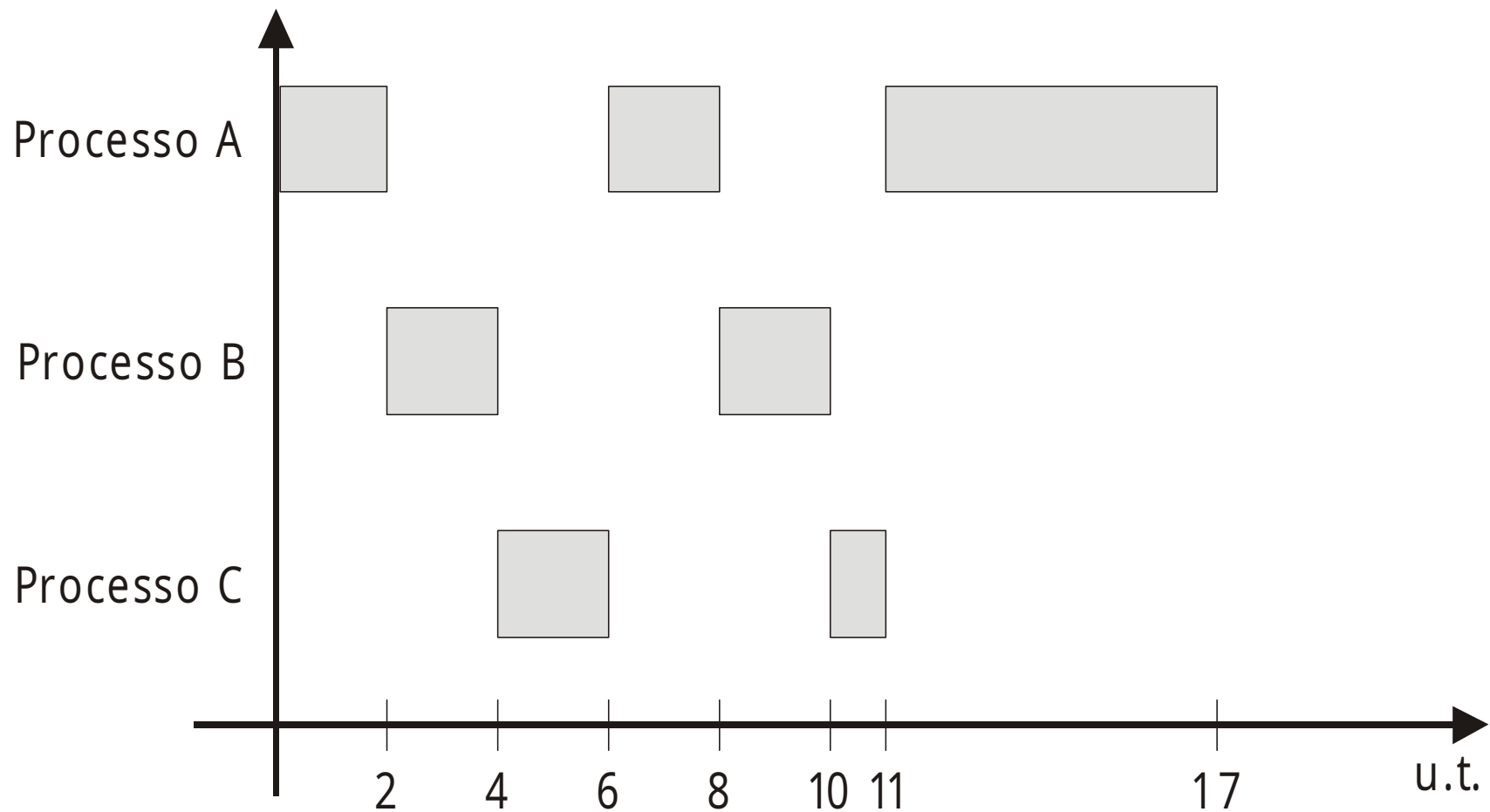
- Aumenta o grau de multiprogramação em políticas de escalonamento não-preemptivos
- A liberação da CPU é uma tarefa do processo em execução
- O processo verifica periodicamente a fila de mensagens
- Exemplo: os primeiros SO's da família Microsoft Windows, até a versão 3.1, conhecidos como multitarefa cooperativa.

Escalonamento Circular

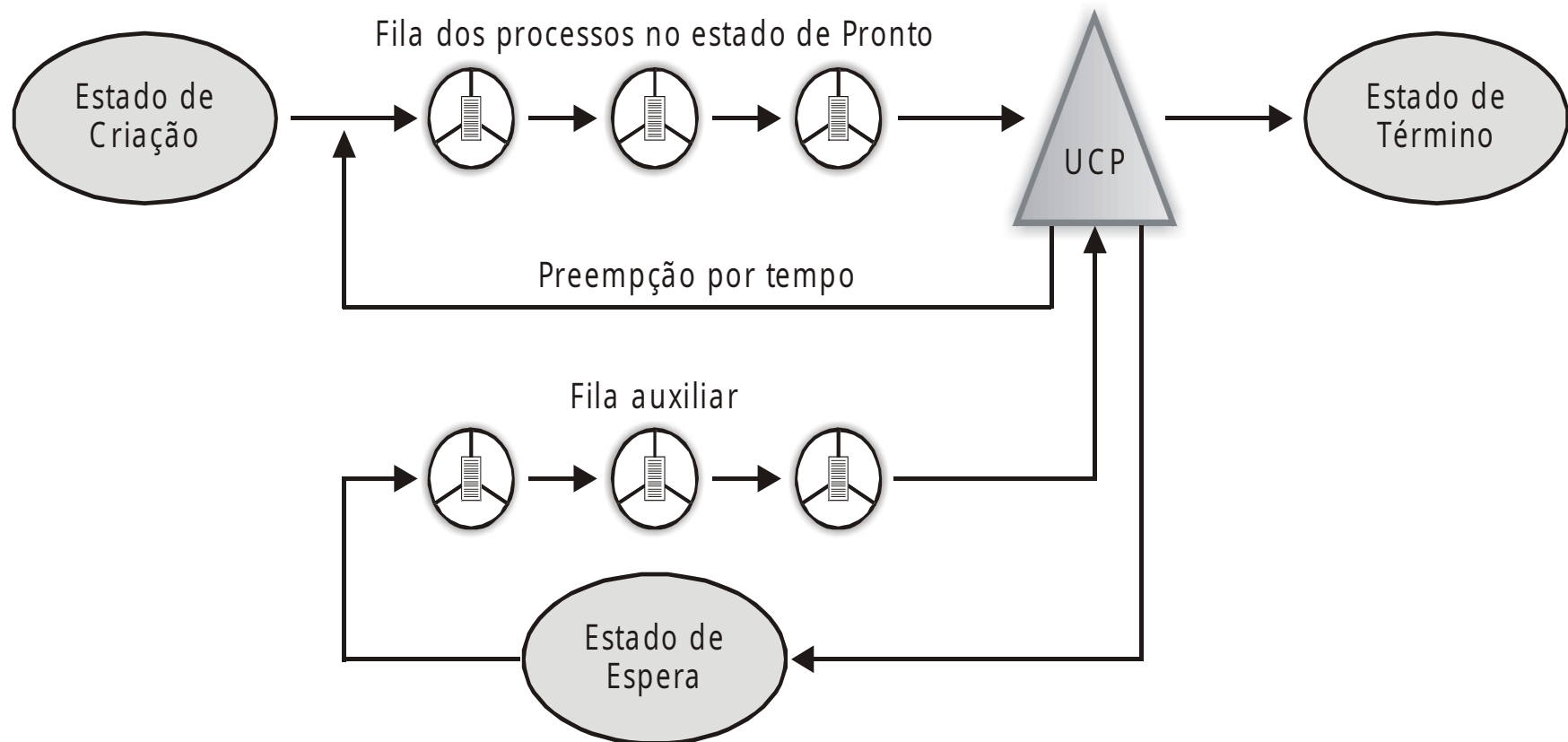
- ***Round Robin Scheduling***
- Escalonamento do tipo preemptivo, projetado para sistemas de tempo compartilhado
- Existe tempo-limite do uso contínuo do processador denominado fatia de tempo (***time-slice***) ou ***quantum*** de tempo
- Preempção por tempo

- Round Robin Scheduling



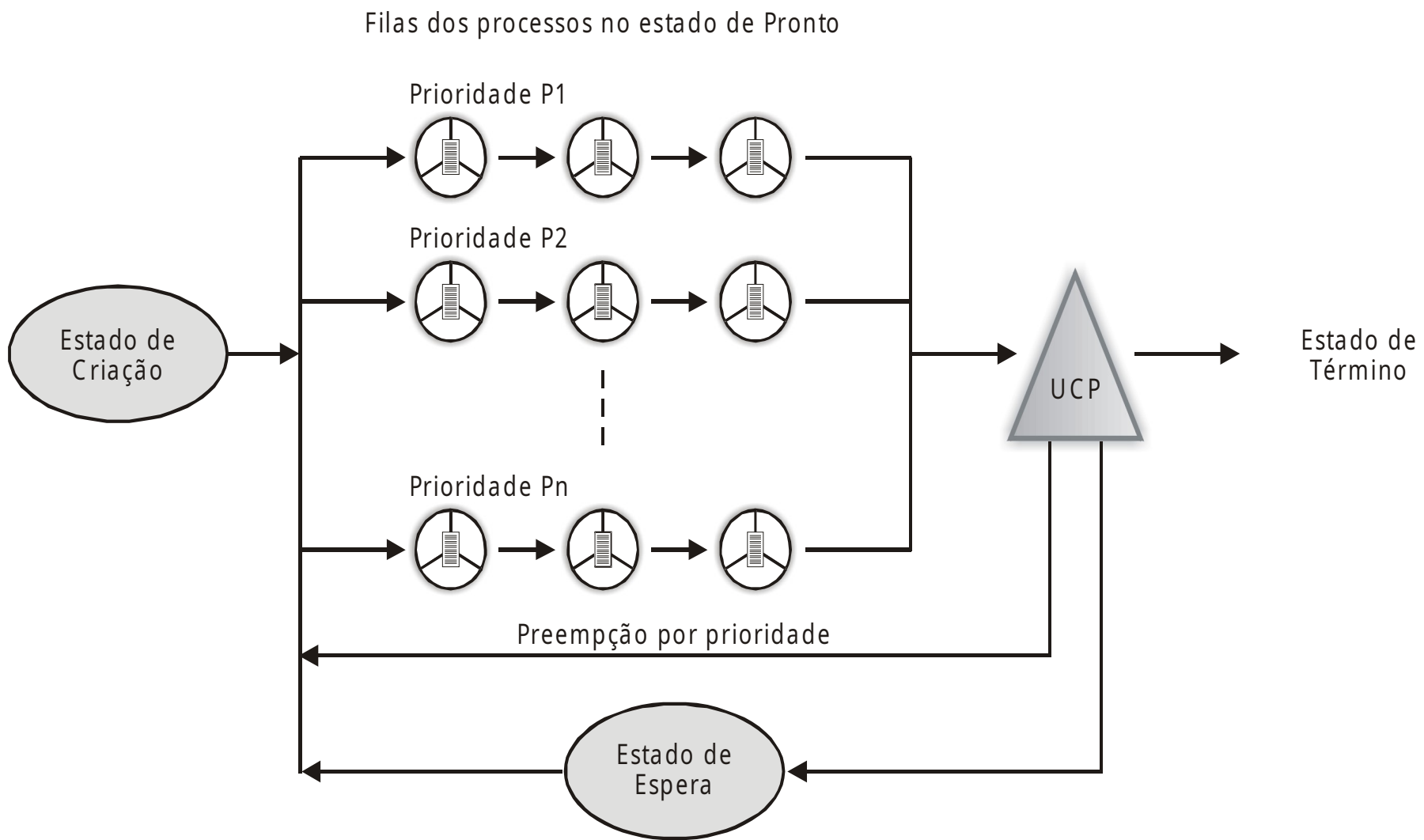


Escalonamento circular virtual

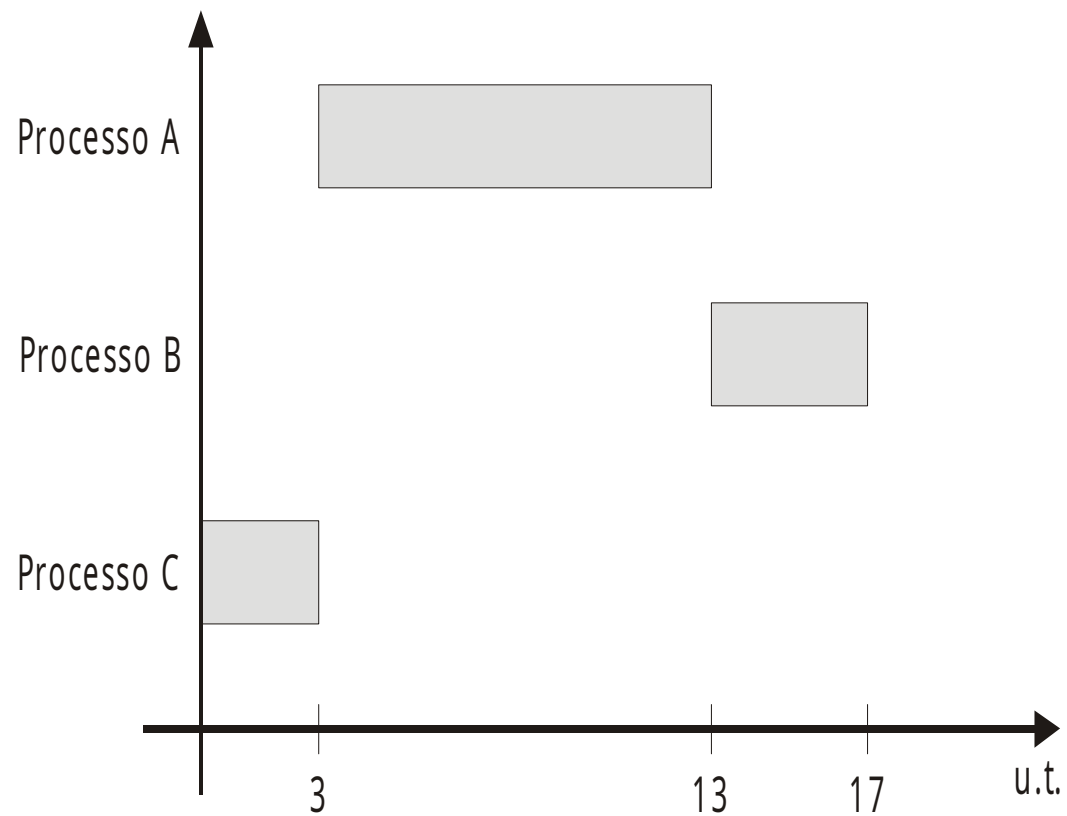


Escalonamento por Prioridades

- Prioridade de execução
- Não existe conceito de fatia de tempo
- Um processo em execução não sofre preempção por tempo
- A perda do processador ocorrerá pela mudança voluntária para o estado de espera ou se um processo de maior prioridade passar para o estado de pronto
- Preempção por prioridade

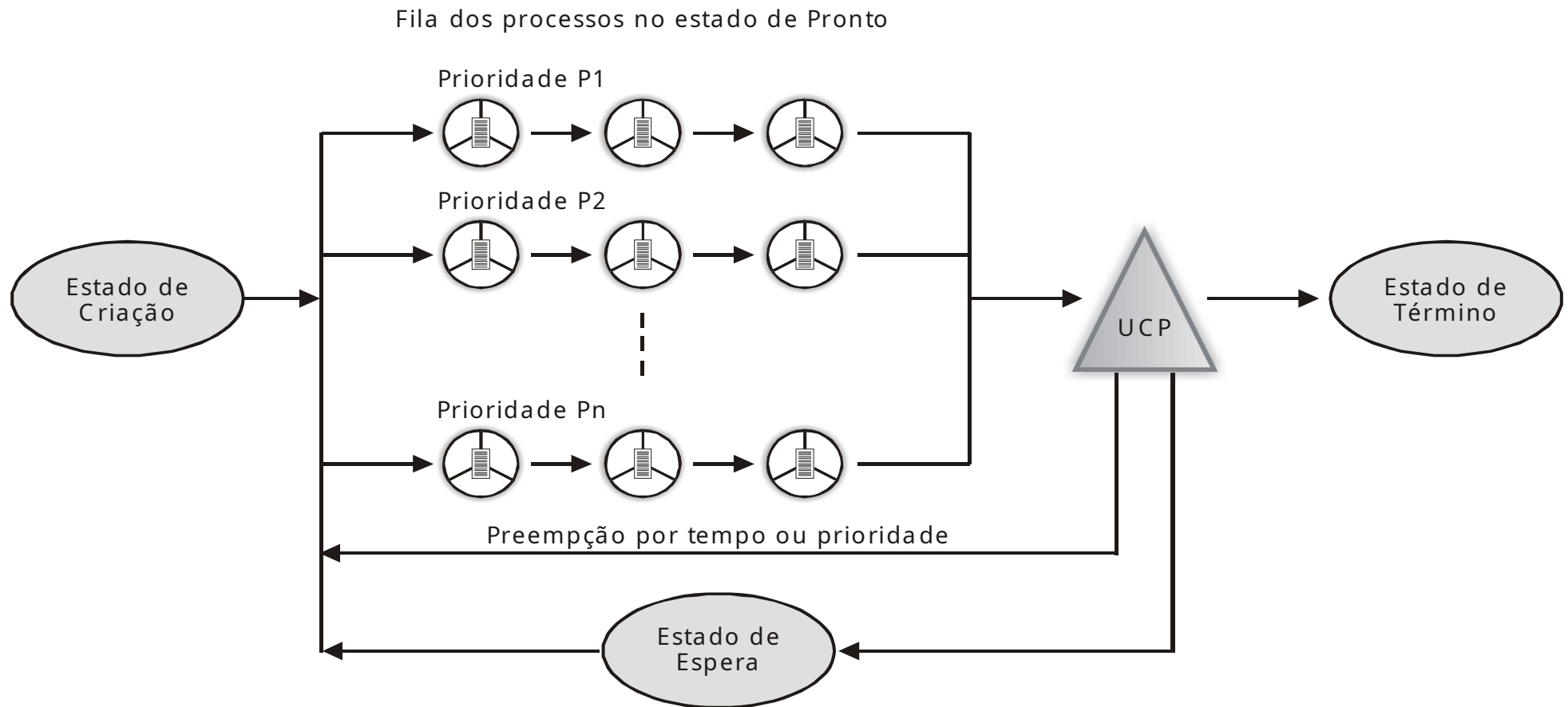


- **Prioridade estática** → o valor da prioridade não é alterada durante a vida do processo
- **Prioridade dinâmica** → o SO pode alterar o valor da prioridade do processo em função do comportamento
- Para evitar o **starvation** → Técnica de **aging**



Processo	Tempo de processador (u.t.)	Prioridade
A	10	2
B	4	1
C	3	3

Escalonamento circular com prioridades

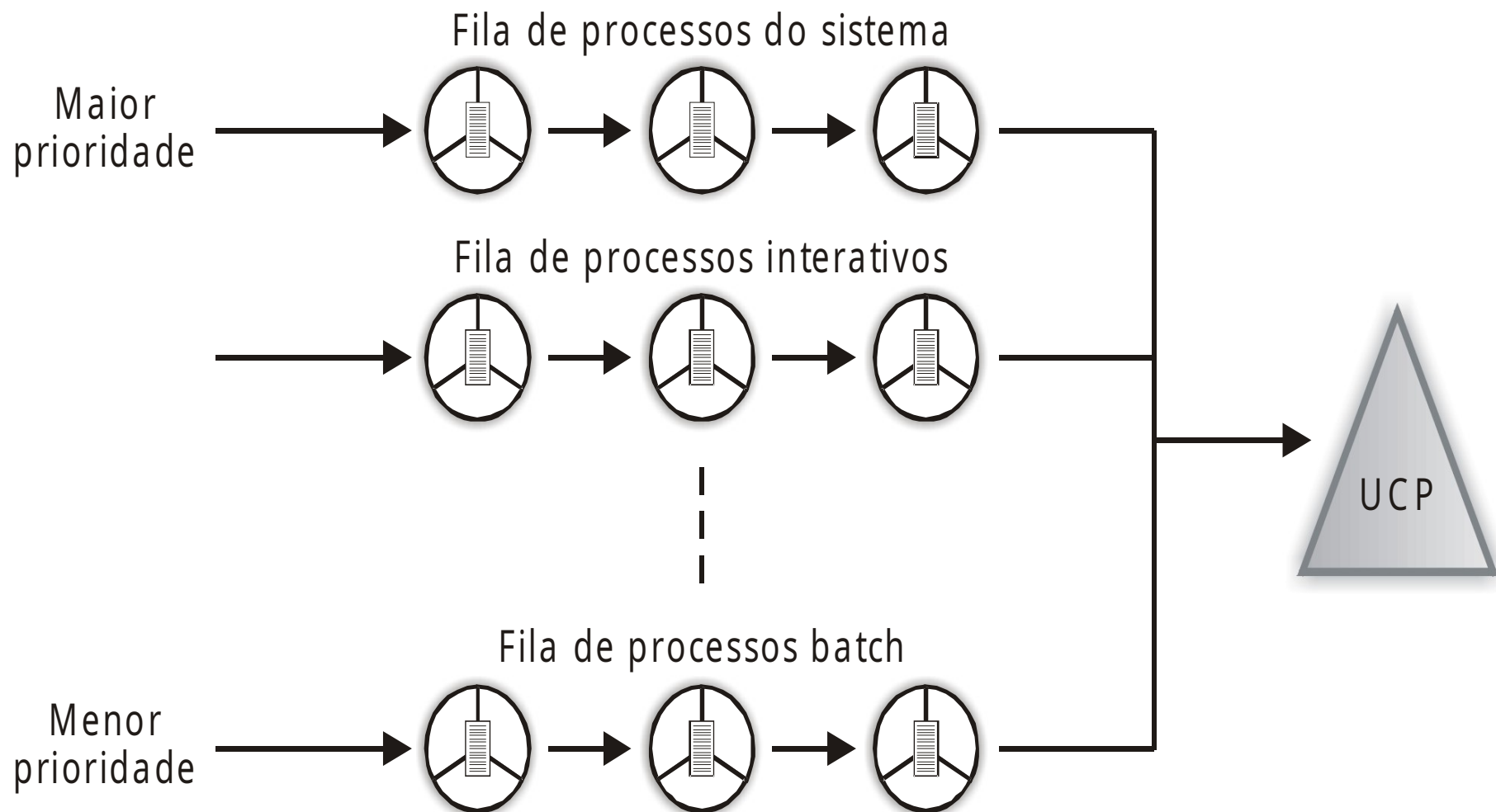


Estática

Dinâmica → administrador do sistema ou SO

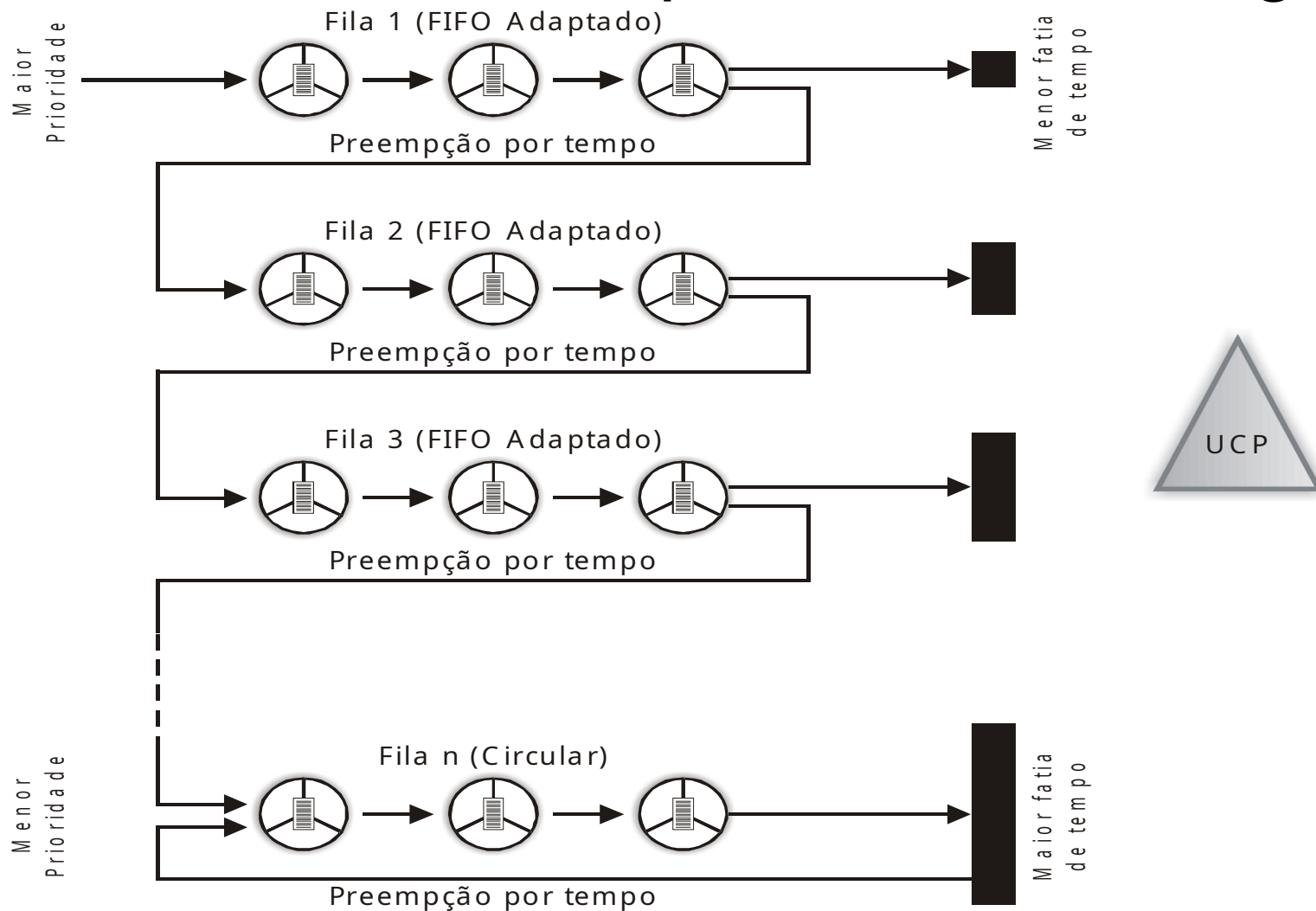
Escalonamento por múltiplas filas

- Multilevel queue scheduling

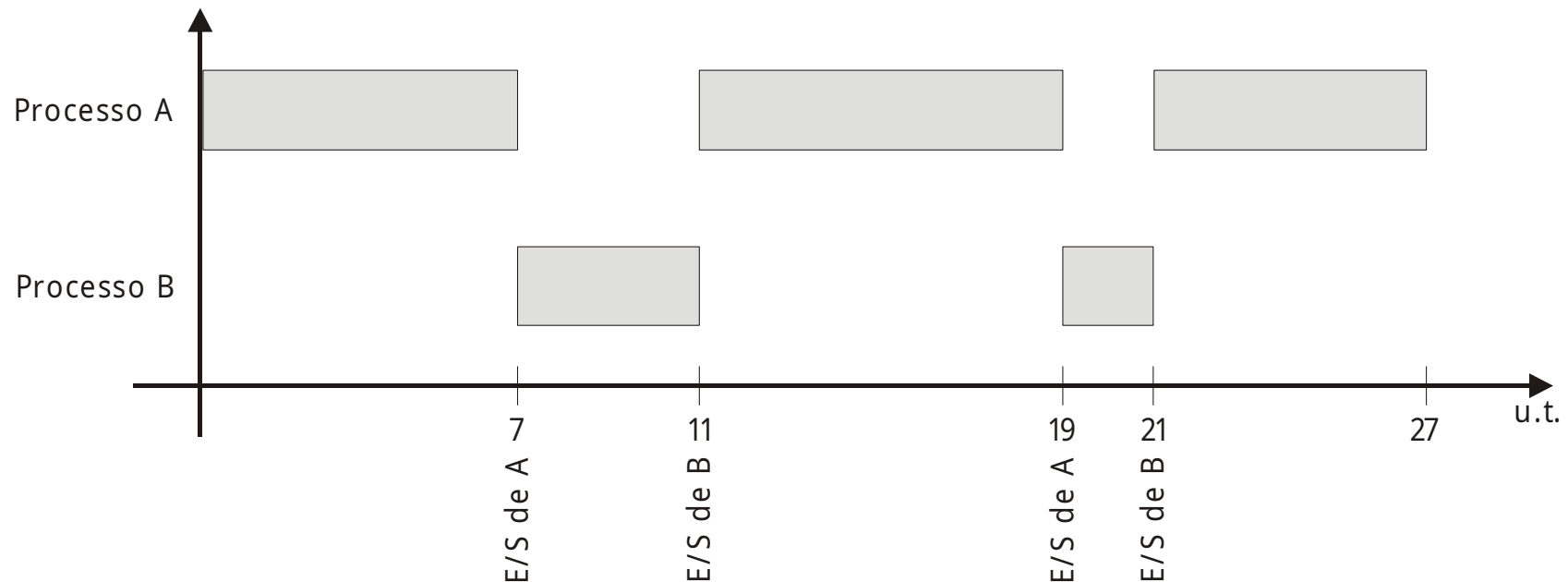


Escalonamento por múltiplas filas com realimentação

- Multilevel feedback queues scheduling

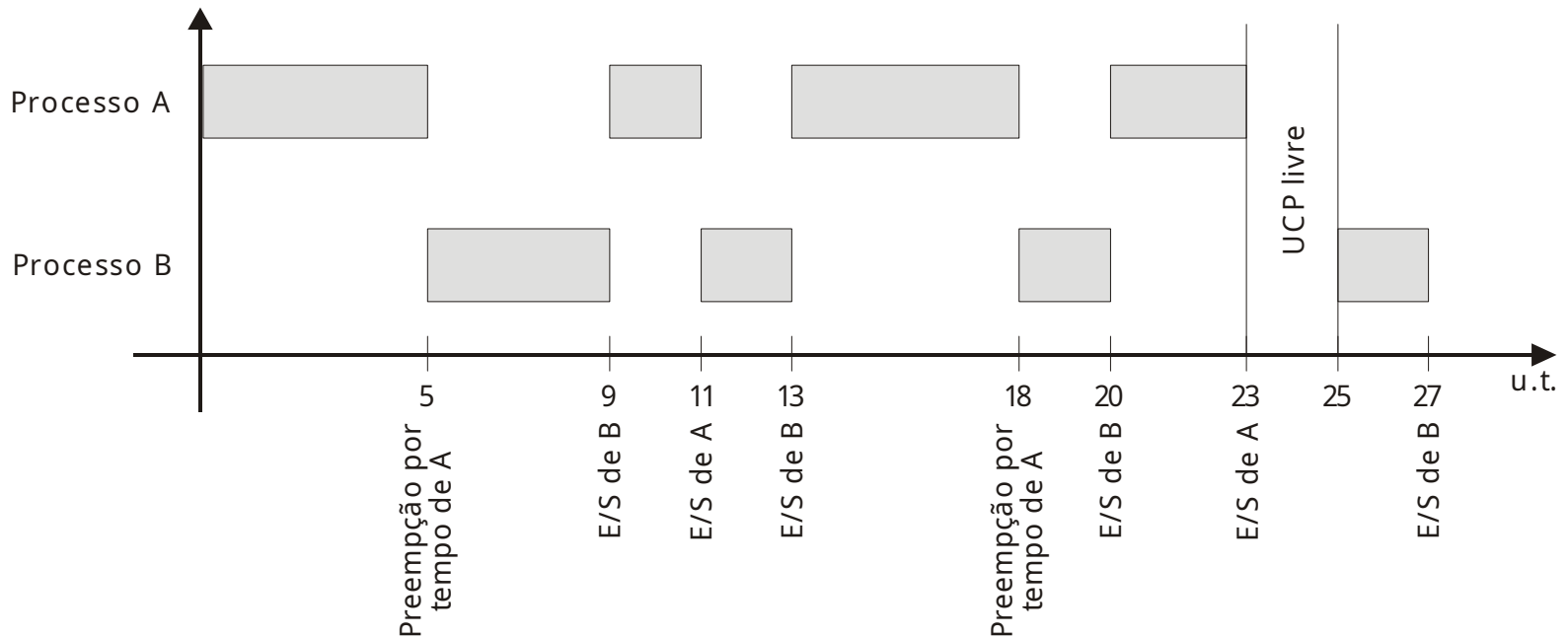


Exemplo de escalonamento FIFO



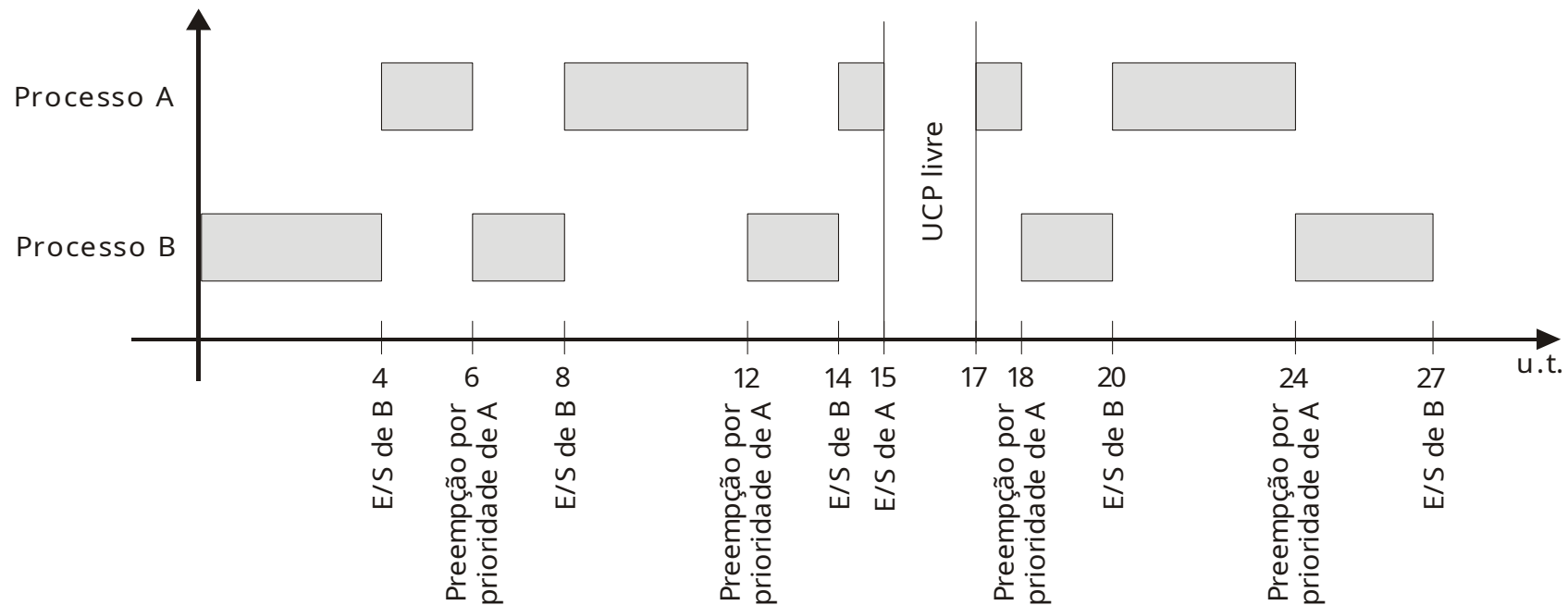
Processo	Tempo de processador (u.t.)	Característica
A	21	CPU-bound
B	6	I/O-bound

Exemplo de escalonamento circular



Processo	Tempo de processador (u.t.)	Característica
A	15	CPU-bound
B	10	I/O-bound

Exemplo de escalonamento circular com prioridades



Processo	Tempo de processador (u.t.)	Característica	Prioridade
A	12	CPU-bound	Baixa
B	13	I/O-bound	Alta

Política de Escalonamento em Sistemas de Tempo Real

- Controle de produção de bens industriais e controle de tráfego aéreo
- Importância relativa de cada tarefa na aplicação
- Escalonamento por prioridades é a mais adequada
- Não deve existir o conceito de fatia de tempo
- Prioridade estática
- OpenVMS

Dúvidas?