

Arquitetura e Organização de Computadores

Aula-08: Suporte ao Sistema Operacional

Eliseu César Miguel

Livro: Arquitetura e Organização de Computadores
William Stallings 8^a Edição
Universidade Federal de Alfenas

December 13, 2022



Organização da Aula

1 Introdução

2 Evolução dos Sistemas Operacionais

3 Sistemas Multiprogramados

4 Gerência de Memória

5 Memória Virtual

Organização da Aula

- 1 Introdução
- 2 Evolução dos Sistemas Operacionais
- 3 Sistemas Multiprogramados
- 4 Gerência de Memória
- 5 Memória Virtual

Organização da Aula

- 1 Introdução
- 2 Evolução dos Sistemas Operacionais
- 3 Sistemas Multiprogramados
- 4 Gerência de Memória
- 5 Memória Virtual

Organização da Aula

- 1 Introdução
- 2 Evolução dos Sistemas Operacionais
- 3 Sistemas Multiprogramados
- 4 Gerência de Memória
- 5 Memória Virtual

Organização da Aula

- 1 Introdução
- 2 Evolução dos Sistemas Operacionais
- 3 Sistemas Multiprogramados
- 4 Gerência de Memória
- 5 Memória Virtual

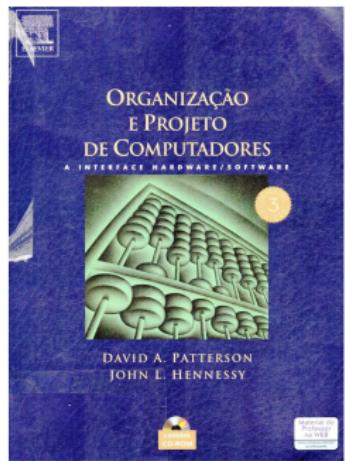
Bibliografia básica



Livro Texto



Complementar



Complementar

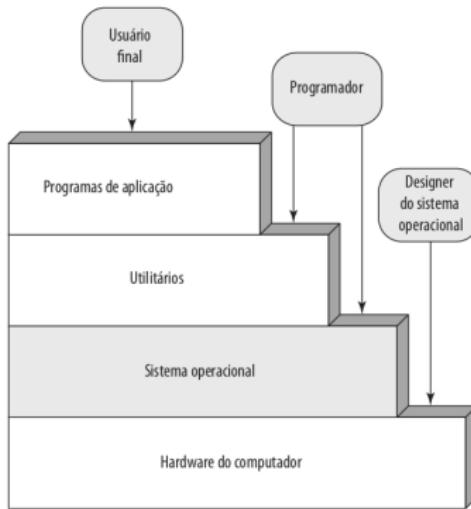
Além da bibliografia básica, visite o programa de ensino para ver outras bibliografias. Também, durante o curso, várias bibliografias em sítios da Internet serão apresentadas.

Introdução: Suporte ao Sistema Operacional

Um SO é um programa que controla a execução dos programas aplicativos e atua como uma interface entre o usuário e o hardware do computador. Ele pode ser imaginado como tendo dois objetivos:

- **conveniência:** um SO torna um computador mais conveniente para uso.
- **eficiência:** um SO permite que os recursos do sistema de computação sejam usados de uma maneira eficiente.

Figura 8.1 Camadas e visões de um sistema de computação



Introdução: Funções do Sistema Operacional

O SO propõe quanto ao uso do sistema computacional: conveniência e eficiência Como interface entre usuário e computador

- Criação de programas
- Execução de programas
- Acesso aos dispositivos de E/S
- Acesso controlado aos arquivos
- Acesso ao sistema
- Detecção e resposta aos erros
- contabilidade

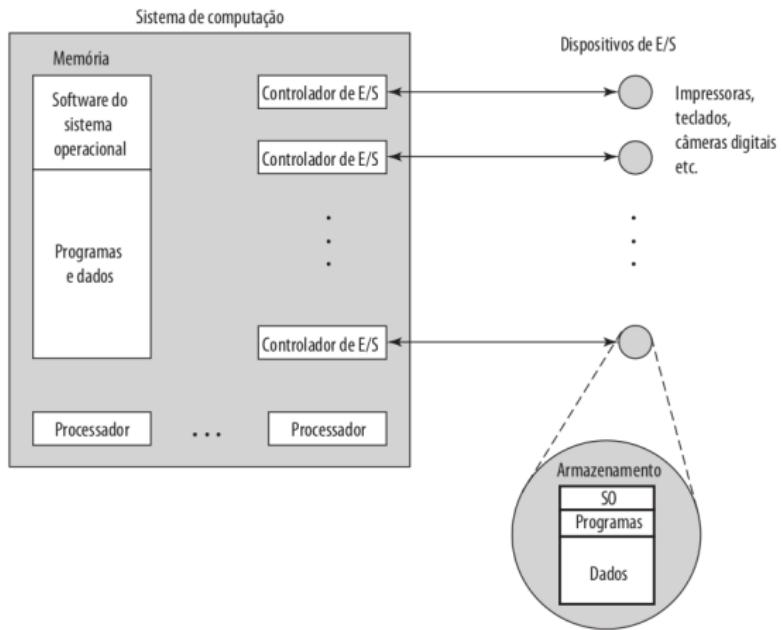
Como gerenciador de recursos

- O SO é implementado em *software*. Então, ele compartilha a CPU
- Deve deixar o controle do sistema para a execução de outros programas
- Escalona os programas a serem executados
- Alocação de memória (SO + *hardware de gerenciamento de memória (MMU)*)

Introdução: SO

Visão geral da memória do computador

Figura 8.2 O sistema operacional como gerenciador de recursos



Evolução dos SO: Execução uniprogramada

Sistemas em lote (*bath*)

- **Escalonamento:** usuários disputam o uso reservando o tempo (45 minutos)
- **Tempo de preparação:** prejuízos com o carregamento de um único programa (*job*). Na ocasião, o carregamento do programa exigia o carregamento do compilador e se algum erro ocorresse, todo o trabalho era perdido

Monitor

- Criação de um programa (residente) para escalar os trabalhos (*jobs*)
- **Escalonamento:** O monitor indicava um *job*. Ao fim, o controle voltava ao monitor, que repetia o processo
- **Tempo de preparação:** Uso de uma linguagem que carregava os compiladores em memória de massa, como fitas
- **Programa aguarda grandes ciclos de processador por E/S**

O que tratar?

- **Proteção de memória:** programa de usuário não deve acessar a área do monitor
- **Temporizador:** impedir que um programa monopolize o processador
- **Instruções privilegiadas:** instruções que somente o monitor pode executar
- **Interrupções:** **não implementadas.** Hardware necessário à multiprogramação



Evolução dos SO: Execução uniprogramada

Sistemas em lote (*bath*)

- **Escalonamento:** usuários disputam o uso reservando o tempo (45 minutos)
- **Tempo de preparação:** prejuízos com o carregamento de um único programa (*job*). Na ocasião, o carregamento do programa exigia o carregamento do compilador e se algum erro ocorresse, todo o trabalho era perdido

Monitor

- Criação de um programa (residente) para escalar os trabalhos (*jobs*)
- **Escalonamento:** O monitor indicava um *job*. Ao fim, o controle voltava ao monitor, que repetia o processo
- **Tempo de preparação:** Uso de uma linguagem que carregava os compiladores em memória de massa, como fitas
- **Programa aguarda grandes ciclos de processador por E/S**

O que tratar?

- **Proteção de memória:** programa de usuário não deve acessar a área do monitor
- **Temporizador:** impedir que um programa monopolize o processador
- **Instruções privilegiadas:** instruções que somente o monitor pode executar
- **Interrupções:** **não implementadas.** Hardware necessário à multiprogramação



Evolução dos SO: Execução uniprogramada

Sistemas em lote (*bath*)

- **Escalonamento:** usuários disputam o uso reservando o tempo (45 minutos)
- **Tempo de preparação:** prejuízos com o carregamento de um único programa (*job*). Na ocasião, o carregamento do programa exigia o carregamento do compilador e se algum erro ocorresse, todo o trabalho era perdido

Monitor

- Criação de um programa (residente) para escalar os trabalhos (*jobs*)
- **Escalonamento:** O monitor indicava um *job*. Ao fim, o controle voltava ao monitor, que repetia o processo
- **Tempo de preparação:** Uso de uma linguagem que carregava os compiladores em memória de massa, como fitas
- **Programa aguarda grandes ciclos de processador por E/S**

O que tratar?

- **Proteção de memória:** programa de usuário não deve acessar a área do monitor
- **Temporizador:** impedir que um programa monopolize o processador
- **Instruções privilegiadas:** instruções que somente o monitor pode executar
- **Interrupções:** **não implementadas.** Hardware necessário à multiprogramação



Evolução dos SO: Execução uniprogramada

Sistemas em lote (*bath*)

- **Escalonamento:** usuários disputam o uso reservando o tempo (45 minutos)
- **Tempo de preparação:** prejuízos com o carregamento de um único programa (*job*). Na ocasião, o carregamento do programa exigia o carregamento do compilador e se algum erro ocorresse, todo o trabalho era perdido

Monitor

- Criação de um programa (residente) para escalar os trabalhos (*jobs*)
- **Escalonamento:** O monitor indicava um *job*. Ao fim, o controle voltava ao monitor, que repetia o processo
- **Tempo de preparação:** Uso de uma linguagem que carregava os compiladores em memória de massa, como fitas
- **Programa aguarda grandes ciclos de processador por E/S**

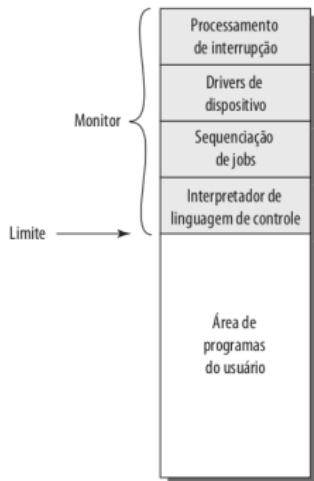
O que tratar?

- **Proteção de memória:** programa de usuário não deve acessar a área do monitor
- **Temporizador:** impedir que um programa monopolize o processador
- **Instruções privilegiadas:** instruções que somente o monitor pode executar
- **Interrupções:** **não implementadas.** Hardware necessário à multiprogramação



Evolução dos SO: Monitor e Linguagem de Preparação

Figura 8.3 Layout de memória para um monitor residente



```
$JOB  
$FTN  
...     } Instruções em FORTRAN  
$LOAD  
$RUN  
...     } Dados  
$END
```

Evolução dos SO: o problema da uniprogramação

Figura 8.5 Exemplo de multiprogramação



Sistemas Operacionais Multiprogramados

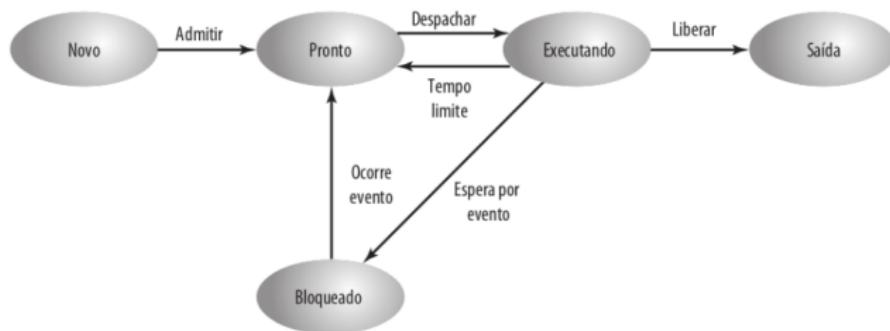
Vários recursos de *hardware* foram desenvolvidos para oferecer suporte aos SOs para a multiprogramação

- Suporte ao escalonamento de processos
 - ▶ *hardware temporizador* no processador
 - ▶ sistema de troca de contexto
 - ▶ sistema de interrupção
- Suporte ao gerenciamento de memória
 - ▶ instruções privilegiadas
 - ▶ estrutura de *swapping*
 - ▶ sistema de paginação
- Suporte ao gerenciamento de endereços
 - ▶ Novas formas de modos endereçamento (afeta mais o compilador)
 - ▶ Permite o uso de memória virtual
- Recursos novos para E/S, como DMA

Multiprogramação: Escalonamento de processos

O escalonamento de processos é a chave para a multiprogramação

Figura 8.7 Modelo de processo com cinco estados



- **Novo:** o programa é admitido mas não está pronto para executar. O SO iniciará o processo, movendo-o para o estado pronto.
- **Pronto:** o processo está pronto para ser executado e está aguardando o acesso ao processador.
- **Em execução:** o processo está sendo executado pelo processador.
- **Suspensão:** o processo está com sua execução suspensa, aguardando por algum recurso do sistema, como a E/S.
- **Concluído:** o processo terminou e será destruído pelo SO.

Multiprogramação: Escalonamento de processos

O SO deve controlar o estado dos processos

Figura 8.7 Modelo de processo com cinco estados

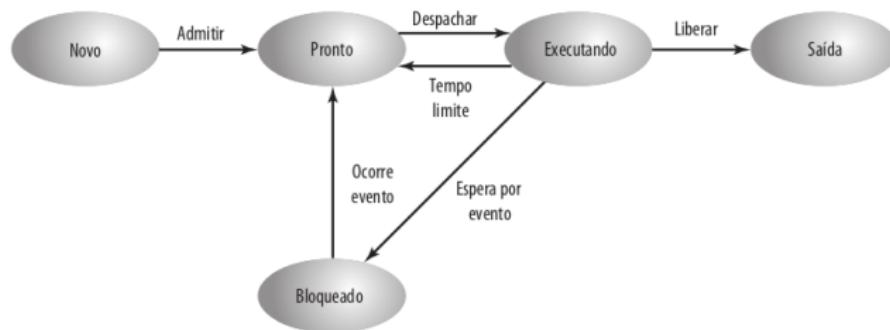


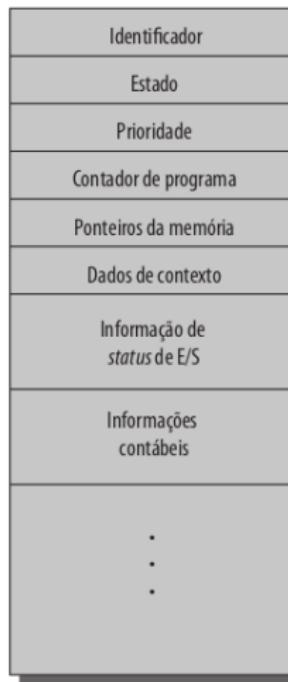
Tabela 8.4 Tipos de escalonamento

Escalonamento de longo prazo	A decisão de acrescentar ao pool de processos a serem executados
Escalonamento a médio prazo	A decisão de acrescentar ao número de processos que estão parcial ou totalmente na memória principal
Escalonamento de curto prazo	A decisão sobre qual processo disponível será executado pelo processador
Escalonamento de E/S	A decisão sobre qual solicitação de E/S pendente do processo será tratada por um dispositivo de E/S disponível

Multiprogramação: Recursos para a troca de contexto

Bloco de controle de processo

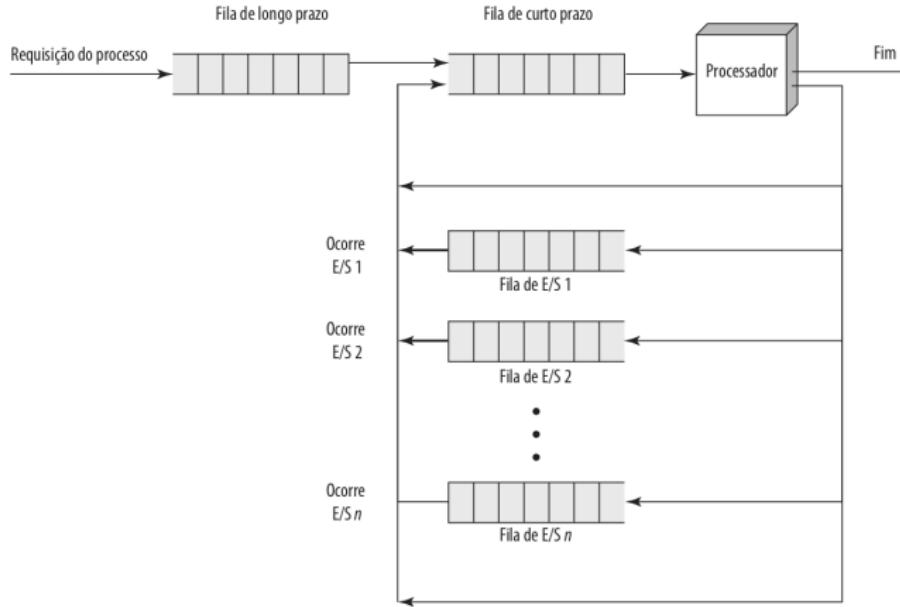
Figura 8.8 Bloco de controle do processo



Multiprogramação: Filas de processos

A figura não mostra a relação do sistema com a memória virtual.
Neste caso, não vemos a função do escalonador de médio prazo

Figura 8.11 Diagrama de filas do escalonamento de processador

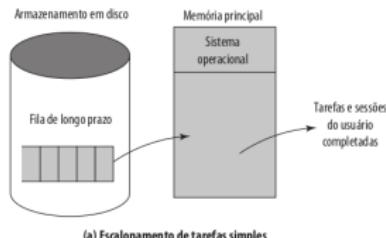


Gerenciamento de Memória: Swapping

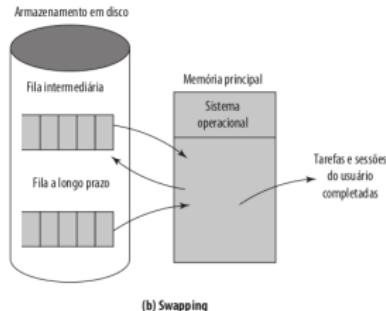
Fundamental para sistemas multiprogramados, o SO divide dinamicamente a memória para alojar vários programas.

- Quanto mais programas, menos ocioso o processador ficará
- As E/S são lentas e a memória principal é cara
- *Swapping*: alternativa para aumentar o número de processos

Figura 8.12 O uso do swapping



(a) Escalonamento de tarefas simples



(b) Swapping

Gerenciamento de Memória: Paginação da Memória

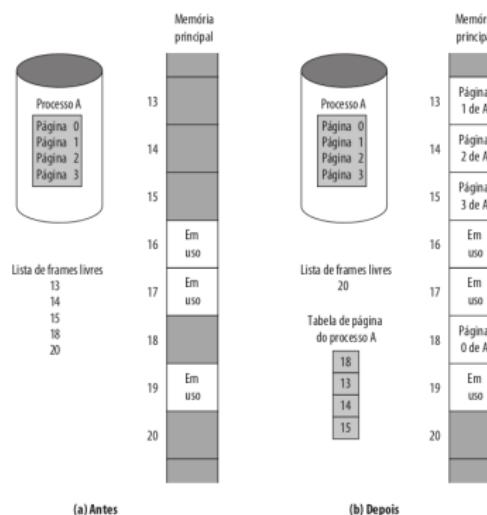
A Memória é dividida em várias partes para conter vários programas.

Há várias formas de se dividir a memória (disciplina de SO).

Pare efeitos didáticos, consideramos apenas a *paginação em tamanho único*.

- **Paginação:** (páginas de tamanho único ou variado)
- Segmentação: tamanhos (dinâmicos) proporcionais aos processos
- Paginação Segmentada: opção que combina as anteriores

Figura 8.15 Alocação de frames livres

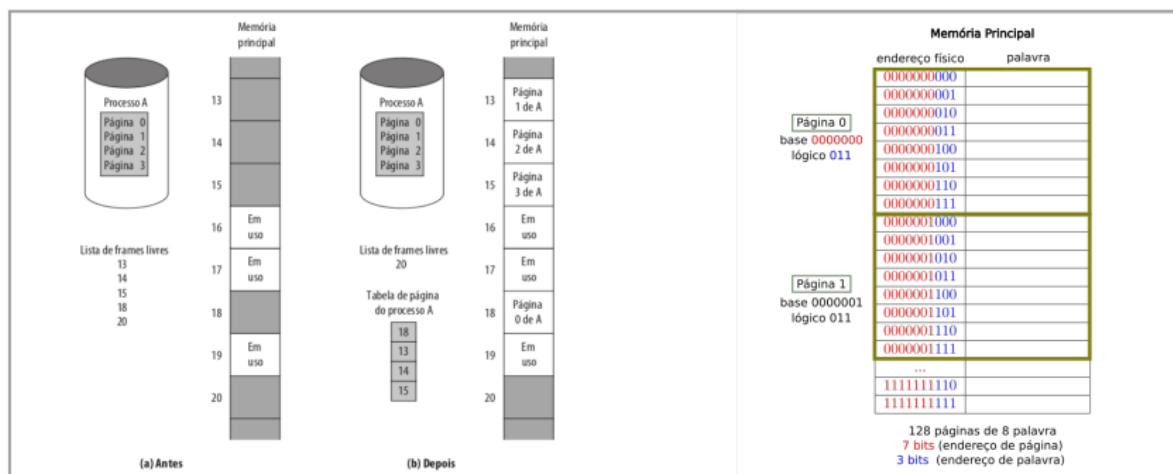


Gerenciamento de Memória: Paginação da Memória

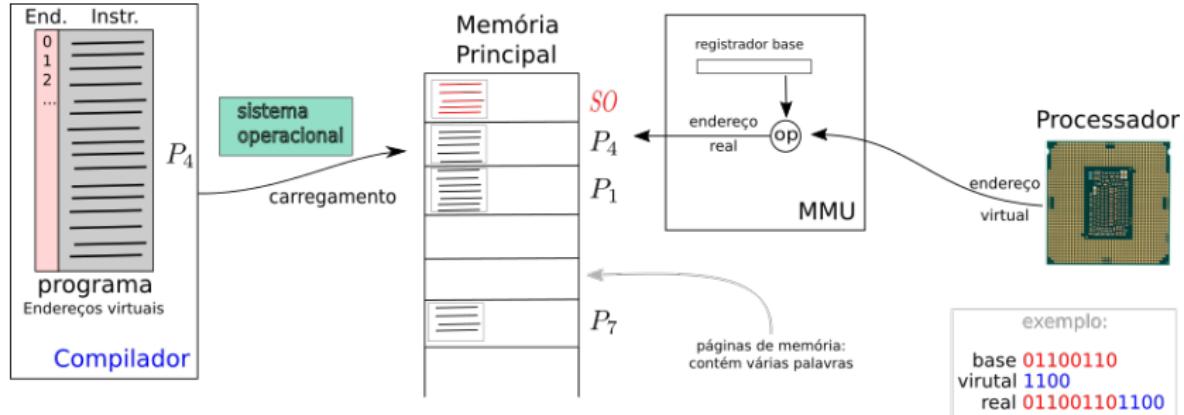
O SO precisa de recursos de hardware para gerenciar as páginas e endereços.

- Endereço Físico é combinação entre endereço virtual e endereço de base
 - Físico (real)**: é o endereço único de uma posição da memória
 - Lógico (virtual)**: utilizado pelo processador como parte do endereço físico
 - De base**: endereço de uma página
- O Compilador gera o código com endereços relativos ao contador de programa (lógicos)
- O Sistema Operacional não dispõe sempre das mesmas páginas para alocar os processos

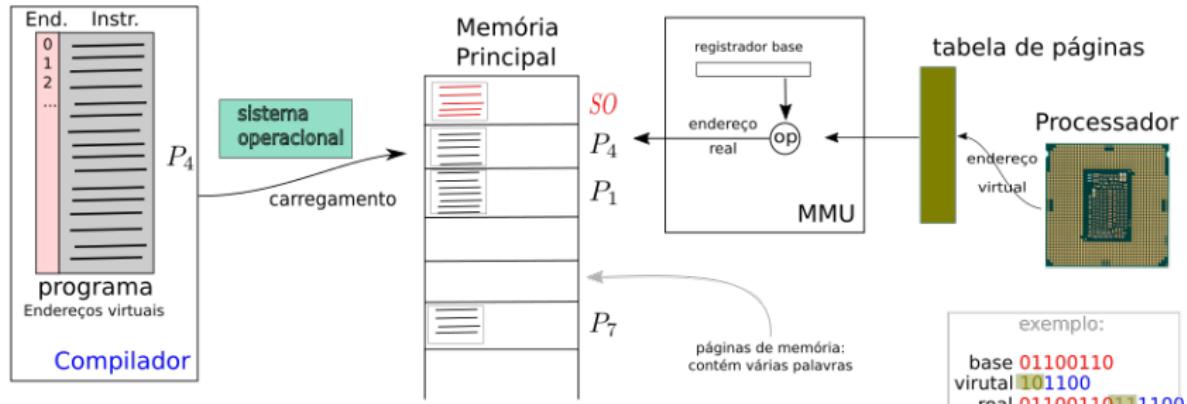
Importante: Necessidade de uma tabela de páginas para gerenciar cada processo



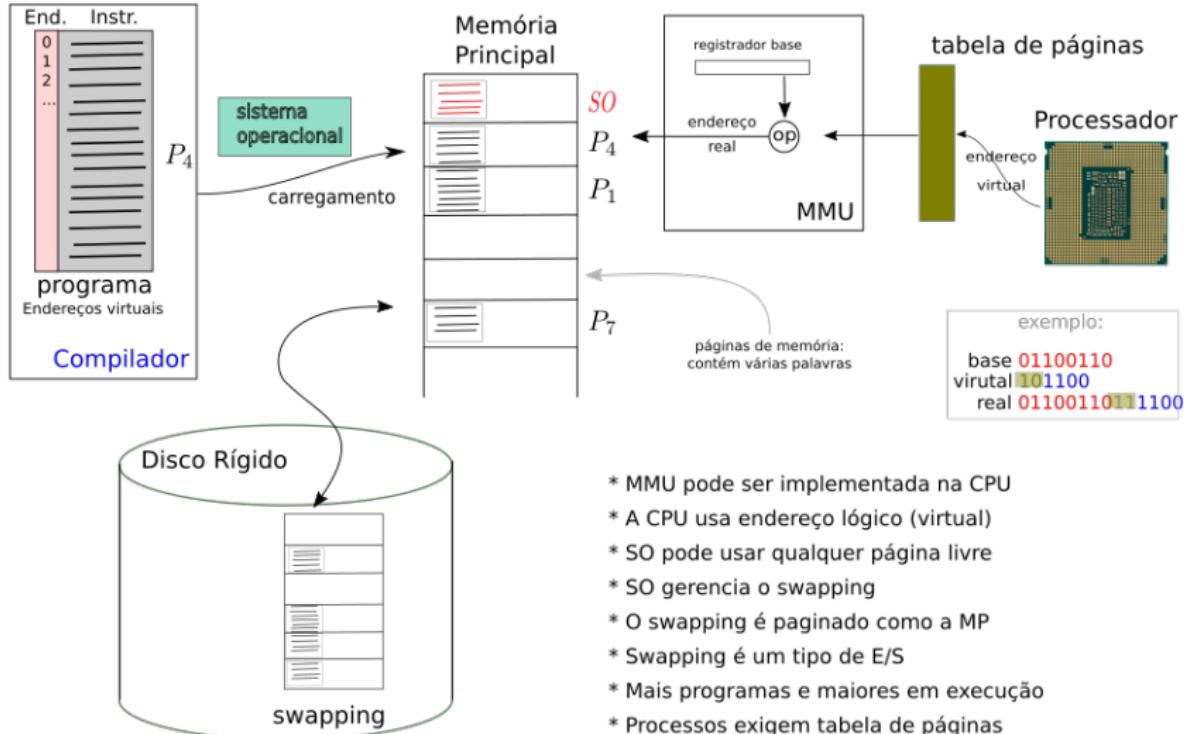
Gerenciamento de Memória: Carregamento de programas



Gerenciamento de Memória: Carregamento de programas



Gerenciamento de Memória: Carregamento de programas

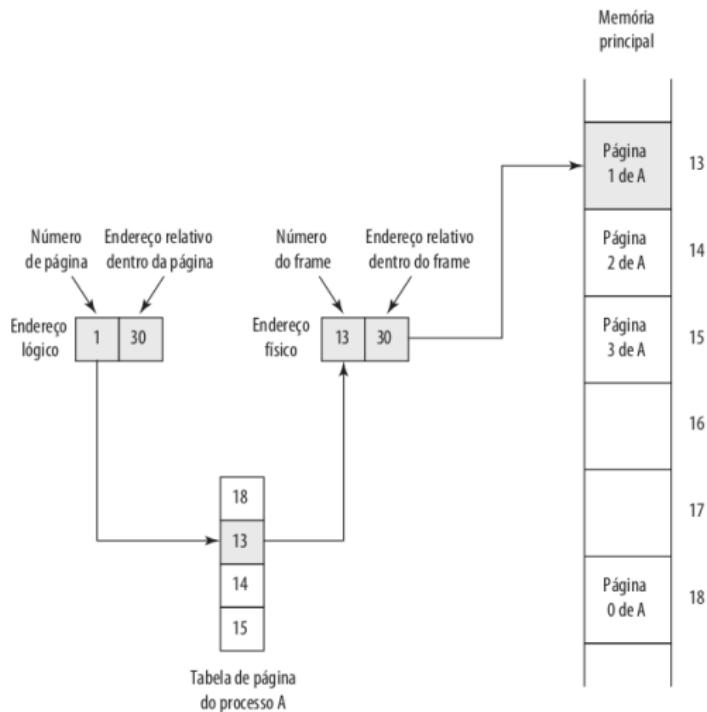


- * MMU pode ser implementada na CPU
- * A CPU usa endereço lógico (virtual)
- * SO pode usar qualquer página livre
- * SO gerencia o swapping
- * O swapping é paginado como a MP
- * Swapping é um tipo de E/S
- * Mais programas e maiores em execução
- * Processos exigem tabela de páginas

Gerenciamento de Memória: Tabela de Página

O processador deseja a trigésima palavra da primeira página do processo A

Figura 8.16 Endereços lógicos e físicos



Memória Virtual: Extensão da MP no disco

Paginação por Demanda

Técnica de trazer para a memória principal as páginas de programas apenas quando necessárias!

O que temos como consequência?

- Programas podem ser paginados mesmo se ultrapassar a capacidade da MP
- Uso de interrupção para falta de página (*page fault*)
- Uso da localidade para manter as páginas mais acessadas na memória
- Algoritmos complexos (pesquisas da década de 70) para a substituição de páginas
- Estrutura da tabela de página (*Armazenada na memória principal e Virtual*)
 - ▶ As tabelas são muito grandes e suscetíveis ao *swapping*
 - ▶ Pelo menos a tabela da página ocupa o processo inteiro na memória principal
- *Translation lookaside buffer (TLB)* hardware de buffer da tabela de páginas
 - ▶ Acessar a memória virtual pode gerar duas faixas de página (tabela e página)
 - ▶ Buscar uma palavra na MP também exigirá dois acessos* (tabela e palavra)
 - ▶ O TLB (hardware) armazena as páginas mais acessadas e evita um acesso à MP.

Memória Virtual: Extensão da MP no disco

Paginação por Demanda

Técnica de trazer para a memória principal as páginas de programas apenas quando necessárias!

O que temos como consequência?

- Programas podem ser paginados mesmo se ultrapassar a capacidade da MP
- Uso de interrupção para falta de página (*page fault*)
- Uso da localidade para manter as páginas mais acessadas na memória
- Algoritmos complexos (pesquisas da década de 70) para a substituição de páginas
- Estrutura da tabela de página (*Armazenada na memória principal e Virtual*)
 - ▶ As tabelas são muito grandes e suscetíveis ao *swapping*
 - ▶ Pelo menos a tabela da página ocupa o processo inteiro na memória principal
- *Translation lookaside buffer (TLB)* hardware de buffer da tabela de páginas
 - ▶ Acessar a memória virtual pode gerar duas fases de página (tabela e página)
 - ▶ Buscar uma palavra na MP também exigirá dois acessos* (tabela e palavra)
 - ▶ O TLB (hardware) armazena as páginas mais acessadas e evita um acesso à MP.

Memória Virtual: Extensão da MP no disco

Paginação por Demanda

Técnica de trazer para a memória principal as páginas de programas apenas quando necessárias!

O que temos como consequência?

- Programas podem ser paginados mesmo se ultrapassar a capacidade da MP
- Uso de interrupção para falta de página (*page fault*)
- Uso da localidade para manter as páginas mais acessadas na memória
- Algoritmos complexos (pesquisas da década de 70) para a substituição de páginas
- Estrutura da tabela de página (*Armazenada na memória principal e Virtual*)
 - ▶ As tabelas são muito grandes e suscetíveis ao *swapping*
 - ▶ Pela mesma estrutura da página ocorre da programação direta em na memória principal
- *Translation lookaside buffer (TLB)* hardware de buffer da tabela de páginas
 - ▶ Acessar a memória virtual pode gerar duas fases de página (tabela e página)
 - ▶ Buscar uma palavra na MP também exigirá dois acessos* (tabela e palavra)
 - ▶ O TLB (hardware) armazena as páginas mais acessadas e evita um acesso à MP.

Memória Virtual: Extensão da MP no disco

Paginação por Demanda

Técnica de trazer para a memória principal as páginas de programas apenas quando necessárias!

O que temos como consequência?

- Programas podem ser paginados mesmo se ultrapassar a capacidade da MP
- Uso de interrupção para falta de página (*page fault*)
- Uso da localidade para manter as páginas mais acessadas na memória
- Algoritmos complexos (pesquisas da década de 70) para a substituição de páginas
- Estrutura da tabela de página (*Armazenada na memória principal e Virtual*)
 - ▶ As tabelas são muito grandes e suscetíveis ao *swapping*
 - ▶ Pelo menos a tabela da página é carregada pelo processo direcionado na memória principal
- *Translation lookaside buffer (TLB)* hardware de buffer da tabela de páginas
 - ▶ Acessar a memória virtual pode gerar duas faixas de página (tabela e página)
 - ▶ Buscar uma palavra na MP também exige dois acessos* (tabela e palavra)
 - ▶ O TLB (hardware) armazena as páginas mais acessadas e evita um acesso à MP.

Memória Virtual: Extensão da MP no disco

Paginação por Demanda

Técnica de trazer para a memória principal as páginas de programas apenas quando necessárias!

O que temos como consequência?

- Programas podem ser paginados mesmo se ultrapassar a capacidade da MP
- Uso de interrupção para falta de página (*page fault*)
- Uso da localidade para manter as páginas mais acessadas na memória
- Algoritmos complexos (pesquisas da década de 70) para a substituição de páginas
- Estrutura da tabela de página (*Armazenada na memória principal e Virtual*)
 - ▶ As tabelas são muito grandes e suscetíveis ao *swapping*
 - ▶ Pelo menos a tabela da página atual do processo deve estar na memória principal
- *Translation lookaside buffer (TLB)* hardware de buffer da tabela de páginas
 - ▶ Acessar a memória virtual pode gerar duas fases de página (tabela e página)
 - ▶ Buscar uma palavra na MP também exigirá dois acessos* (tabela e palavra)
 - ▶ O TLB (hardware) armazena as páginas mais acessadas e evita um acesso à MP.

Memória Virtual: Extensão da MP no disco

Paginação por Demanda

Técnica de trazer para a memória principal as páginas de programas apenas quando necessárias!

O que temos como consequência?

- Programas podem ser paginados mesmo se ultrapassar a capacidade da MP
- Uso de interrupção para falta de página (*page fault*)
- Uso da localidade para manter as páginas mais acessadas na memória
- Algoritmos complexos (pesquisas da década de 70) para a substituição de páginas
- Estrutura da tabela de página (*Armazenada na memória principal e Virtual*)
 - ▶ As tabelas são muito grandes e susceptíveis ao *swapping*
 - ▶ Pelo menos a tabela da página atual do processo deve estar na memória principal
- *Translation lookaside buffer (TLB)* hardware de buffer da tabela de páginas
 - ▶ Acessar a memória virtual pode gerar duas fases de página (tabela e página)
 - ▶ Buscar uma palavra na MP também exige dois acessos (tabela e palavra)
 - ▶ O TLB (hardware) armazena as páginas mais acessadas e evita um acesso à MP.

Memória Virtual: Extensão da MP no disco

Paginação por Demanda

Técnica de trazer para a memória principal as páginas de programas apenas quando necessárias!

O que temos como consequência?

- Programas podem ser paginados mesmo se ultrapassar a capacidade da MP
- Uso de interrupção para falta de página (*page fault*)
- Uso da localidade para manter as páginas mais acessadas na memória
- Algoritmos complexos (pesquisas da década de 70) para a substituição de páginas
- Estrutura da tabela de página (*Armazenada na memória principal e Virtual*)
 - ▶ As tabelas são muito grandes e susceptíveis ao *swapping*
 - ▶ Pelo menos a tabela da página atual do processo deve estar na memória principal
- *Translation lookaside buffer (TLB)* hardware de buffer da tabela de páginas
 - ▶ Acessar a memória virtual pode gerar duas faixas de página (palavra e página)
 - ▶ Buscar uma palavra na MP também exige dois acessos (tabela e palavra)
 - ▶ O TLB (hardware) armazena as páginas mais acessadas e evita um acesso à MP.

Memória Virtual: Extensão da MP no disco

Paginação por Demanda

Técnica de trazer para a memória principal as páginas de programas apenas quando necessárias!

O que temos como consequência?

- Programas podem ser paginados mesmo se ultrapassar a capacidade da MP
- Uso de interrupção para falta de página (*page fault*)
- Uso da localidade para manter as páginas mais acessadas na memória
- Algoritmos complexos (pesquisas da década de 70) para a substituição de páginas
- Estrutura da tabela de página (*Armazenada na memória principal e Virtual*)
 - ▶ As tabelas são muito grandes e susceptíveis ao *swapping*
 - ▶ Pelo menos a tabela da página atual do processo deve estar na memória principal
- *Translation lookaside buffer (TLB)* *hardware de buffer* da tabela de páginas
 - ▶ Acessar a memória virtual pode gerar duas faltas de página (tabela e página)
 - ▶ Buscar uma palavra na MP também exigiria dois acessos: (tabela e palavra)
 - ▶ O TLB (*hardware*) armazena as páginas mais acessadas e evita um acesso à MP.

Memória Virtual: Extensão da MP no disco

Paginação por Demanda

Técnica de trazer para a memória principal as páginas de programas apenas quando necessárias!

O que temos como consequência?

- Programas podem ser paginados mesmo se ultrapassar a capacidade da MP
- Uso de interrupção para falta de página (*page fault*)
- Uso da localidade para manter as páginas mais acessadas na memória
- Algoritmos complexos (pesquisas da década de 70) para a substituição de páginas
- Estrutura da tabela de página (*Armazenada na memória principal e Virtual*)
 - ▶ As tabelas são muito grandes e susceptíveis ao *swapping*
 - ▶ Pelo menos a tabela da página atual do processo deve estar na memória principal
- *Translation lookaside buffer (TLB)* hardware de buffer da tabela de páginas
 - ▶ Acessar a memória virtual pode gerar duas faltas de página (tabela e página)
 - ▶ Buscar uma palavra na MP também exigiria dois acessos: (tabela e palavra)
 - ▶ O TLB (hardware) armazena as páginas mais acessadas e evita um acesso à MP.

Memória Virtual: Extensão da MP no disco

Paginação por Demanda

Técnica de trazer para a memória principal as páginas de programas apenas quando necessárias!

O que temos como consequência?

- Programas podem ser paginados mesmo se ultrapassar a capacidade da MP
- Uso de interrupção para falta de página (*page fault*)
- Uso da localidade para manter as páginas mais acessadas na memória
- Algoritmos complexos (pesquisas da década de 70) para a substituição de páginas
- Estrutura da tabela de página (*Armazenada na memória principal e Virtual*)
 - ▶ As tabelas são muito grandes e susceptíveis ao *swapping*
 - ▶ Pelo menos a tabela da página atual do processo deve estar na memória principal
- *Translation lookaside buffer (TLB)* hardware de buffer da tabela de páginas
 - ▶ Acessar a memória virtual pode gerar duas faltas de página (tabela e página)
 - ▶ Buscar uma palavra na MP também exigiria dois acessos: (tabela e palavra)
 - ▶ O TLB (hardware) armazena as páginas mais acessadas e evita um acesso à MP.



Memória Virtual: Extensão da MP no disco

Paginação por Demanda

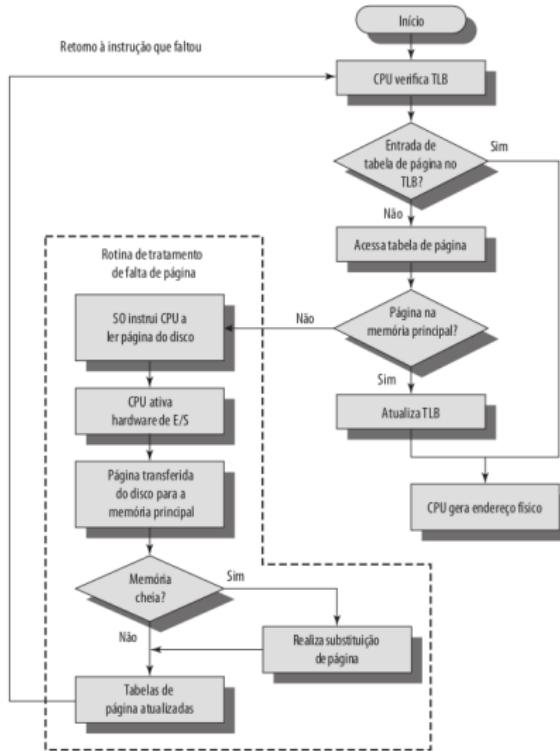
Técnica de trazer para a memória principal as páginas de programas apenas quando necessárias!

O que temos como consequência?

- Programas podem ser paginados mesmo se ultrapassar a capacidade da MP
- Uso de interrupção para falta de página (*page fault*)
- Uso da localidade para manter as páginas mais acessadas na memória
- Algoritmos complexos (pesquisas da década de 70) para a substituição de páginas
- Estrutura da tabela de página (*Armazenada na memória principal e Virtual*)
 - ▶ As tabelas são muito grandes e susceptíveis ao *swapping*
 - ▶ Pelo menos a tabela da página atual do processo deve estar na memória principal
- *Translation lookaside buffer (TLB)* hardware de buffer da tabela de páginas
 - ▶ Acessar a memória virtual pode gerar duas faltas de página (tabela e página)
 - ▶ Buscar uma palavra na MP também exigiria dois acessos: (tabela e palavra)
 - ▶ O TLB (hardware) armazena as páginas mais acessadas e evita um acesso à MP.

Gerencia de Memória: TLB

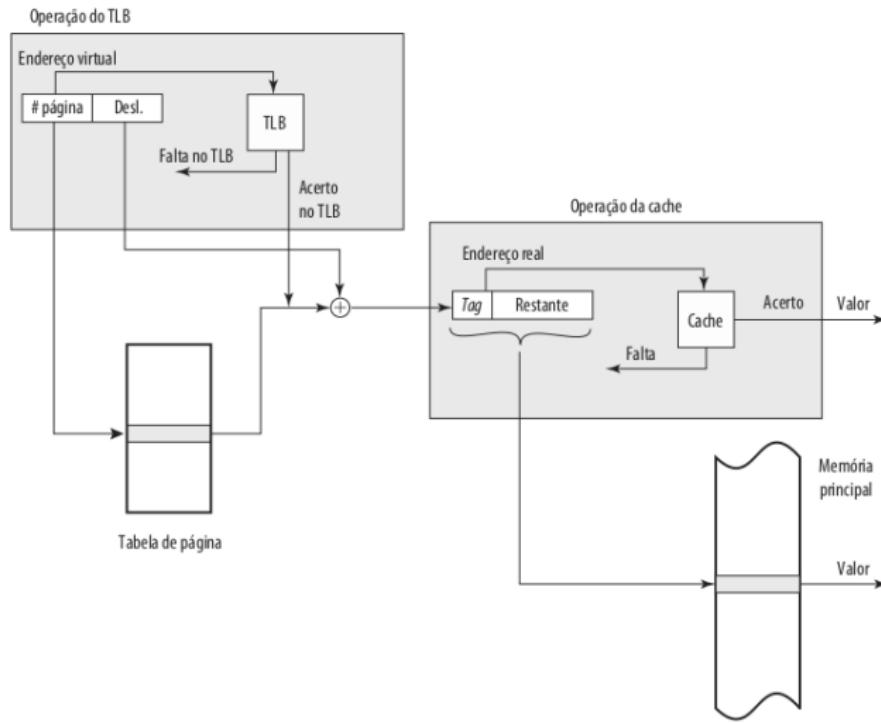
Figura 8.18 Operação da paginação e do *translation lookaside buffer* (TLB)



Gerencia de Memória: TLB

Em caso de acerto na TLB, antes de acessar a memória principal, a palavra pode estar na cache

Figura 8.19 Translation lookaside buffer (TLB) e operação da cache



Agradecimentos

Agradecimentos Especiais:

Agradeço a toda a comunidade L^AT_EX.
Em especial a *Till Tantau* pelo *Beamer*.

<https://www.tcs.uni-luebeck.de/mitarbeiter/tantau/>

Desta forma, tornou-se possível a escrita deste material didático.

Exercícios:

Lista de exercícios divulgada no Moodle

