

Sistemas Operacionais

Marcos Grillo

Literatura

- ▶ MACHADO, Francis Berenger; MAIA, Luiz Paulo (orgs.). **Arquitetura de Sistemas Operacionais**. 4ª ed. Rio de Janeiro: LTC - Livros Técnicos e Científicos, 2008

Programa Livro-Texto.

Conteúdo Programático

Conceitos básicos de sistemas operacionais, uma visão geral:

Sistemas Monoprogramáveis/Monotarefa,

Sistemas Multiprogramáveis/Multitarefa,

Sistemas com Múltiplos processadores,

Sistemas Fortemente acoplados,

Sistemas Fracamente acoplados.

Estrutura do Sistema Operacional

Processo:

Modelo de processo, estados, mudanças de estados,

Subprocesso e Thread,

Tipos de processos.

Comunicação entre processos

Especificação de concorrência em programas,

Problemas de compartilhamento de recursos,

Problemas de sincronização,

Deadlock.

Gerência do Processador:

CrITÉRIOS de Escalonamento,

Escalonamento Não-preenptivo,

Escalonamento Preenptivo,

Escalonamento com Múltiplos Processadores

Gerência de Memória:

Alocação Contígua Simples,

Alocação Particionada,

Memória Virtual,

Segmentação, segmentação com paginação,

Proteção,

Compartilhamento de memória.

Sistema de Arquivos:
Organização de Arquivos,
Métodos de acesso, operações de I/O e Atributos,
Diretórios,
Alocação de espaço em disco,
Proteção de acesso,
Implementação de Cachês.
Gerência de Dispositivos:
Operações de I/O,
Subsistemas de I/O,
Device Drivers,
Controladores,
Dispositivos de Entrada/Saída

Ementa – 1ª etapa.

- Introdução a sistemas operacionais;
- Visão geral de sistemas operacionais;
- Conceitos básicos de SO, hardware e software; Comunicação;
- Estrutura do Sistema Operacional;
- Tipos de processos, processos e Threads;
- Processos e Threads;
- Sincronização e comunicação entre processos e threads;
- Revisão, exercícios, seminários;

Ementa - 2ª etapa.

- Gerência do processador;
- Gerência de memória;
- Gerência de dispositivos;
- Sistemas com múltiplos processadores;
- Sistemas operacionais comerciais/Livre;
- Prova escrita oficial;
- Revisão;
- Prova Substitutiva;

Horários.

- ▶ 1ª aula 19:10 – 20:00
- ▶ 2ª aula 20:00 – 20:50
- ▶ 3ª aula 21:10 – 22:00
- ▶ 4ª aula 22:00 – 22:50 – Orientação ATPS

Avaliação.

- ▶ 1º Bimestre peso 4;
 - ▶ Prova + ATPS

- ▶ 2º Bimestre peso 6;
 - ▶ Prova + ATPS

Gerência de memória.

- ▶ Programas em execução necessitam que os dados acessados pelos seus processos estejam em memória, porém a memória principal é mais cara, assim o sistema operacional deve gerenciar da melhor forma possível os acessos a este espaço, compartilhando com diferentes processos e ainda mantendo **integridade** e **segurança**;
- ▶ Base da multiprogramação.

Gerência de memória, principais características.



- ▶ Diminuir ao máximo a necessidade de acesso à memória secundária durante a execução de um processo;
- ▶ Maximizar o número de processos residentes na memória principal;
- ▶ Permitir a criação e execução de novos processos, mesmo que não haja mais espaço livre na memória principal (*swapping*: transferência temporária de processos para a memória secundária).

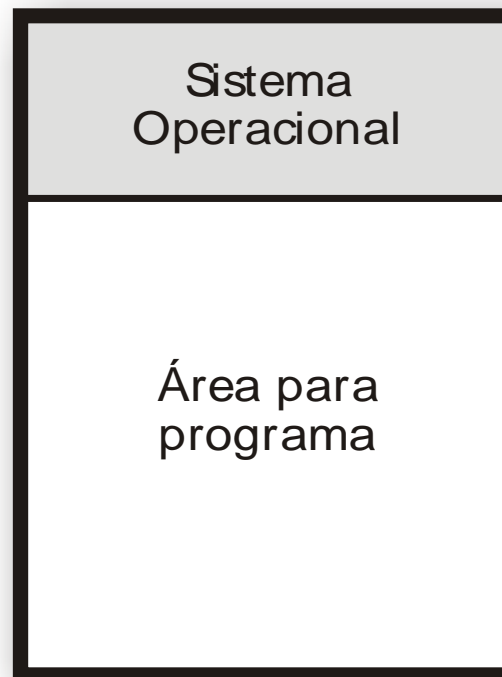
Gerência de memória, principais características.



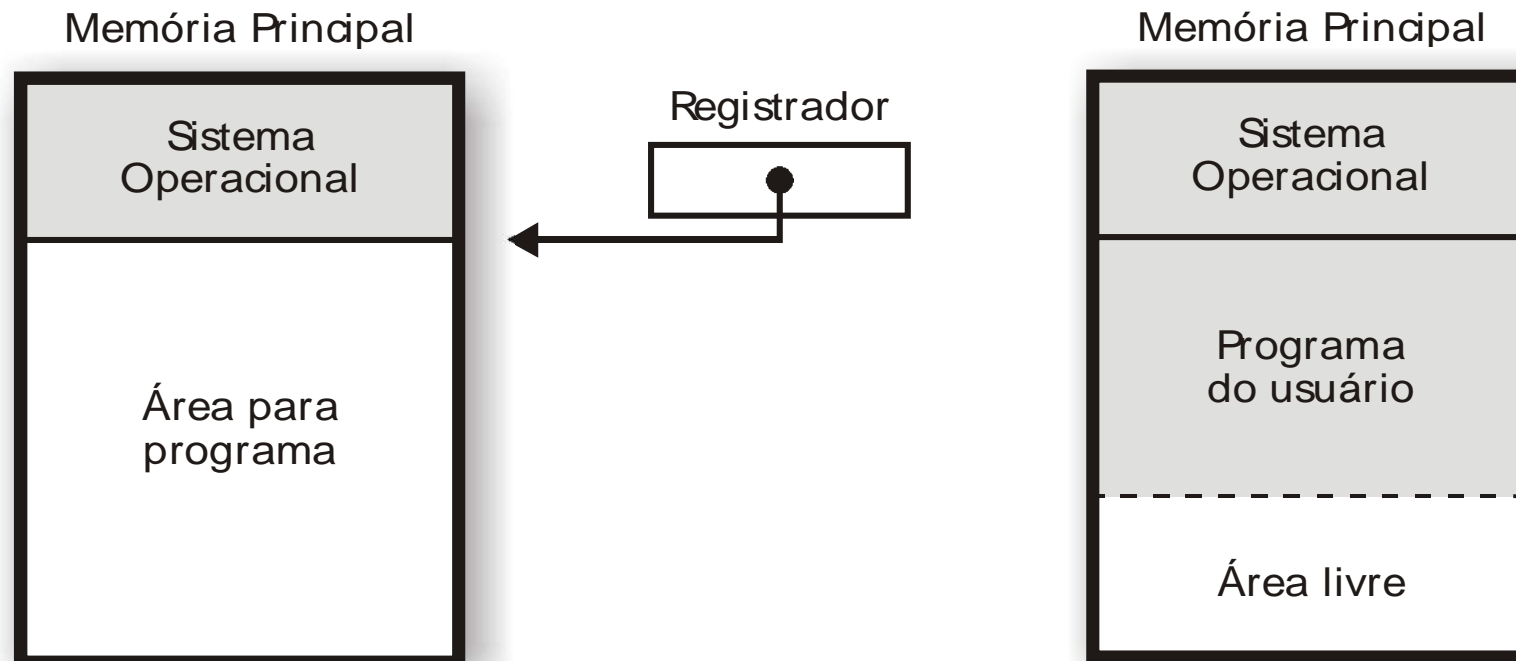
- ▶ Divisão da memória em duas partes: uma para o S.O. e uma para processos do usuário;
- ▶ Necessidade de um registrador contendo limite de endereço da memória para proteção da área do sistema (registrador de fronteira);
- ▶ Sempre que o programa do usuário faz referência a um endereço de memória, ocorre a comparação com esse registrador para saber se o endereço acessado faz parte da área do usuário.

Alocação Contígua Simples

Memória Principal



Alocação Contígua Simples



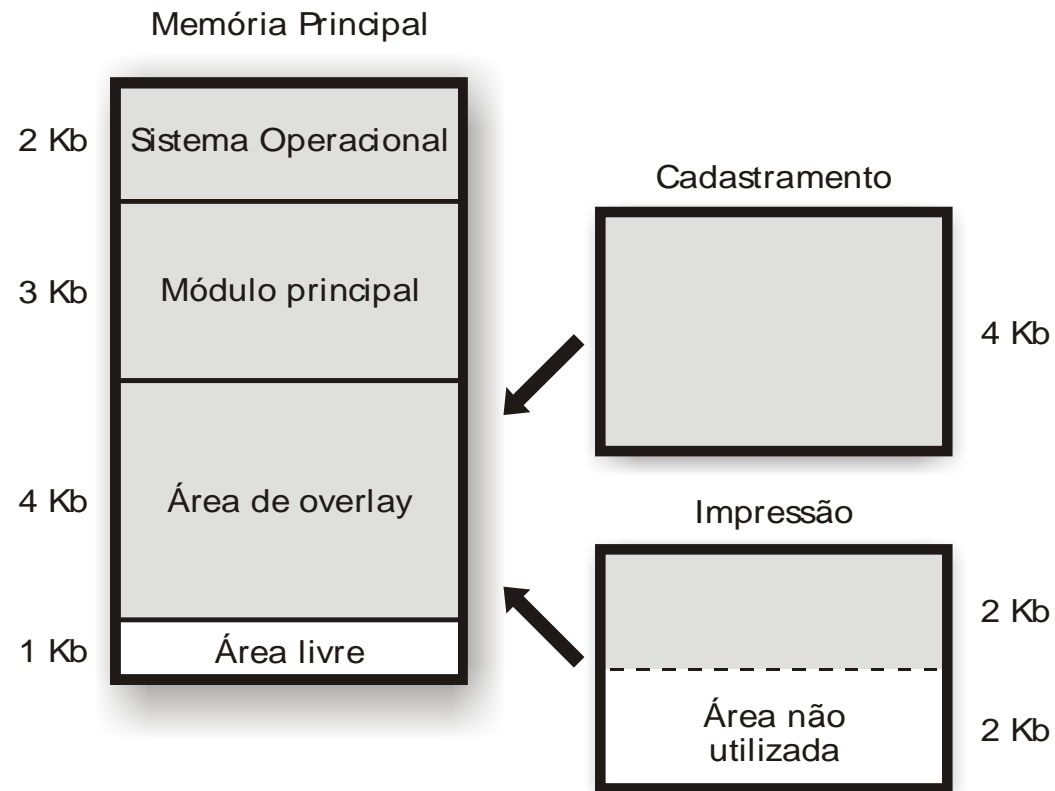
Overlay:

- ▶ A definição das áreas de *overlay* é função do programador (áreas sempre do tamanho do maior módulo);
- ▶ Risco: transferência excessiva de módulos entre memória principal e memória secundária;
- ▶ Permite ao programador expandir os limites da memória principal;

Overlay:

- ▶ Programador divide em módulos o programa, Ex: um programa principal, um de cadastramento e um de impressão;
- ▶ Se o processo do usuário é maior que o tamanho da memória principal, ele é dividido em módulos;
- ▶ A área da memória compartilhada por esses módulos é chamada de área de overlay.

Técnica de Overlay



Alocação Particionada:

- ▶ Sistemas multiprogramáveis mais eficientes (uso do processador);
- ▶ Precisam de mais programas em memória ao mesmo tempo;
- ▶ Proteção dos programas em memória.

Alocação Particionada Estática ou fixa.

- ▶ Memória dividida em pedaços fixos, na inicialização do sistema;
- ▶ Baseado nos tamanhos dos programas que irão executar no ambiente;
- ▶ Para alterar o tamanho das partições de memória é necessário reiniciar o sistema;

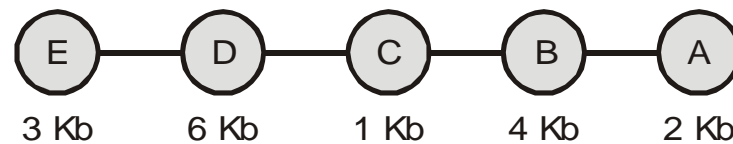
- ▶ Dois tipos:
 - ▶ Partições estáticas absolutas;
 - ▶ Partições estáticas relocável;

Alocação Particionada Estática

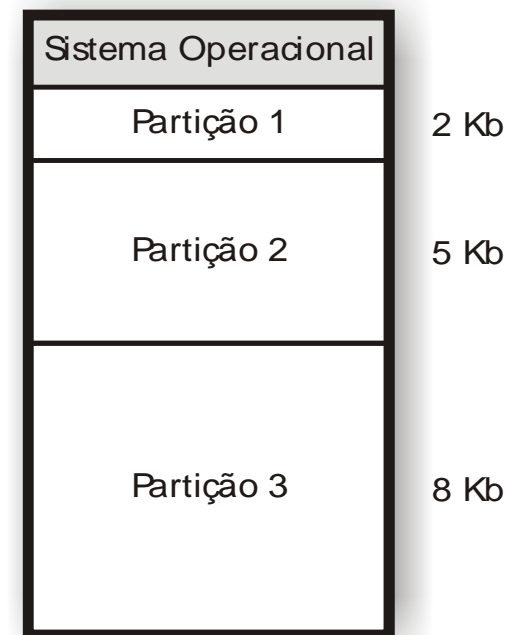
Tabela de partições

Partição	Tamanho
1	2 Kb
2	5 Kb
3	8 Kb

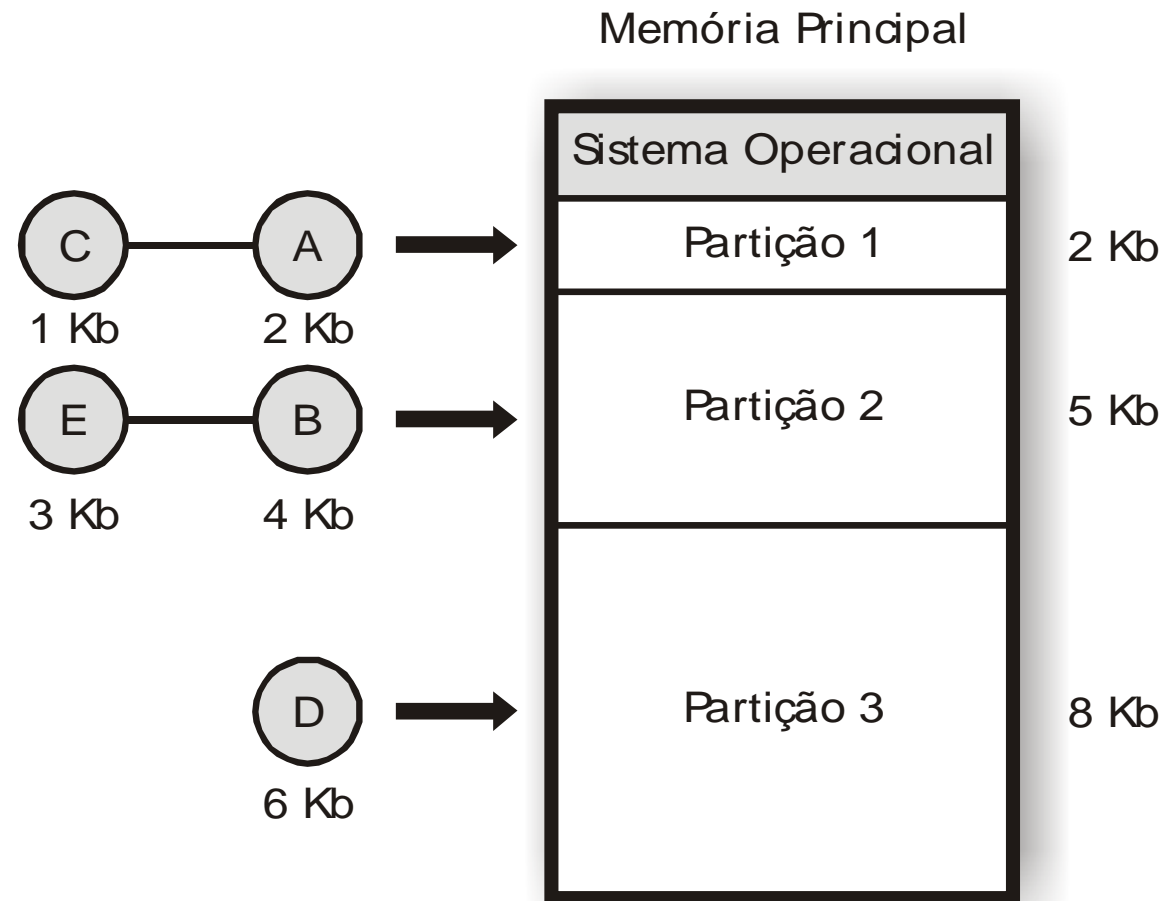
Programas a serem executados:



Memória Principal



Alocação Particionada Estática Absoluta



Alocação Particionada Relocável.

- ▶ Referências são no início do código do programa;
- ▶ *Loader* calcula todos endereços a partir da posição inicial onde o programa foi locado;
- ▶ O programa não é mais carregado a partir de um endereço já previamente definido;
- ▶ Maior combinações de possibilidades de carregamento na memória.

Alocação Particionada Estática Relocável

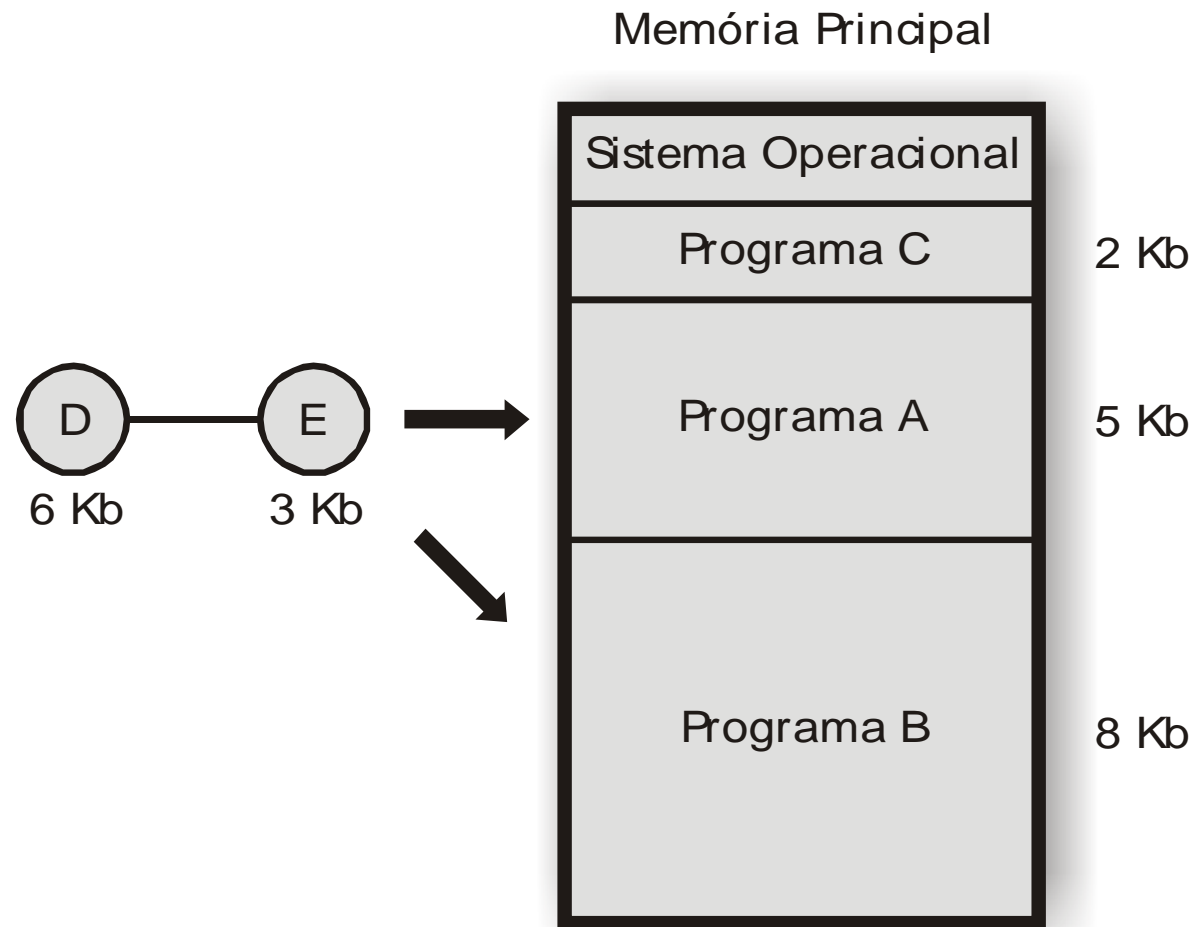
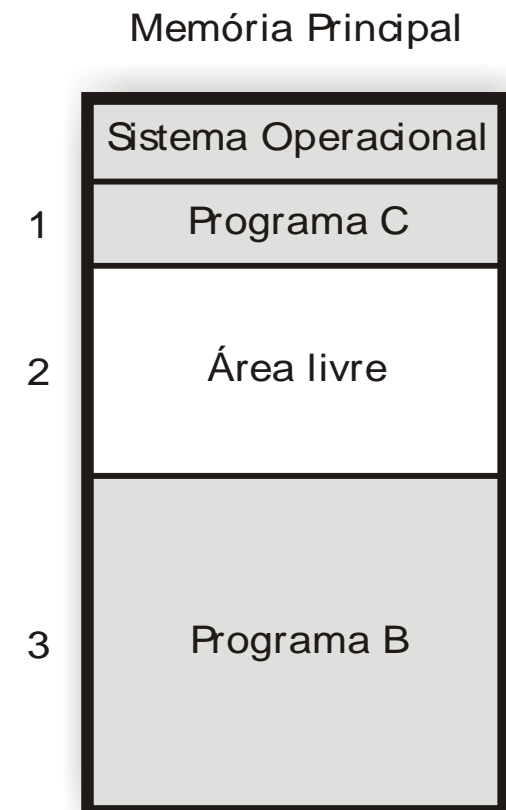


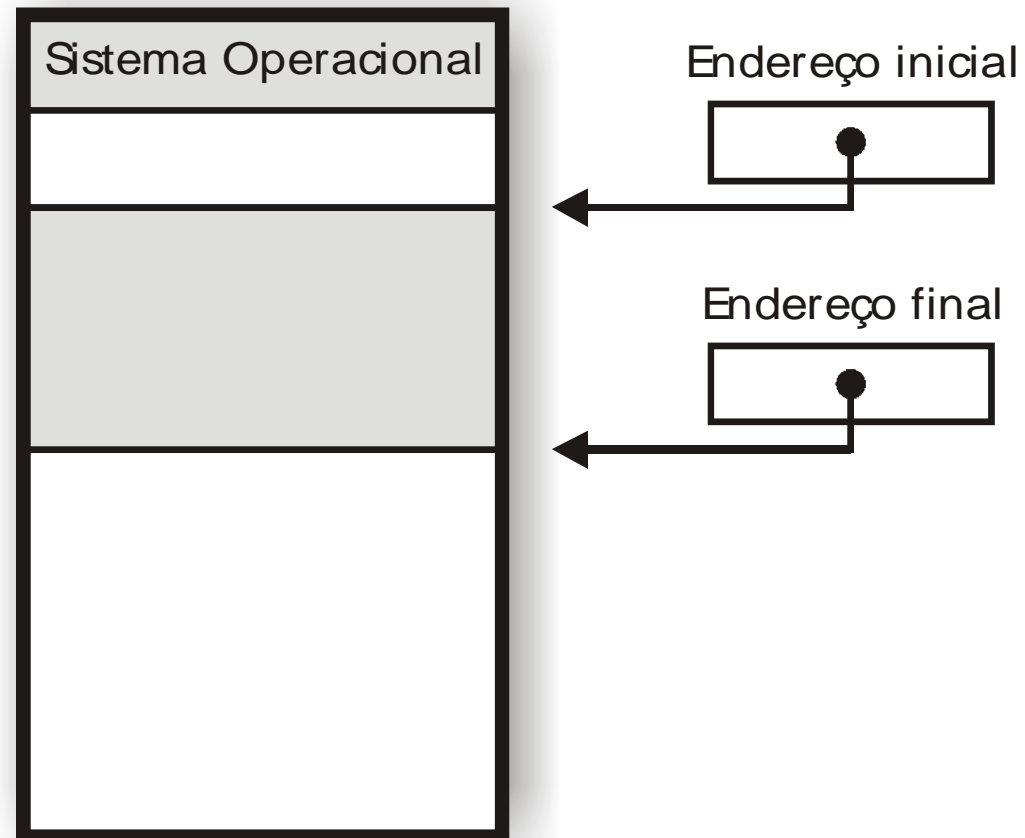
Tabela de Alocação de Partições.

Partição	Tamanho	Livre
1	2 Kb	Não
2	5 Kb	Sim
3	8 Kb	Não



Proteção na Alocação Particionada

Memória Principal



Fragmentação Interna

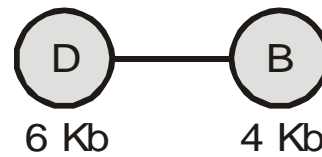
Fragmentação interna:

- ▶ Porção de memória de uma determinada partição que não é utilizada devido aos requisitos reduzidos dos processos.

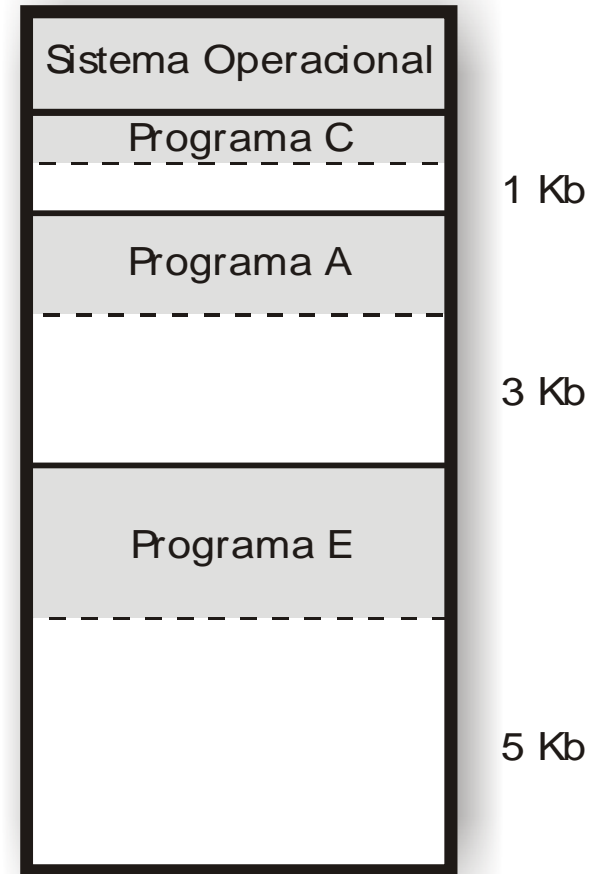
Fragmentação externa:

- ▶ Ocorre quando existe espaço suficiente de memória livre para satisfazer uma requisição, porém **não é contínuo** e portanto não pode ser utilizado.

Fragmentação Interna



Memória Principal

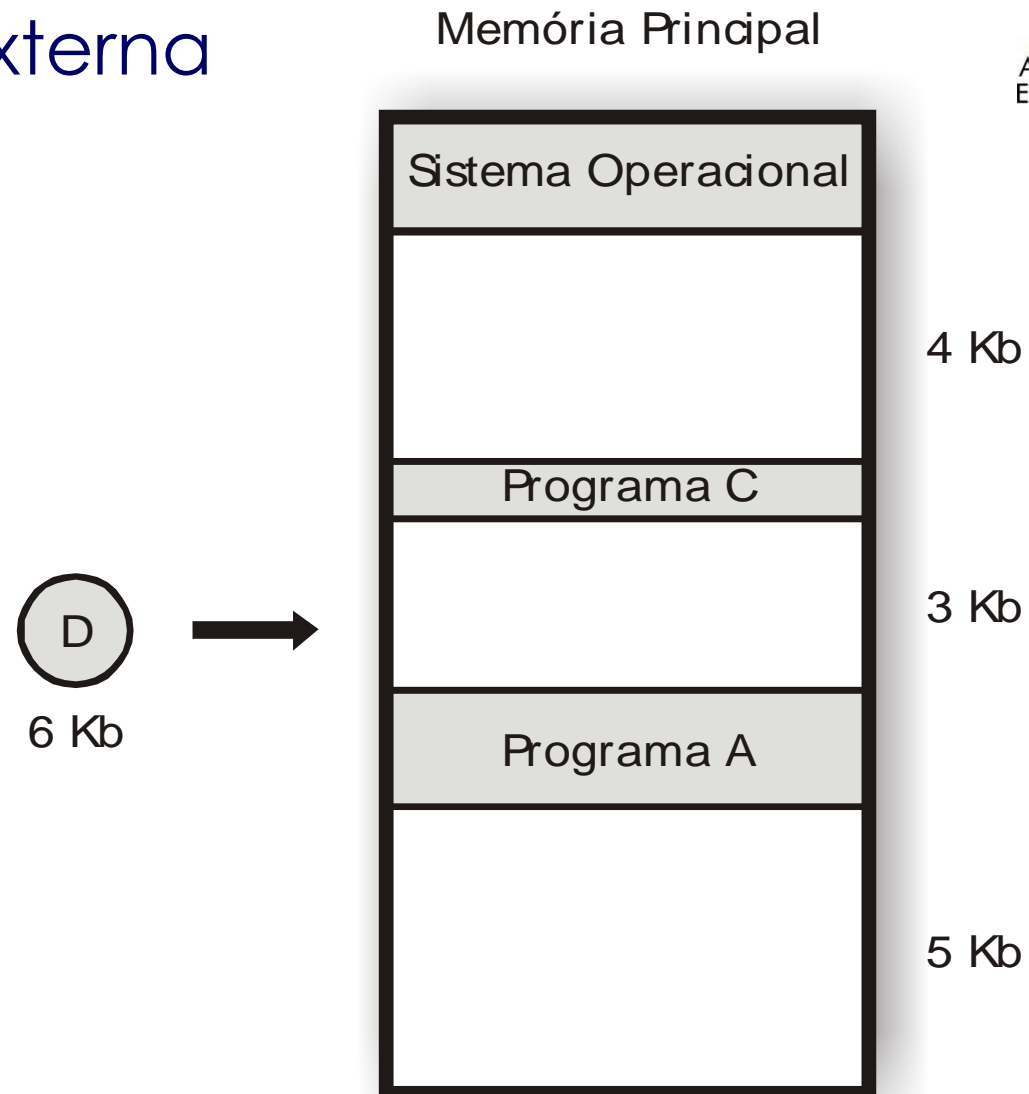


Alocação Particionada Dinâmica

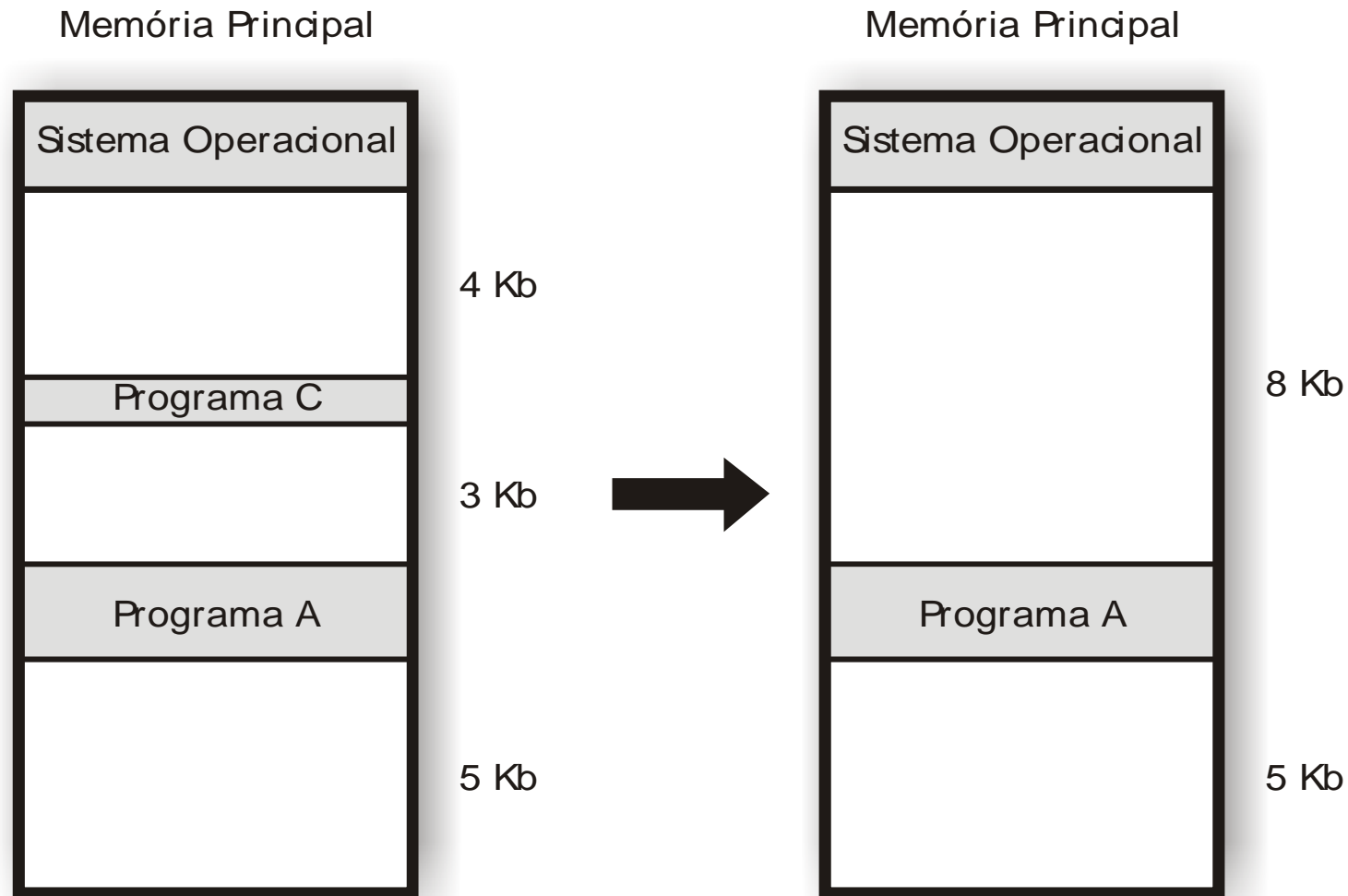
Objetivo: reduzir o problema da fragmentação interna:

- ▶ Cada programa utiliza somente o espaço necessário;
- ▶ Problema: fragmentação externa (várias partições livres pequenas);
- ▶ Possíveis soluções:
 - ▶ Reunir os espaços livres adjacentes;
 - ▶ Reunir as partições ocupadas, criando uma partição livre única;
- ▶ Algoritmo complexo -> consumo de recursos do sistema;
- ▶ Recebe o nome de alocação particionada dinâmica com relocação.

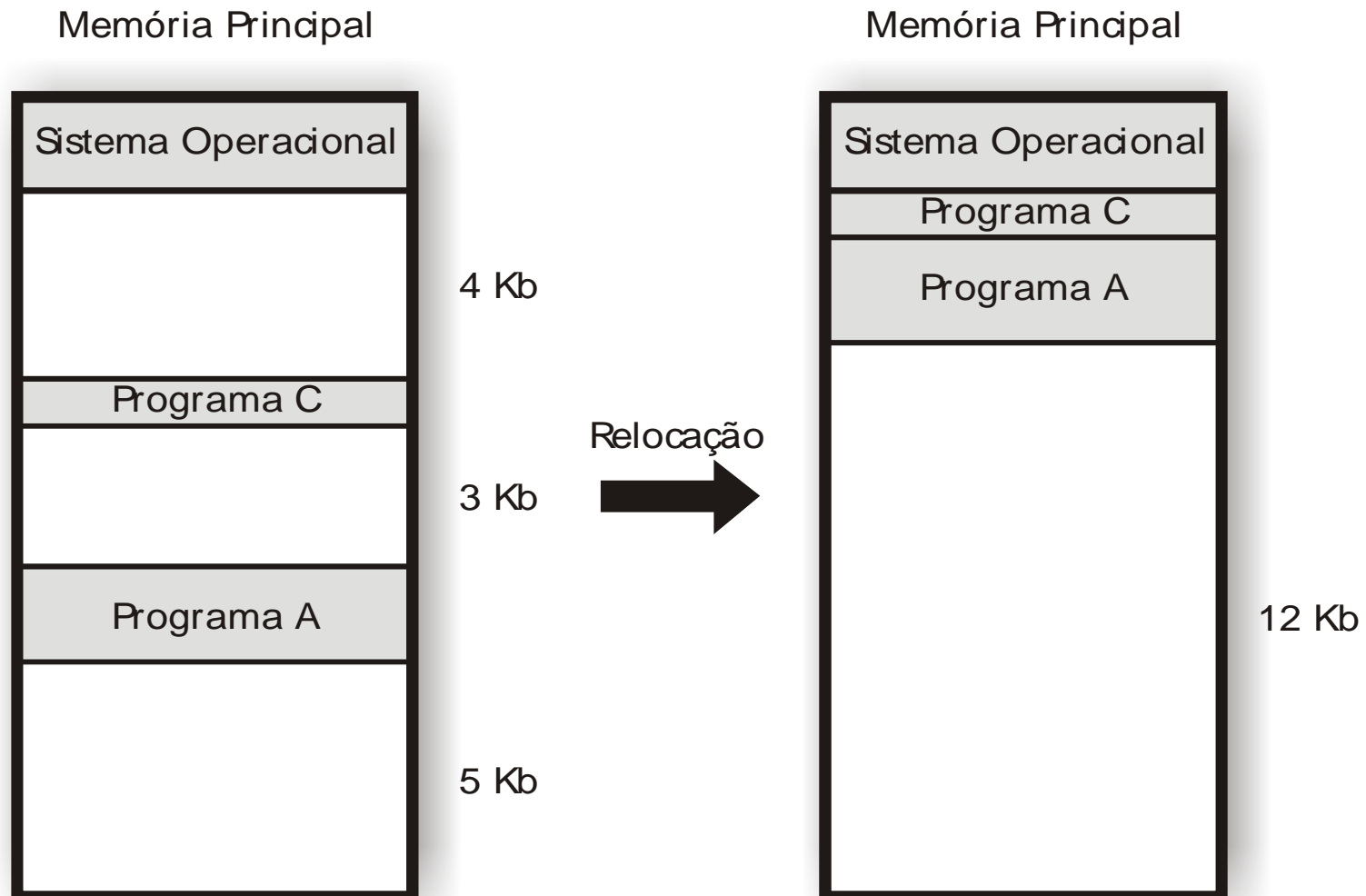
Fragmentação Externa



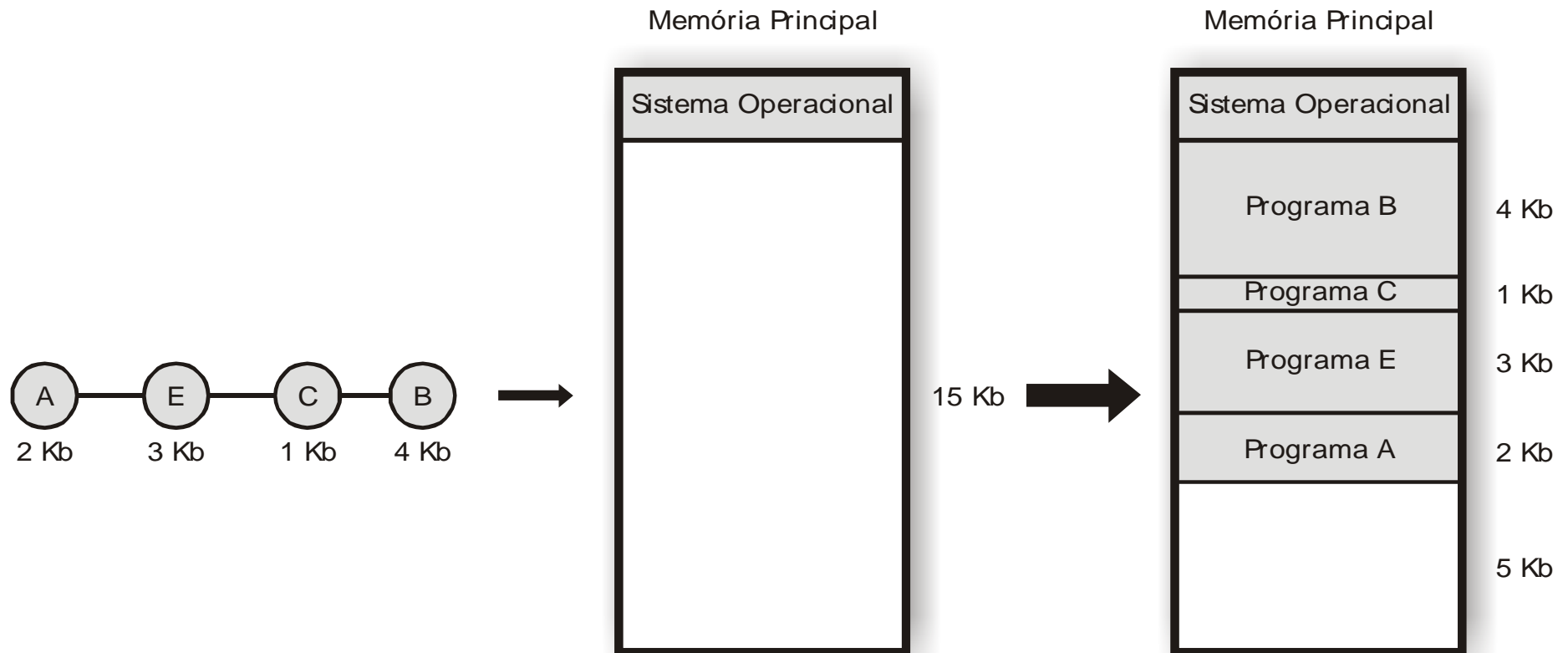
Solução para a Fragmentação Externa



Solução para a Fragmentação Externa



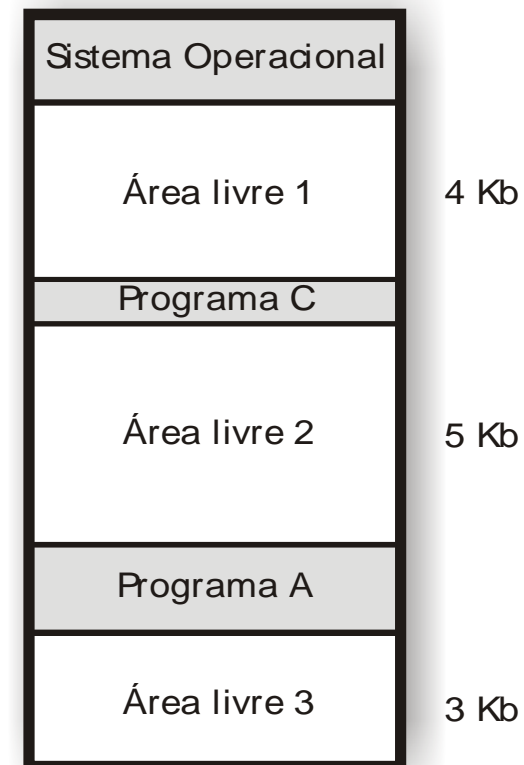
Alocação Particionada Dinâmica



Lista de Áreas Livres

Áreas livres	Tamanho
1	4 Kb
2	5 Kb
3	3 Kb

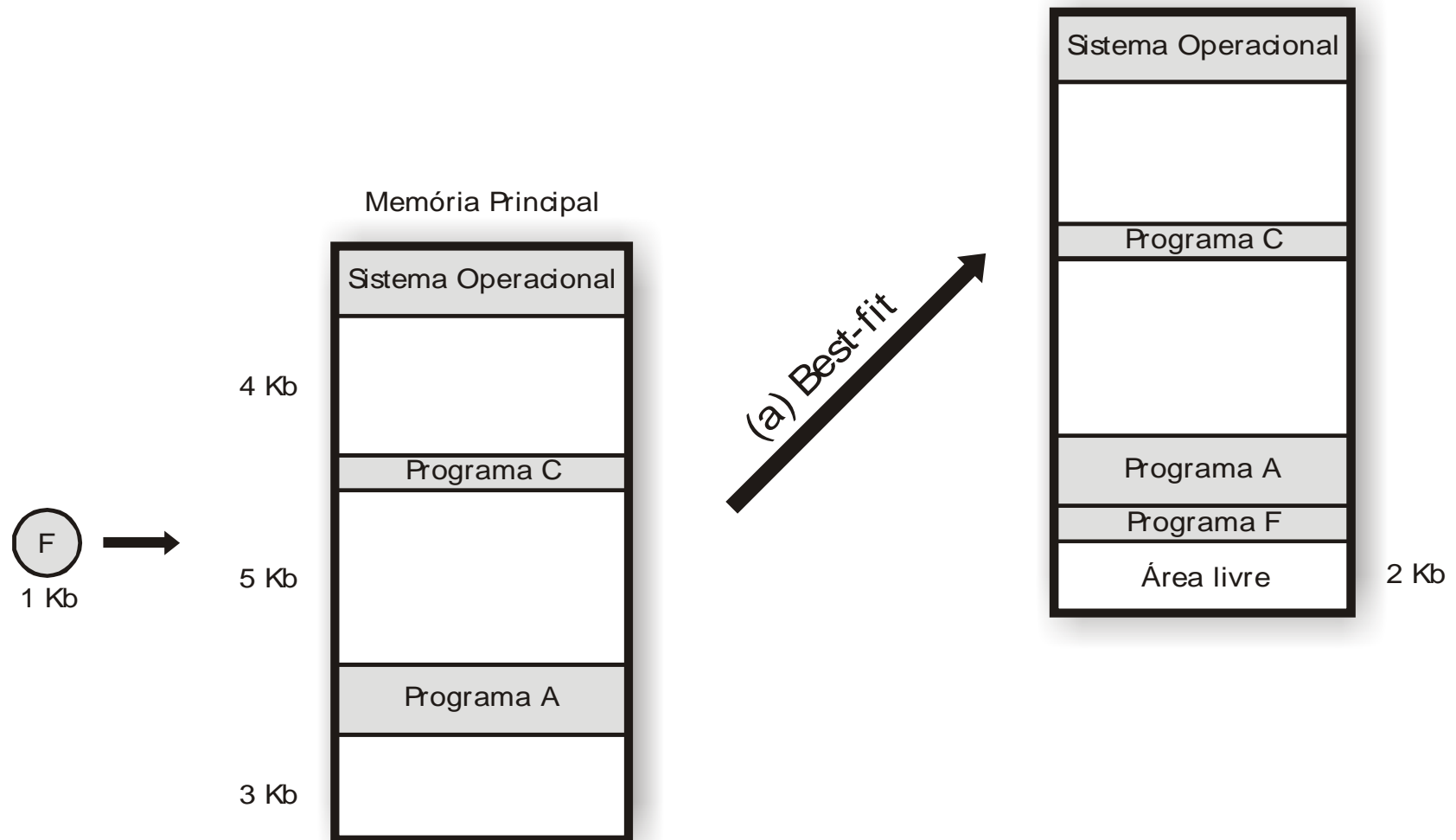
Memória Principal



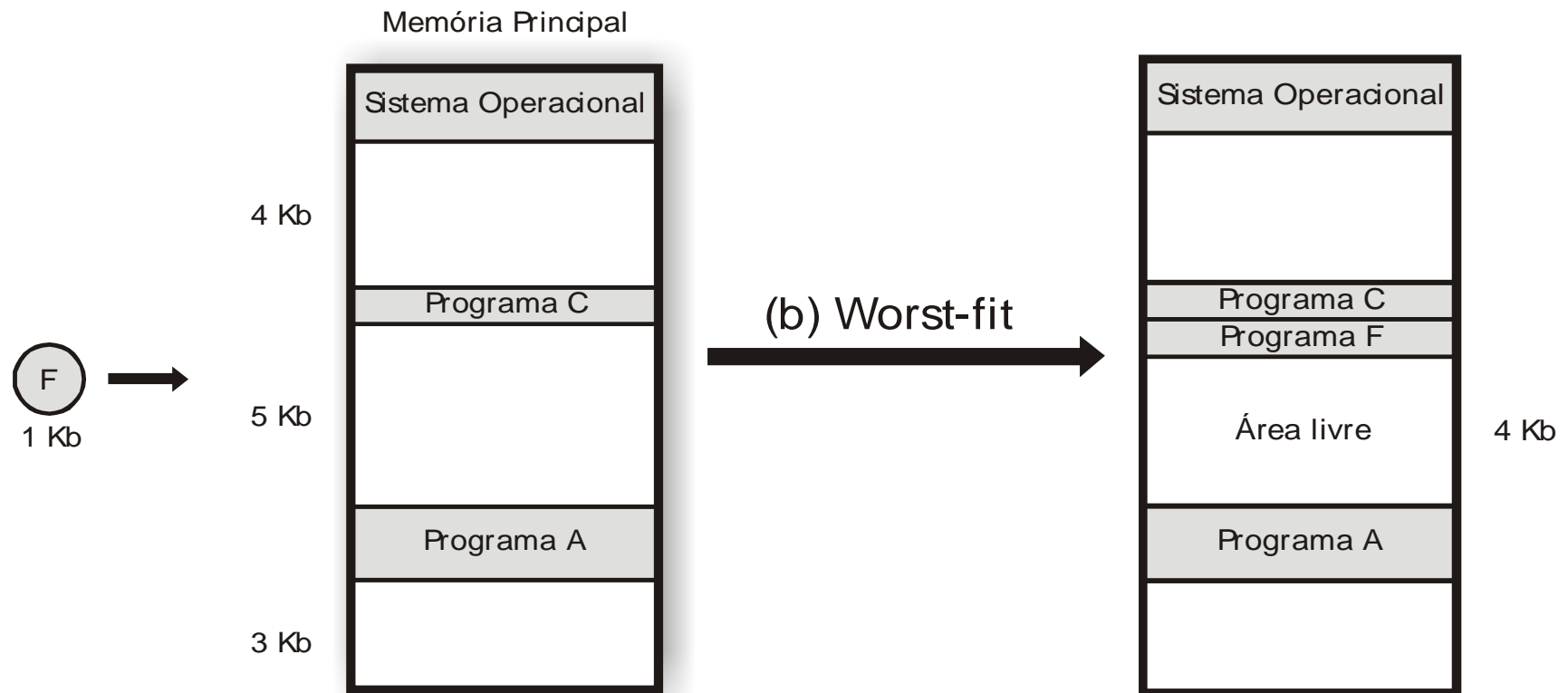
Modos de seleção de espaço para um programa

- ▶ **Best-fit:** é escolhida a partição livre que resultar em menor espaço sem utilização;
 - ▶ Desvantagem: surgimento de pequenas áreas não contíguas;
- ▶ **Worst-fit:** é escolhida a partição que resultar em maior espaço livre;
- ▶ **First-fit:** é escolhida a primeira partição livre com tamanho suficiente;
 - ▶ Estratégia mais simples (menor consumo de recursos).

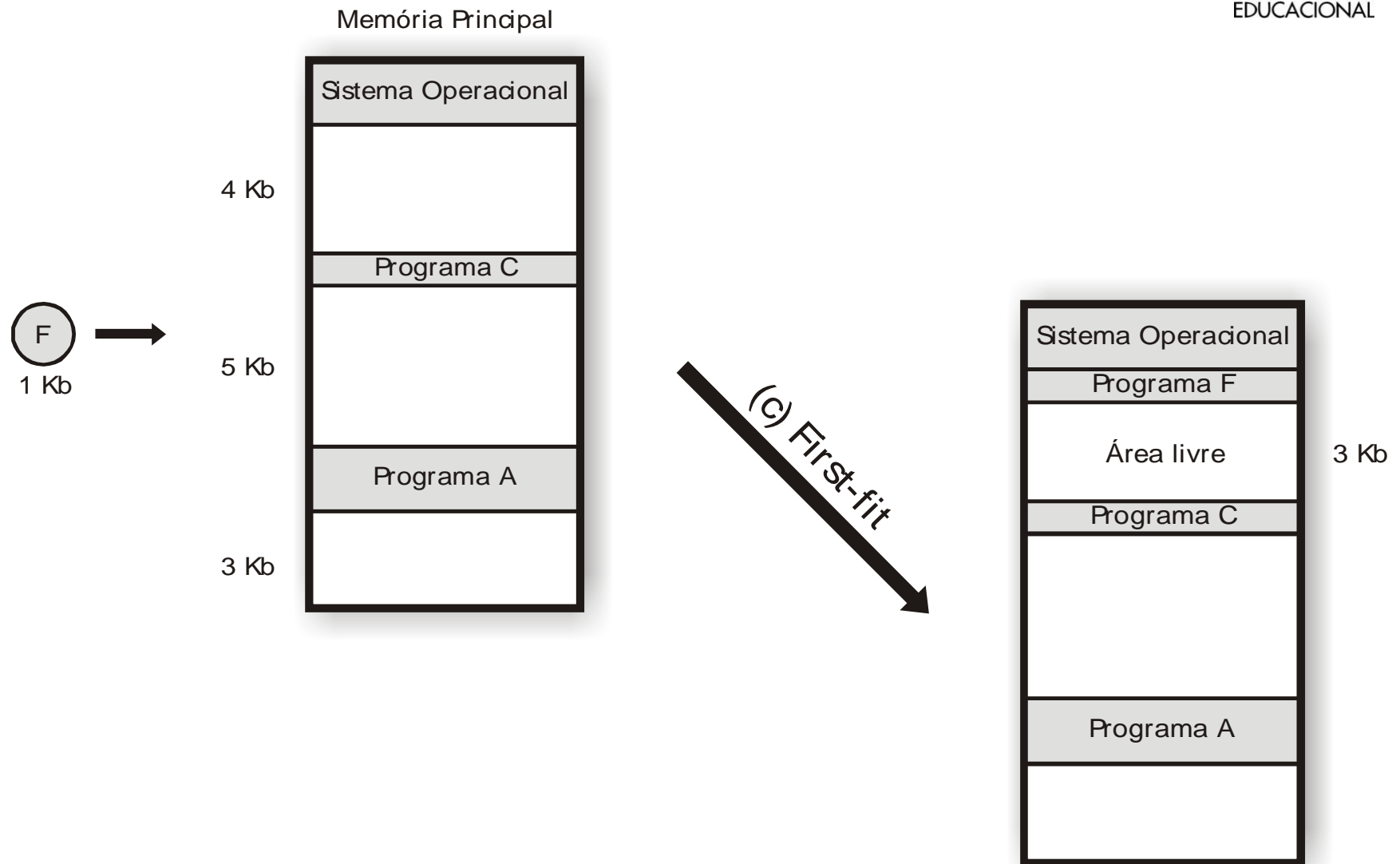
Best-Fit



Worst-Fit



First-Fit



Swapping

- ▶ Técnica utilizada para execução de processos quando não existir partição livre com tamanho suficiente;
- ▶ Retira temporariamente da memória principal processos que não estão sendo executados (em espera ou pronto), colocando-os em disco;
- ▶ Deve ser retirado da memória principal o processo com menor chance de ser executado (menor prioridade);
- ▶ É necessário que o **loader** faça a relocação a cada **swap in**;
- ▶ A técnica de *swapping* é utilizada no gerenciamento da memória virtual.
- ▶ **Problema:** elevado custo das operações de I/O;

Swapping

