

Memória Principal

Componente essencial dentro da arquitetura de "programa armazenado", de John von Neumann.

Funções do Gerente de Memória

Controlar quais as unidades de memória estão ou não estão em uso, para que sejam alocadas quando necessário;

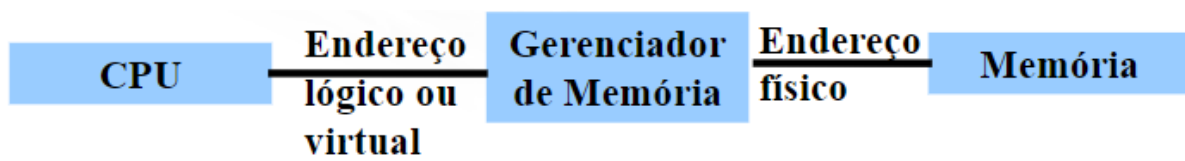
Liberar as unidades de memória que foram desocupadas por um processo que finalizou;

Tratar do Swapping entre memória principal e memória secundária (Transferência temporária de processos residentes na memória principal para memória secundária);

Gerência de Memória

Memória Lógica / Virtual - é aquela que o processo enxerga, o processo é capaz de acessar.

Memória Física - é aquela implementada pelos circuitos integrados de memória, pela eletrônica do computador (memória real, RAM)



Espaço de Endereços de um Programa

- Código absoluto:
 - Não possui memória virtual (por causa da área que a MMU ocupa).
 - Atribuição de endereços pode ocorrer diretamente (edição, na compilação ou na ligação).
 - Endereços relativos ao início da memória (endereços reais)
 - Gerado quando a localização do processo na memória é conhecida a priori
- Código relocável
 - O programa pode ser carregado em qualquer posição da memória.
 - Deve haver uma tradução de endereços (ou relocação de endereços)
 - No caso de relocação de endereços 'estática', a tradução é feita toda de uma vez quando este é carregado em memória/ quando o processo é criado.
 - No caso de relocação de endereços 'dinâmica', o código é mantido em memória contendo os endereços todos "lógicos". A tradução dos endereços é feita no momento em que um endereço é referenciado: ele é traduzido em tempo real (pela MMU) para um endereço físico.

Técnicas de Gerência de Memória Real

- Alocação Contígua Simples
 - A memória é dividida em duas áreas e usuário não pode usar uma área maior do que a disponível;
 - Não permite utilização eficiente de processador/memória;
 - Área de Overlay (área de memória comum onde módulos compartilham o mesmo espaço);



Alocação Particionada

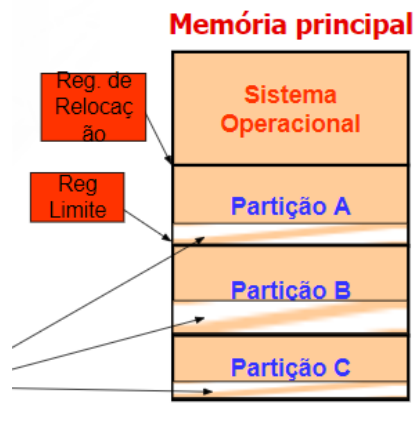
- Memória dividida em pedaços de tamanho fixo chamados partições;
- alteração do particionamento, era necessário uma nova inicialização com uma nova configuração;

Alocação Particionada Estática

Programas não ocupam totalmente o espaço das partições, gerando uma fragmentação interna.

1. Alocação Particionada Estática Absoluta:
 - Código absoluto;
 - Programas exclusivos para partições específicas.
 - Simples de gerenciar
2. Alocação Particionada Estática Relocável:
 - Código relocável

- Programas podem rodar em qualquer partição



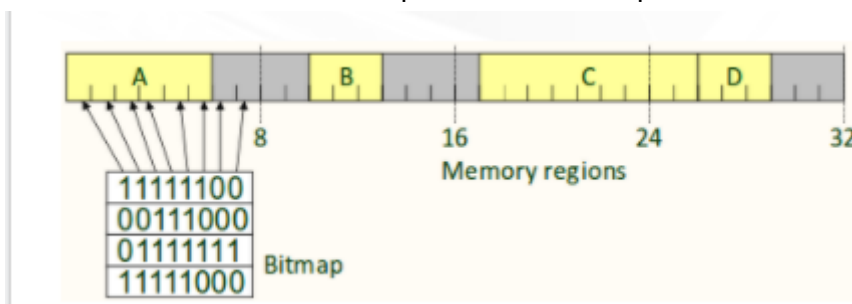
Alocação Particionada Dinâmica

- O espaço utilizado por um programa é a sua partição;
- Não ocorre fragmentação interna (o tamanho da memória alocada é igual ao tamanho do programa);
- Ao terminarem, os programas deixam espalhados espaços pequenos de memória, provocando a fragmentação externa (pequeno demais para serem reaproveitados);



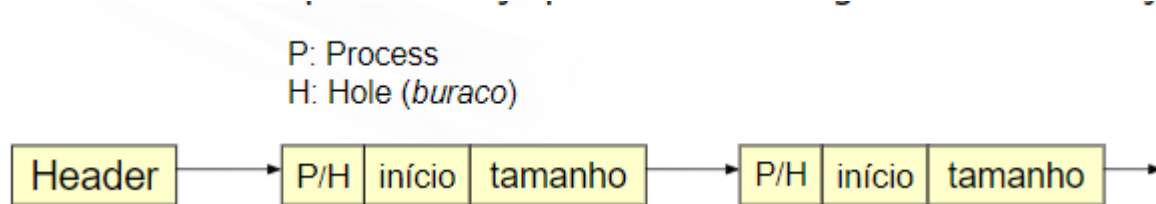
Gerenciamento de Espaço Livre -> Mapa de bits

- Usado para o gerenciamento com alocação dinâmica;
- Memória é dividida em unidades de alocação (de algumas palavras a vários kilobytes);
 - Qto menor → maior o mapa de bits
 - Qto maior → desperdício na última unidade
- A cada unidade é associado um bit que descreve a disponibilidade da unidade



Gerenciamento de Espaço Livre -> Mapeamento da Memória com lista encadeada

- Lista ligada de segmentos alocados ou livres (um segmento é uma área de memória alocada ou livre);
- Cada elemento da lista indica
 - Estado do segmento (P) Alocado por um processo ou (H) Buraco livre
 - Unidade em que inicia
 - Tamanho em unidades
- Lista duplamente encadeada facilita de concatenação de segmentos
- Lista ordenada por endereço permite vários algoritmos de alocação



Existem 4 maneiras de percorrer a lista de espaços livre atrás de uma lacuna de tamanho suficiente, são eles:

Best-fit (utiliza a lacuna que resultar a menor sobra)

- Espaço mais próximo do tamanho do processo;
- Tempo de busca grande;
- Provoca fragmentação.

Worst-Fit (utiliza a lacuna que resultar na maior sobra):

- Escolhe o maior espaço possível;
- Tempo de busca grande;
- Não apresenta bons resultados.

First-Fit (primeira alocação):

- Utiliza a primeira lacuna que encontrar com tamanho suficiente
- Melhor performance.

Circular-fit ou Next-Fit (próxima alocação):

- Como first-fit mas inicia a procura na lacuna seguinte a última sobra
- Performance inferior ao First-Fit.