

Sistemas Operacionais

Esquemas de Paginação e Segmentação

Lesandro Ponciano

2024

Objetivos da Aula

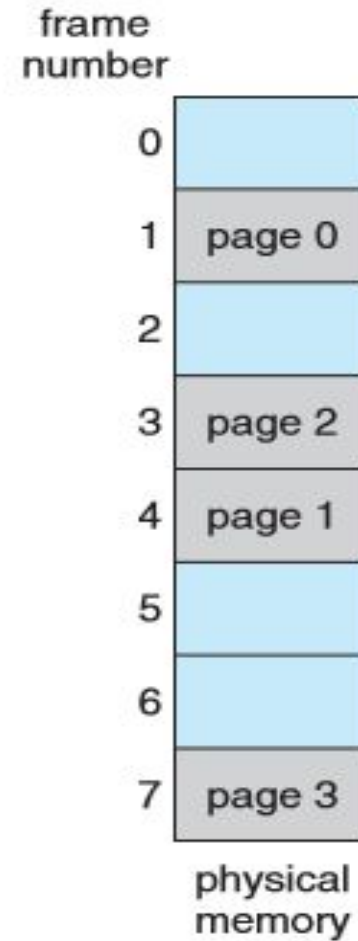
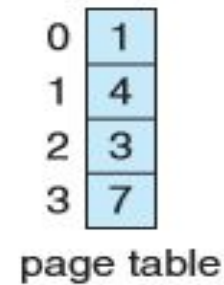
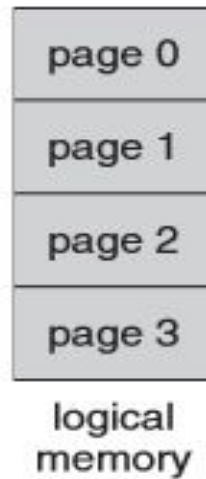
- Analisar
 - Paginação
 - Segmentação

Paginação

- É um **esquema de gerenciamento da memória** que permite que o espaço de endereçamento físico de um processo não seja contíguo
 - Um processo pode estar distribuído em diversas "brechas" (ou páginas) na memória
- Evita a fragmentação externa
 - Evita a necessidade de compactação
- Está sujeita a fragmentação interna

Método Básico

- A memória física é dividida em blocos de tamanho fixo, chamados **quadros** (*frames*)
- A memória lógica é dividida em blocos do mesmo tamanho, chamados **páginas** (*pages*)
- Quando um processo está para ser executado, suas páginas são carregadas em quaisquer quadros de memória disponíveis
 - Para cada processo, há uma tabela de páginas que faz o mapeamento entre páginas e quadros



- O hardware define o tamanho da página e o tamanho do quadro
 - Utiliza-se potência de 2

Endereçamento

- Cada endereço gerado pela CPU é um número binário dividido em duas partes:
 - um número de páginas (p), que é usado como índice na tabela de páginas, que contém o endereço base de cada página na memória física
 - um deslocamento de página (d), que é combinado com o endereço base para definir o endereço na memória física

Endereço Lógico

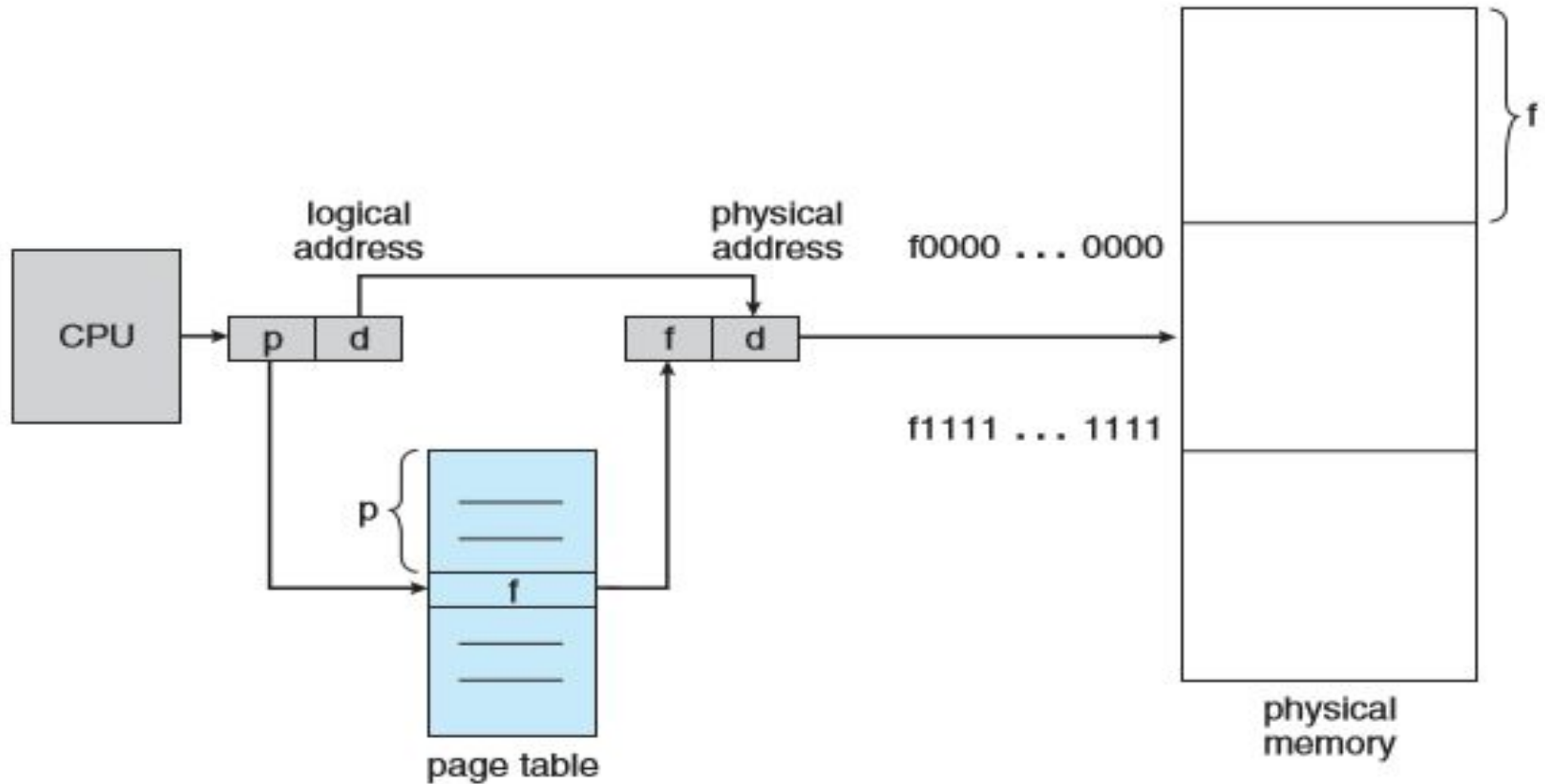


- p é o índice da tabela de páginas
- d é o deslocamento dentro da página

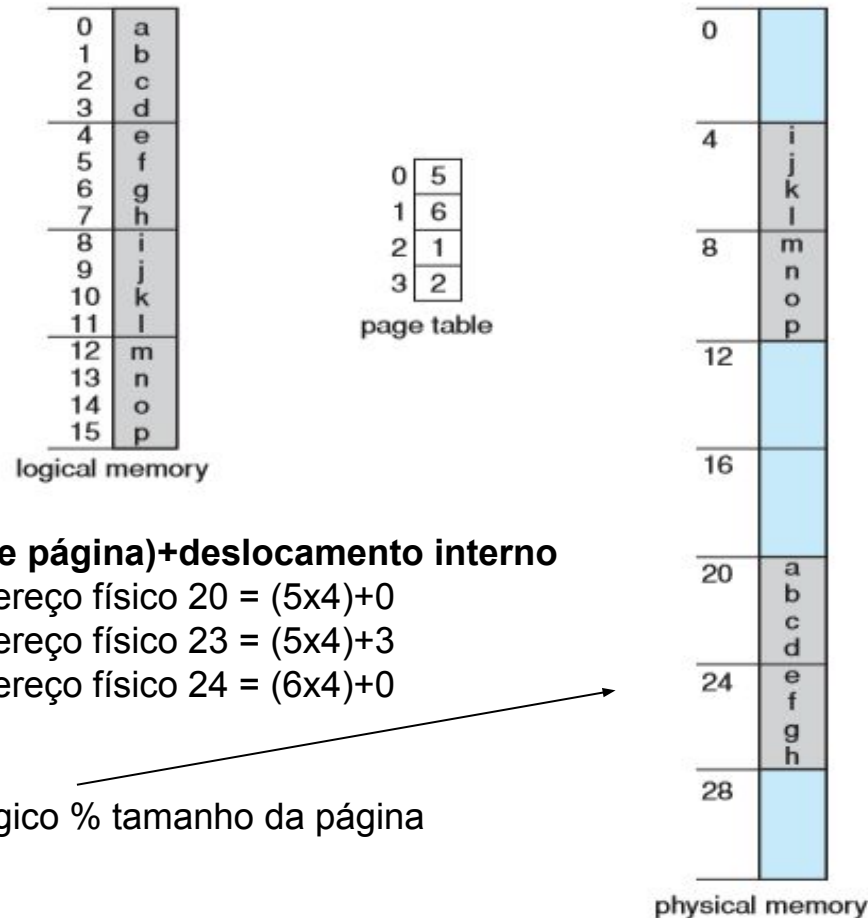
Endereçamento

- Endereços lógicos de 16 bits
 - Endereços lógicos de 0 até $2^{16} = 65536 = 64 \text{ K}$
 - Página = 1 Byte => 64 KB de memória lógica
- 32 KB de RAM física
 - $32 \text{ KB} = 32768 = 2^{15} \text{ Bytes}$
- Páginas e quadros de 4 KB
 - Tem-se $64 \text{ KB} / 4 \text{ KB} = 16$ páginas, ou seja 2^4
 - Precisa-se de 4 bits para endereçar as páginas
 - Tem-se $32 \text{ KB} / 4 \text{ KB} = 8$ quadros, ou seja 2^3
 - Precisa-se de 3 bits para endereçar os quadros

Hardware de Paginação



Paginação para uma memória de 32 bytes com páginas de 4 bytes



Exemplos (valor na tabela x tamanho de página)+deslocamento interno

- Endereço lógico 0 é mapeado para o endereço físico 20 = $(5 \times 4) + 0$
- Endereço lógico 3 é mapeado para o endereço físico 23 = $(5 \times 4) + 3$
- Endereço lógico 4 é mapeado para o endereço físico 24 = $(6 \times 4) + 0$

Deslocamento interno = $\text{Endereço lógico} \% \text{ tamanho da página}$

Atividade de Fixação 1

0	a
1	b
2	c
3	d
4	e
5	f
6	g
7	h
8	i
9	j
10	k
11	l
12	m
13	n
14	o
15	p

logical memory

0	5
1	6
2	1
3	2

page table

0	
4	i j k l
8	m n o p
12	
16	
20	a b c d
24	e f g h
28	

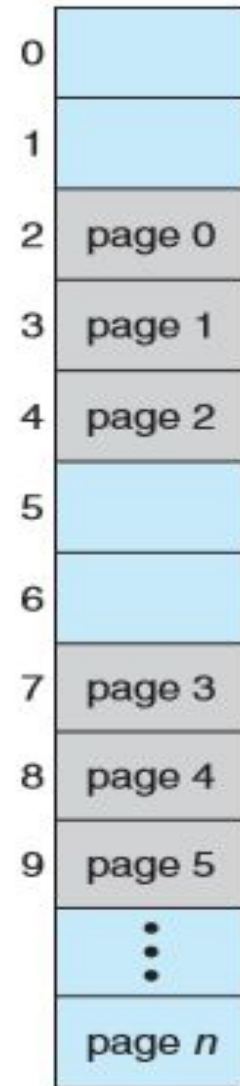
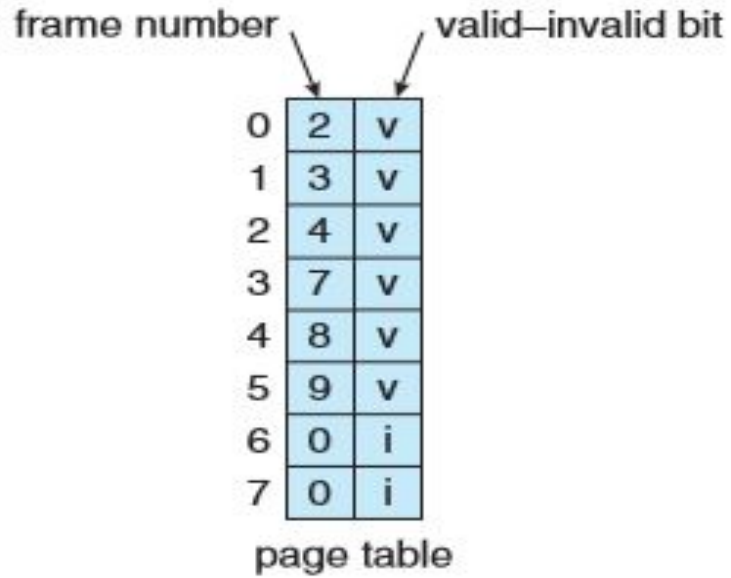
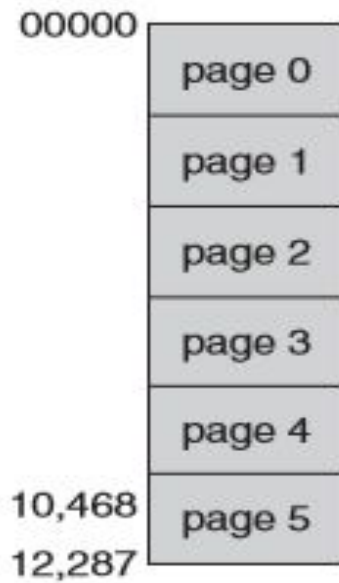
physical memory

Mostre o cálculo

- Endereço lógico 6 é mapeado para o endereço físico?
- Endereço lógico 10 é mapeado para o endereço físico?
- Endereço lógico 13 é mapeado para o endereço físico?

Proteção

- Geralmente adiciona-se uma informação adicional (1 bit) a cada entrada da tabela de páginas, chamado bit **válido - inválido**
 - Quando o bit é posicionado como "**válido**", a página associada está no espaço de endereçamento lógico do processo e, portanto, é uma página legal (ou válida)
 - Quando o bit é posicionado como "**inválido**", a página não está no espaço de endereçamento lógico do processo



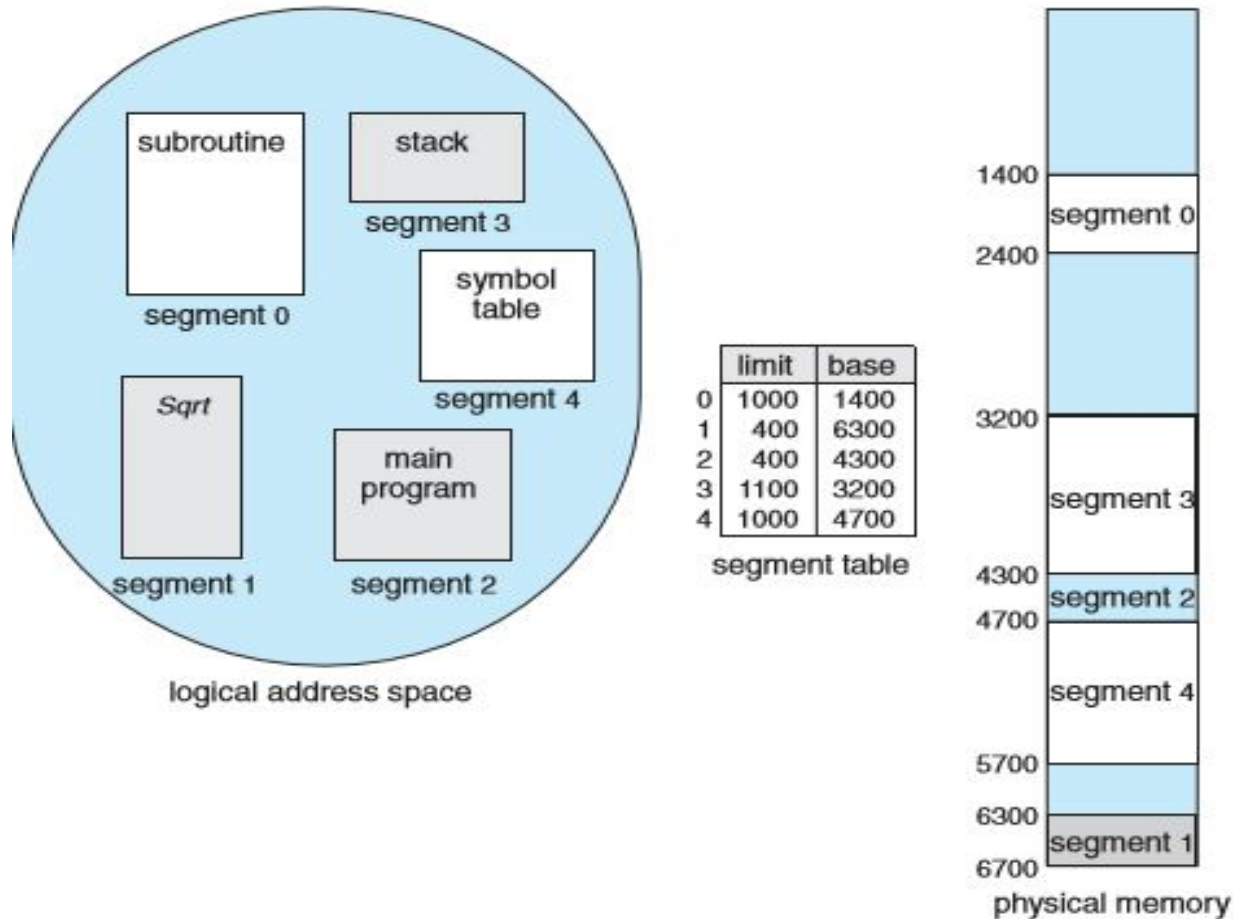
Segmentação

- Segmentação é um **esquema de gerenciamento da memória** pelo qual um espaço de endereçamento lógico é um conjunto de segmentos
- O compilador geralmente constrói segmentos para os seguintes elementos
 - O código; Variáveis globais; Heap; Pilha; Bibliotecas
- Segmentos podem crescer e diminuir independentemente, sem afetar os demais
 - Não gera fragmentação interna, mas pode gerar fragmentação externa

Suporte de Hardware

- Há uma tabela de segmentos que mapeia segmentos em endereços físicos
- Cada entrada na tabela de segmento tem
 - Base de segmento: endereço físico inicial em que o segmento reside
 - Limite de segmento: tamanho do segmento

Exemplo de Segmentação



Atividade de Fixação 2

- Dada a tabela de segmentos abaixo (com 5 segmentos), desenhe a locação dos segmentos na memória física e responda se há ou não fragmentação interna e externa na memória física

	Limite	Base
0	1000	1300
1	400	6200
2	400	4200
3	1100	3100
4	1000	4600

Referências

SILBERSCHATZ, Abraham; GALVIN, Peter B.; GAGNE, Greg. Fundamentos de sistemas operacionais: princípios básicos. Rio de Janeiro, RJ: LTC, 2013. xvi, 432 p. (Capítulo 7)

TANENBAUM, Andrew S. Sistemas operacionais modernos. 3. ed. São Paulo: Pearson Prentice Hall, 2009. xvi, 653 p. ISBN 9788576052371