

Sistemas Operativos

Licenciatura em Engenharia Informática
Licenciatura em Engenharia Computacional
Licenciatura em Física (opcional)

Ano letivo 2025/2026

Pedro Azevedo Fernandes (paf@ua.pt)

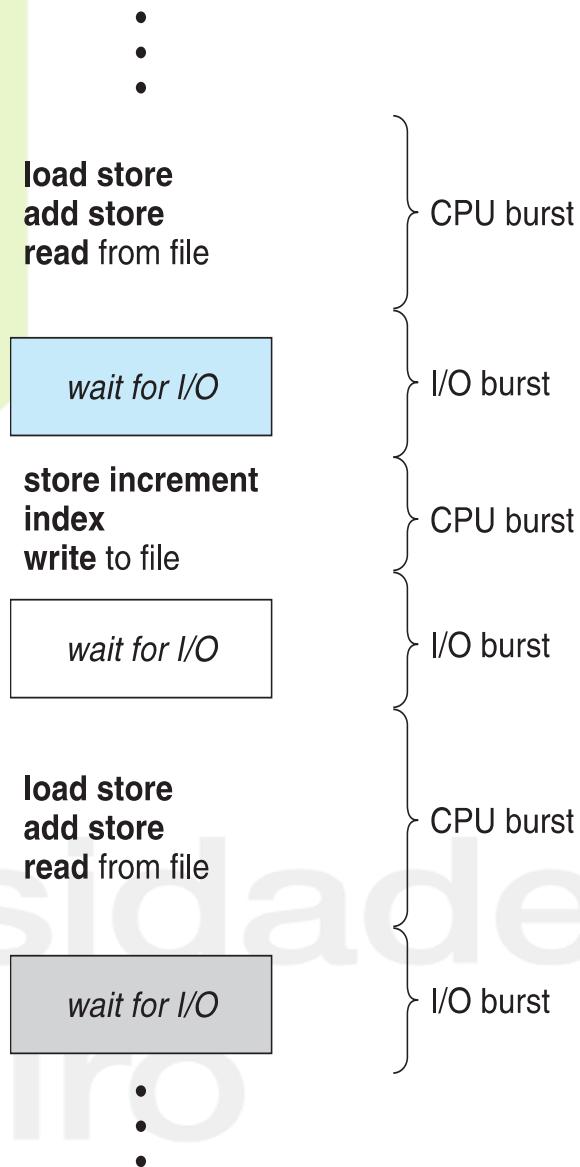
Scheduling

Scheduling – Objectivo (Silberschatz)

- Introdução ao scheduling, que é a base dos sistemas operativos multiprogramados
- Descrição dos diversos algoritmos de scheduling
- Discussão dos critérios para a seleção de um algoritmo para um sistema em particular
- Examinar os algoritmos de scheduling de vários sistemas operativos

Scheduling – Conceitos Básicos (Silberschatz)

- Utilização ao máximo da CPU que é possível obter com multiprogramação
- CPU e I/O burst cycles – a execução de um processo consiste num ciclo de execução da CPU seguido de uma espera por I/O
- CPU burst seguido de I/O burst



Scheduler – Tipos

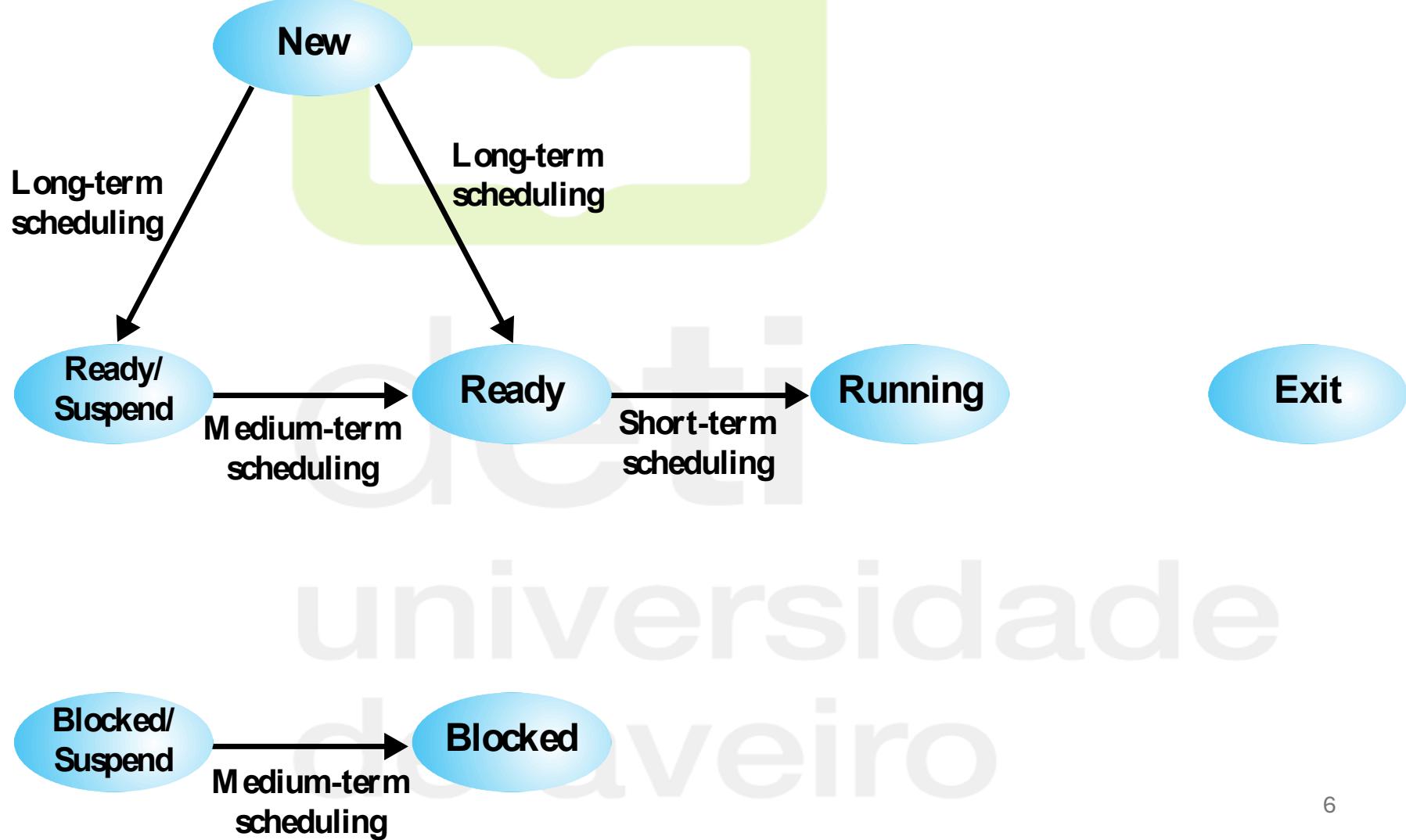
- Long-term scheduling
 - a decisão de adicionar ao conjunto de processos a serem executados
- Medium-term scheduling
 - a decisão de adicionar ao número de processos que se encontram parcialmente ou na totalidade em memória
- Short-term scheduling
 - a decisão de qual o processo entre os disponíveis na fila de Ready que será executado pelo processador
- I/O scheduling
 - a decisão de qual o processo entre os bloqueados para I/O será tratado por um dispositivo de I/O

Scheduler – Tipos

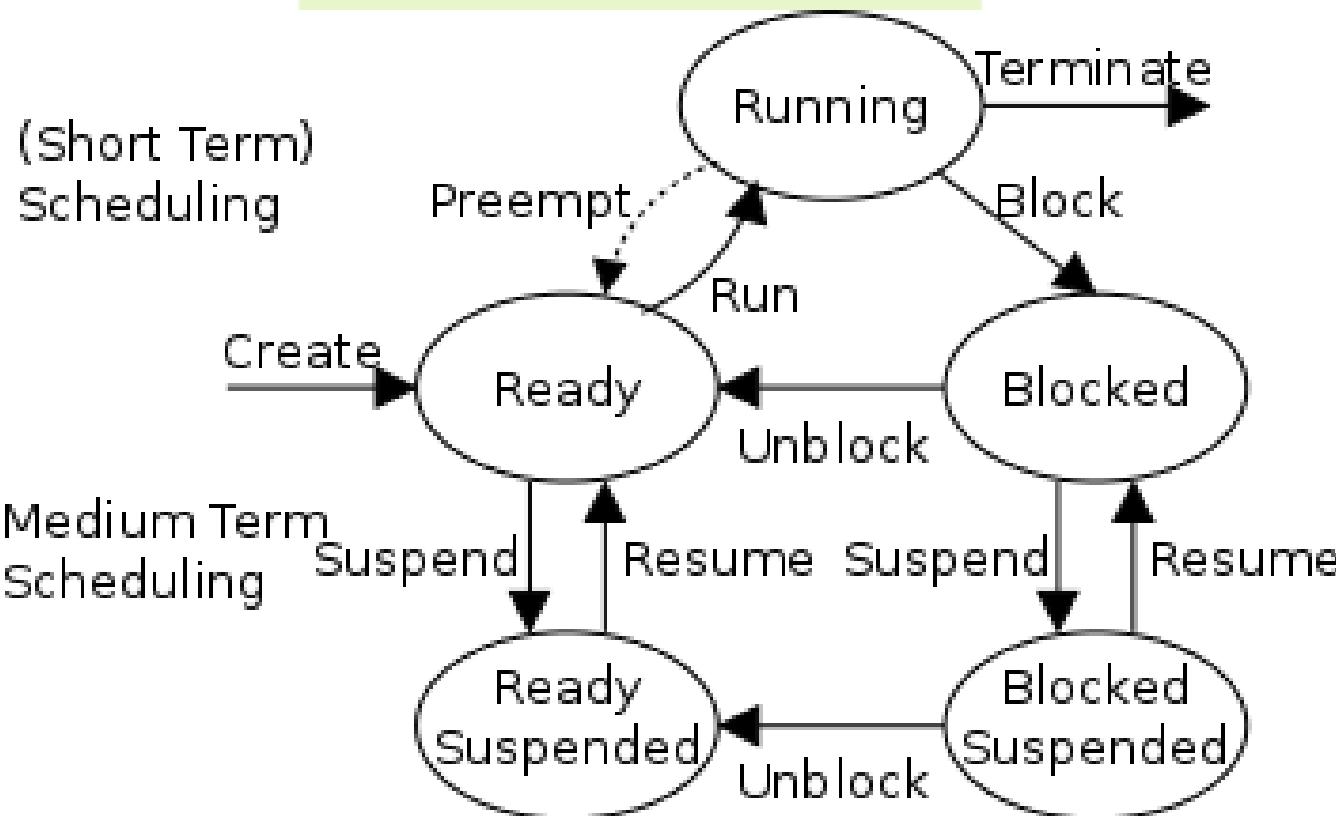
- O scheduling pode decompor-se em três funções separadas:



Scheduling – Transição de Estados dos Processos

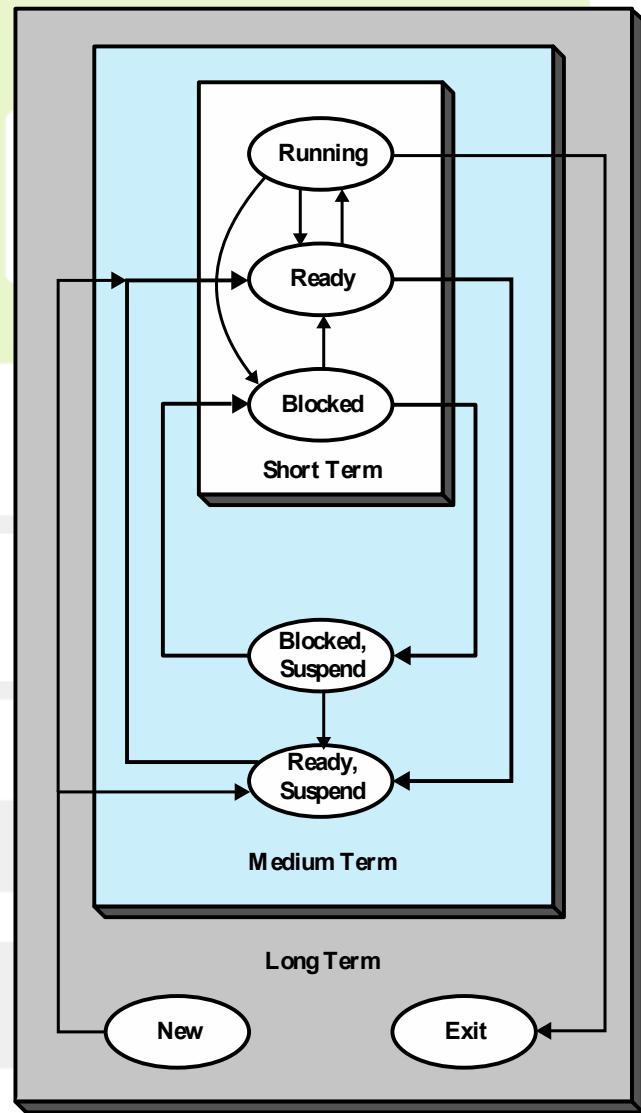


Scheduling – Transição de Estados dos Processos (Gottlieb)

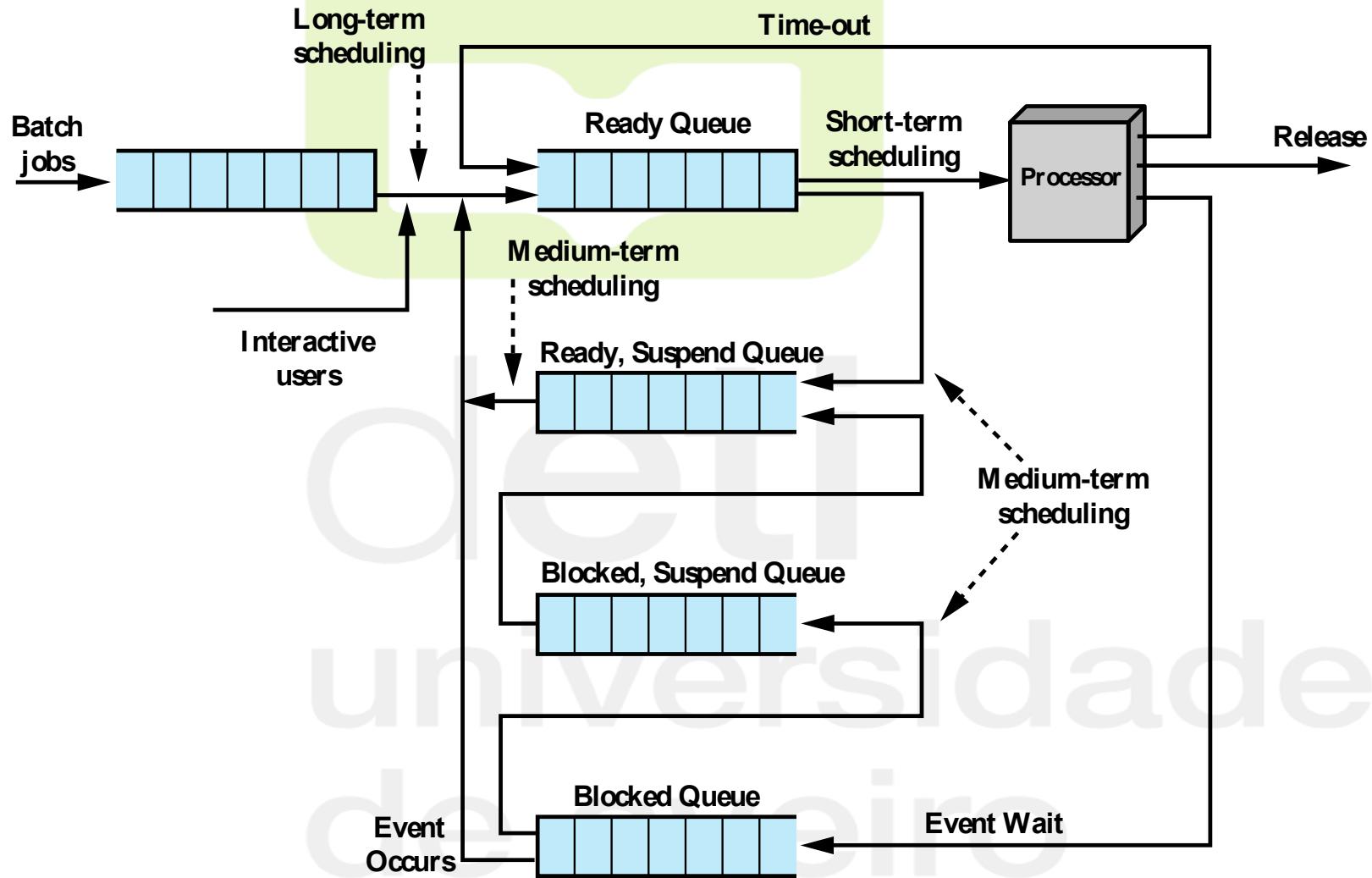


Unblock is done by another task (a.k.a. wakeup, release, V)
Block is a.k.a. sleep, request, P)

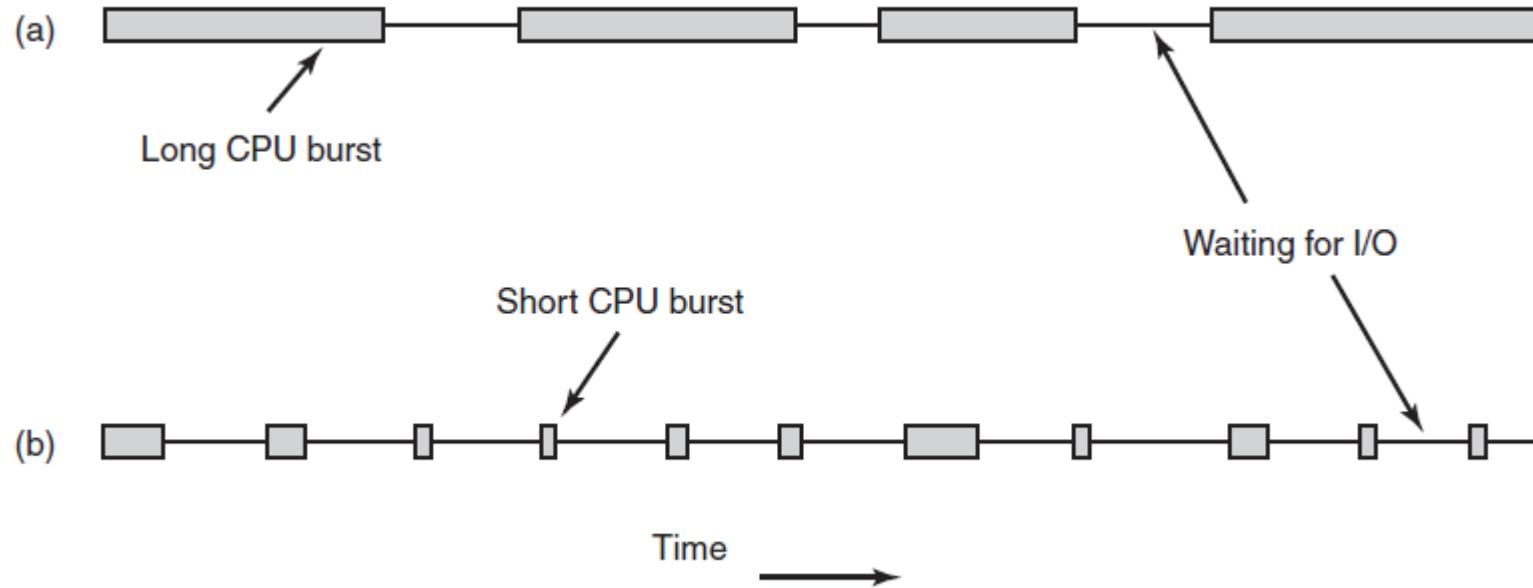
Scheduling – Níveis



Scheduling – Diagrama de filas



Scheduling – Comportamento dos processos



- A execução de um processo consiste num ciclo de execução da CPU seguido de uma espera por I/O
 - a) um processo CPU-bound
 - b) um processo I/O-bound

Scheduler – CPU (Silberschatz)

- Short-term scheduler seleciona um de entre os processos na fila de Ready, e aloca-lhe a CPU
 - A fila pode ser ordenada de diferentes maneiras
- Decisões de scheduling da CPU podem ter lugar quando um processo:
 1. comuta de Running para um estado de Waiting
 2. comuta de Running para o estado Ready
 3. comuta de um estado de Waiting para Ready
 4. termina
- Os scheduling de 1 e 4 são **não-preemptivos**
- Os outros (2 e 3) são **preemptivos**

Scheduler – Dispatcher (Silberschatz)

- O módulo de dispatcher dá ao processo selecionado pelo short-term scheduler controlo sobre a CPU. Este procedimento envolve:
 - A comutação de contexto
 - A comutação para modo de utilizador
 - Saltar para a posição correta no programa de utilizador e reiniciar o programa
- Latência do dispatcher – tempo que leva para que o dispatcher pare um processo e inicie outro no estado de Running

Scheduling – Categorias de Algoritmos

- Batch
- Interativos
- Tempo real

Scheduling – Objetivos dos Algoritmos

- Gerais
 - Justiça – cada processo tem o tempo justo de CPU
 - Garantir o cumprimento das políticas
 - Equilíbrio – manter todas as partes ocupadas
- Batch
 - Desempenho – maximizar os jobs por hora
 - Tempo de turnaround – minimizar tempo sub/term
 - Utilização da CPU – manter a CPU sempre ocupada
- Interativos
 - Tempo de resposta – responde a pedidos rápido
 - Proporcionalidade – dá o que utilizadores esperam
- Tempo real
 - Cumprimento de prazos – evita a perda de dados
 - Previsibilidade – evita degradação do serviço

Scheduling – Sistemas Batch

- First-Come First-Served
- Shortest Job First
- Shortest Remaining Time Next

de
ti
universidade
de aveiro

Scheduling – Sistemas Batch

First-Come First-Served (FCFS)

- **Não preemptivo**
- A mais simples política de scheduling
- Também conhecido por first-in-first-out (FIFO)
- Quando o processo corrente termina, é escolhido o processo da fila de Ready que está à espera há mais tempo
- O desempenho é melhor para processos longos do que para os curtos
- Tende a favorecer processos CPU-bound, em detrimento dos I/O-bound

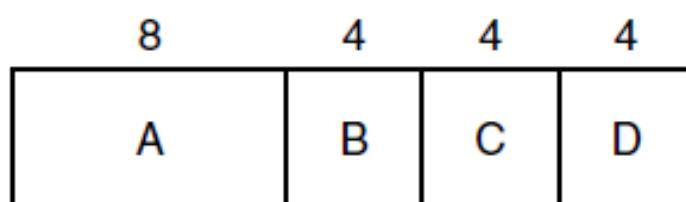
Scheduling – Sistemas Batch

Shortest Job First (SJF)

- **Não preemptivo**
- Associa a cada processo o tempo da próxima utilização da CPU
 - Utiliza esses tempos para calendarizar o processo com o mais curto tempo
- SJF fornece um valor mínimo de tempo de espera para um determinado conjunto de processos
 - A dificuldade consiste em saber de antemão o tempo da próxima utilização de CPU
 - Pode-se perguntar ao utilizador (...)

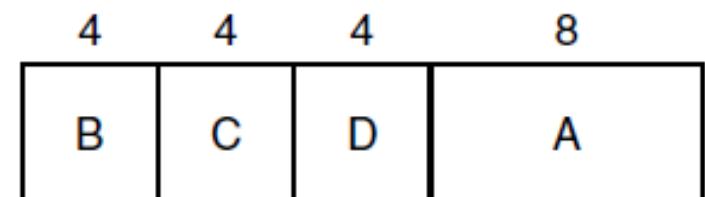
Scheduling – Sistemas Batch

Shortest Job First (SJF) - Exemplo



(a)

quatro processos a
correr na ordem original



(b)

quatro processos a correr
na ordem do SJF

Scheduling – Sistemas Batch

Shortest Remaining Time Next (SRTN)

- Versão **preemptiva** do SJF
- O scheduler escolhe o processo que possui o mais curto tempo expectável de tempo de processamento
- Existe o risco de processos mais longos não terem acesso à CPU

Scheduling – Sistemas Interativos

- Round-Robin Scheduling
- Priority Scheduling
- Multiple Queues
- Shortest Process Next
- Guaranteed Scheduling
- Lottery Scheduling
- Fair-Share Scheduling

universidade
de aveiro

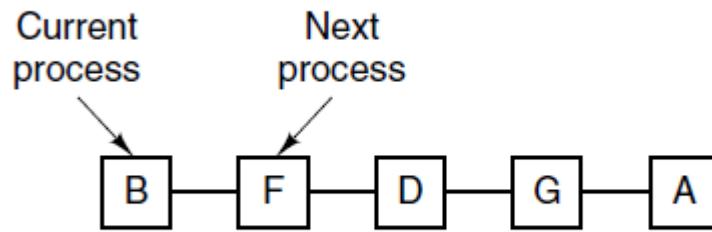
Scheduling – Sistemas Interativos

Round-Robin (RR)

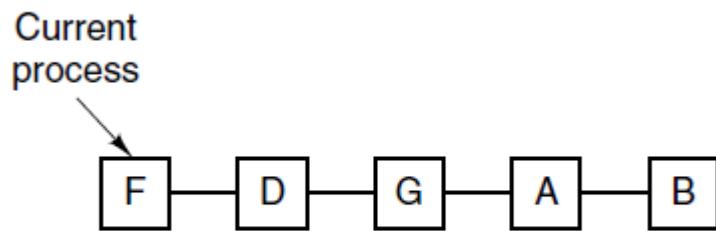
- Scheduling **preemptivo**, baseado em relógio
- Também conhecido por **time-slicing**, pois a cada processo é facultado um time slice antes de ser interrompido
- Um ponto importante consiste no tempo do time slice, ou *time quantum* (q), a ser utilizado
 - Se o q é grande, o algoritmo é análogo a um FIFO
 - Se o q é pequeno, o algoritmo é interativo
 - Um q muito pequeno pode levar a um desperdício de recursos com alto overhead
- Particularmente eficaz em sistemas generalistas de time-sharing

Scheduling – Sistemas Interativos

Round-Robin (RR) – Exemplo



(a)
Lista dos processos runnable.



(b)
Lista dos processos runnable depois
de B usar o seu quantum q

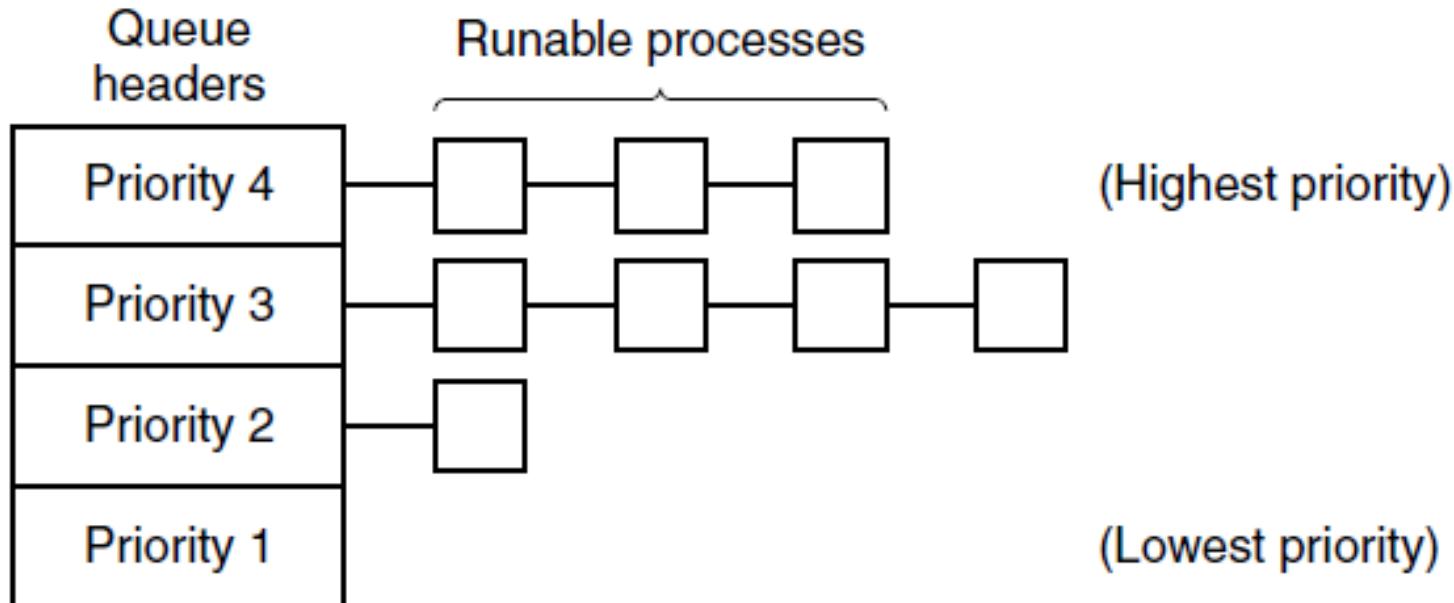
Scheduling – Sistemas Interativos

Priority Scheduling

- Scheduling pode ser **preemptivo** ou não **preemptivo**
- Um número de prioridade (inteiro) é associado a cada processo (menor inteiro=prioridade mais alta)
- O processo com maior prioridade é alocado à CPU
 - Problema: inanição: processos de baixa prioridade podem nunca ter acesso à CPU
 - Solução: envelhecimento: à medida que o tempo decorre, aumentar a prioridade dos processos

Scheduling – Sistemas Interativos

Priority Scheduling - Exemplo



Quatro classes de prioridade

Scheduling – Sistemas Interativos

Multiple Queues

- Scheduling preemptivo
- A fila de Ready é particionada em filas separadas
- Por exemplo:
 - foreground (interativos)
 - background (batch)
- Cada fila tem o seu algoritmo de scheduling
 - foreground - RR
 - background - FCFS

Scheduling – Sistemas Interativos

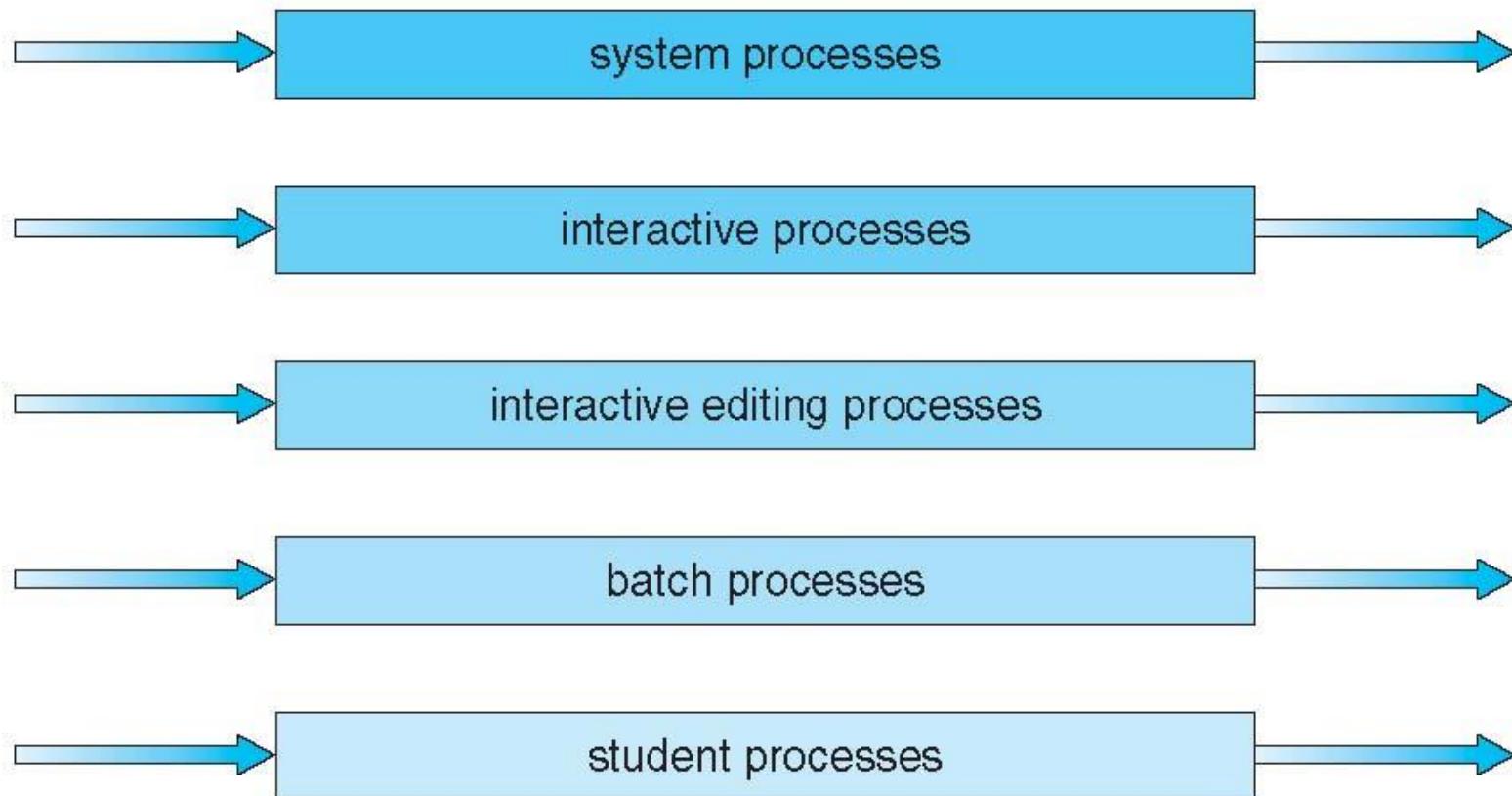
Multiple Queues

- Deve ser efetuado scheduling entre filas diferentes
 - Scheduling de prioridade fixa – por exemplo, servir todos os processos em foreground, e depois os de background – possibilidade de inanição
 - Time slice – cada fila recebe uma determinada quantidade de tempo que pode ser utilizado para fazer o scheduling dos processos da fila, por exemplo, 80% do tempo para foreground em RR, 20% do tempo para background em FCFS

Scheduling – Sistemas Interativos

Multiple Queues – Exemplo

highest priority



lowest priority

Scheduling – Sistemas Interativos

Shortest Process Next (SPN)

- **Não preemptivo**
- O processo com o expectável menor tempo de processamento é selecionado a seguir
- Um processo curto salta para o início da fila
- Possibilidade de inanição de processos mais longos
- Uma dificuldade consiste em estimar o tempo de processamento requerido por cada processo

Scheduling – Sistemas em Tempo Real

- O tempo tem um papel essencial
- Categorias
 - Hard real time
 - Soft real time
 - Periodic or aperiodic