

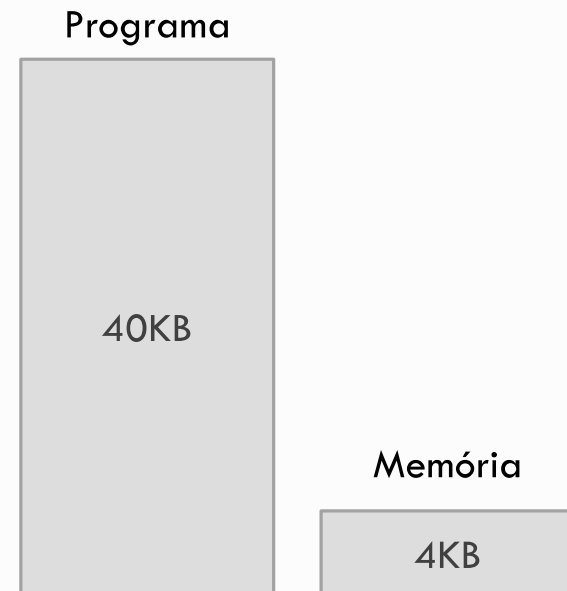
Sistemas Operacionais

Gerência de Memória

Memória Virtual

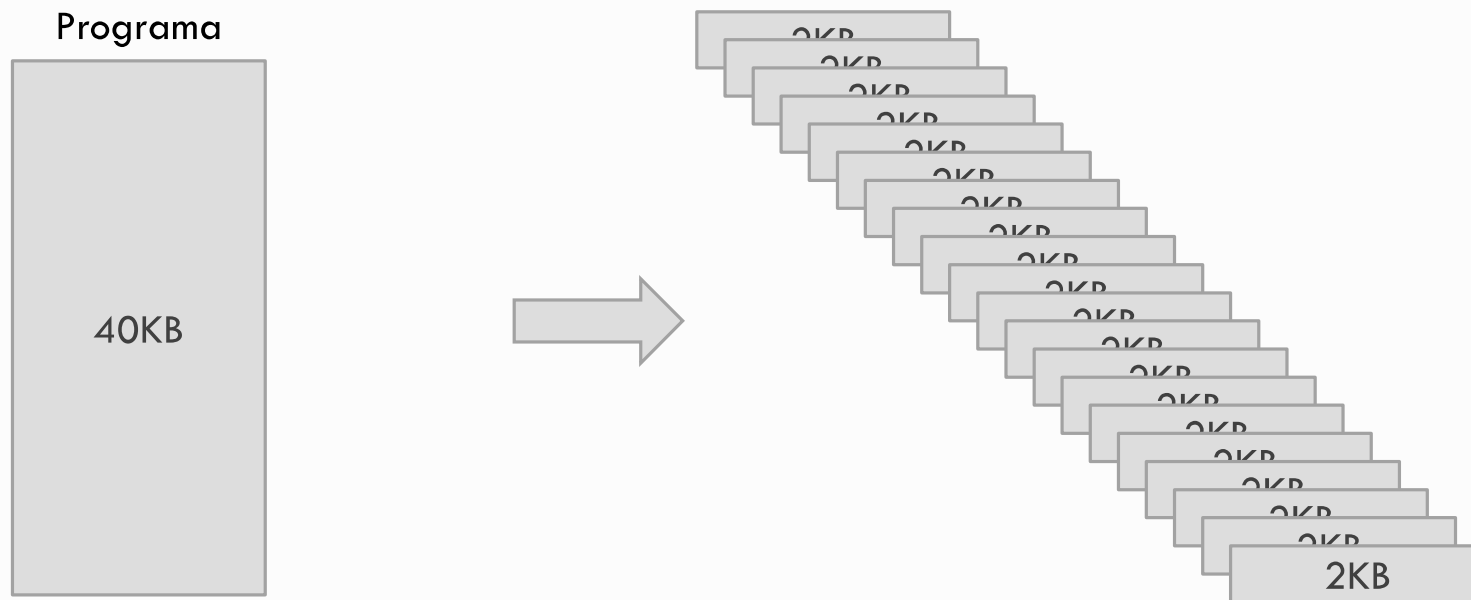
Desde os primórdios da computação que é comum a existências de programas maiores que o tamanho de memória disponível.

Como este problema pode ser resolvido?



Memória Virtual

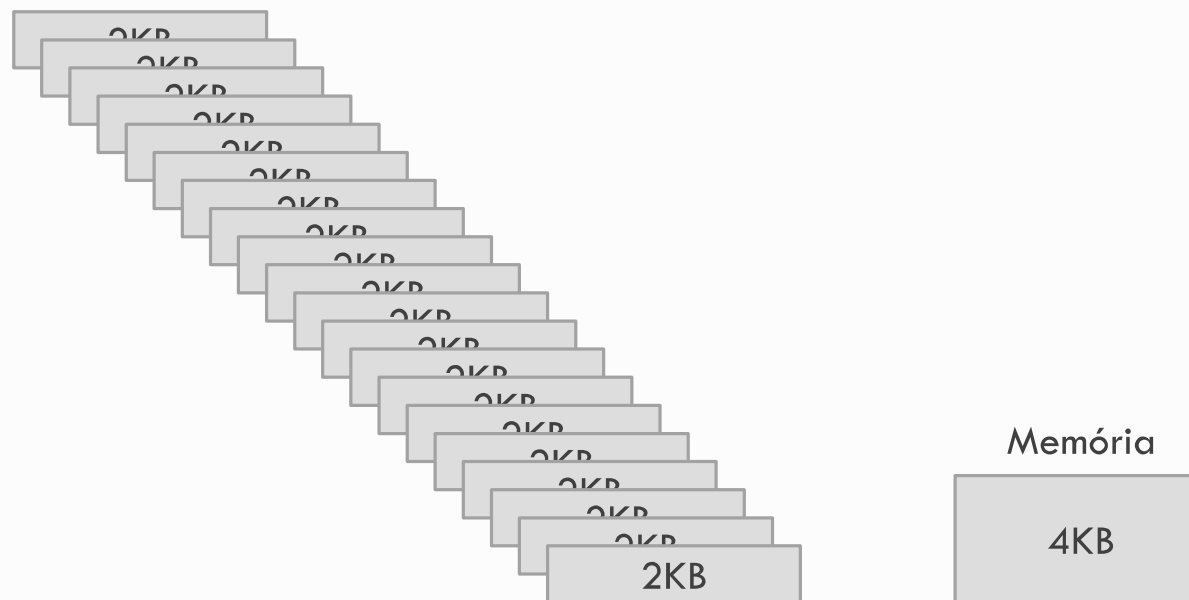
Uma solução utilizada com muita frequência no passado era a divisão do programa em subprogramas, denominados overlays (módulos de sobreposição).



Exemplo com 20 módulos de 2KB cada

Memória Virtual

O módulo 0 é sempre o primeiro a ser executado e o programador deve garantir a correta interação entre os módulos.



Programa dividido em 20 módulos

Memória Virtual

O S.O. era responsável apenas pela troca de módulos na memória, seguindo as instruções encontradas ao executar cada módulo.

- Definidas pelo programador.
- Responsabilidade do programador.

A divisão do programa em módulos era um trabalho lento e totalmente dependente da arquitetura.

- Se o programa fosse executado em uma máquina com menos memória se comparado a máquina que o mesmo foi projetado, ele deveria ser totalmente reorganizado.

Memória Virtual

Não demorou muito para a comunidade científica pensar em atribuir tal responsabilidade para o Sistema Operacional.

Em 1961 aparece pela primeira vez o conceito de **memória virtual**.

Memória Virtual

Ideia básica:

- Um programa maior que a memória primária pode ser executado por meio de uma escolha inteligente sobre qual parte vai estar na memória em cada instante.
- Partes do programa podem ser dinamicamente carregadas para a memória primária.

Paginação

Sistemas operacionais modernos utilizam o conceito de paginação para implementar a memória virtual.

Quando um programa executa a instrução

- **move REG, 500**

Ele deseja copiar o conteúdo do endereço de memória 500 para o registrador REG.

Paginação

move REG, 500

Na verdade, 500 é um endereço virtual. Não necessariamente 500 do endereço virtual corresponde ao endereço físico 500.

Também é possível que o endereço virtual requisitado não esteja na memória principal.

Paginação

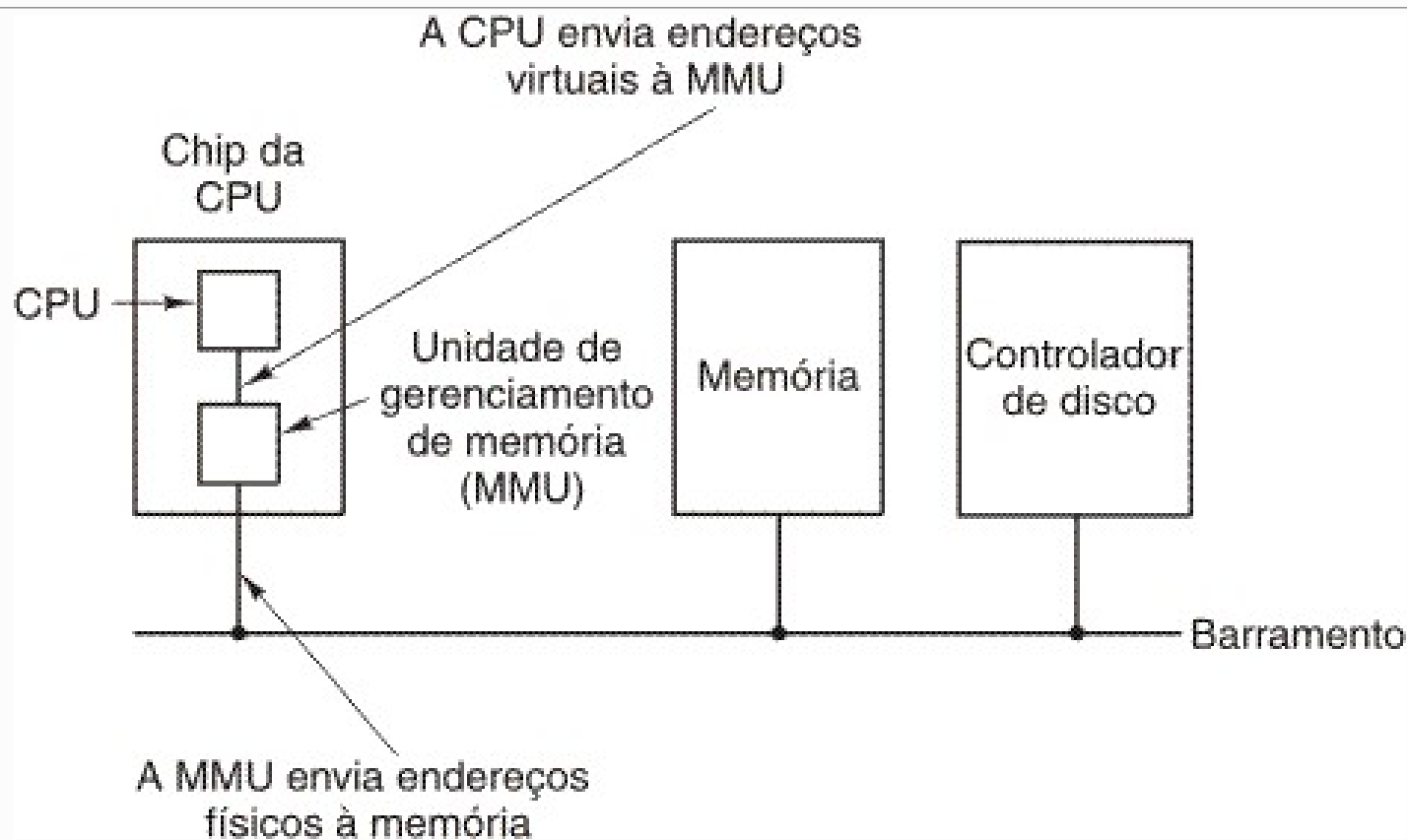
Em computadores sem memória virtual, o endereço virtual é idêntico ao endereço físico.

Em computadores com memória virtual é necessário um sistema especial para efetuar as conversões de endereços necessárias.

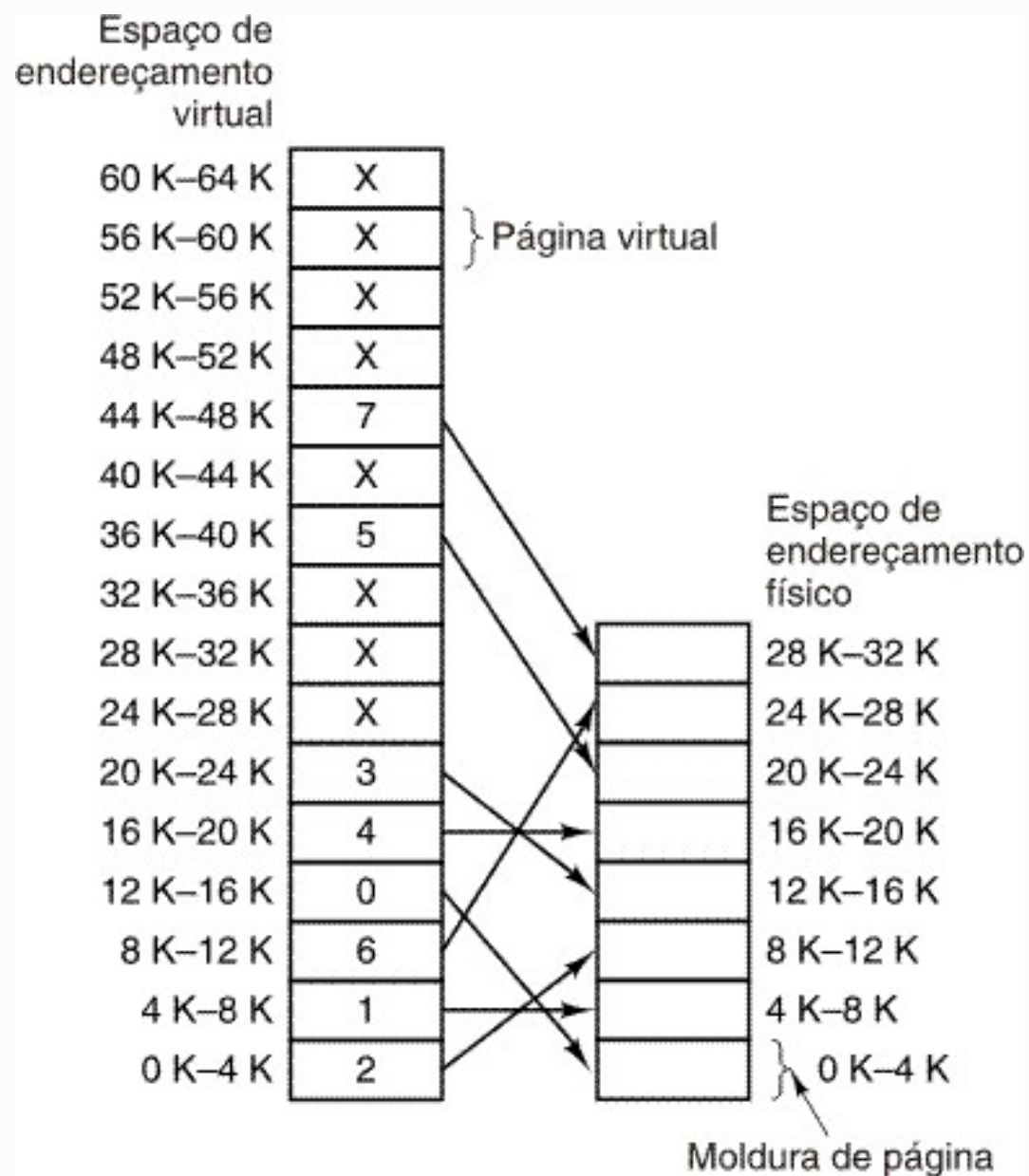
Este módulo é denominado MMU (*memory management unit*) – unidade de gerenciamento de memória.

Paginação – MMU

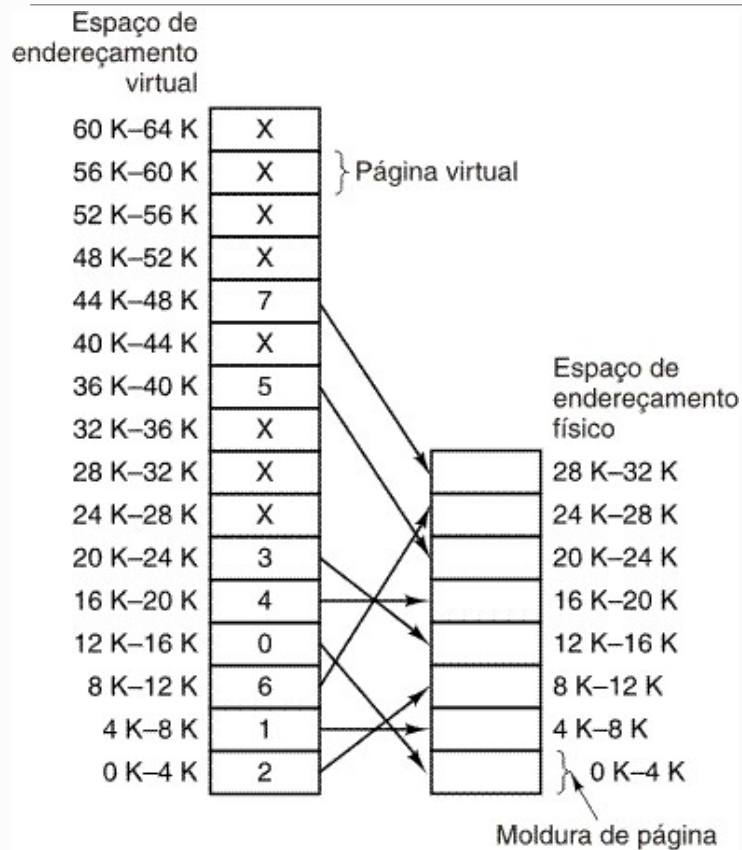
Unidade de Gerenciamento de Memória



Estrutura utilizada pela MMU



Conversão utilizada pela MMU



A instrução

move REG, 0

É convertida pela MMU para

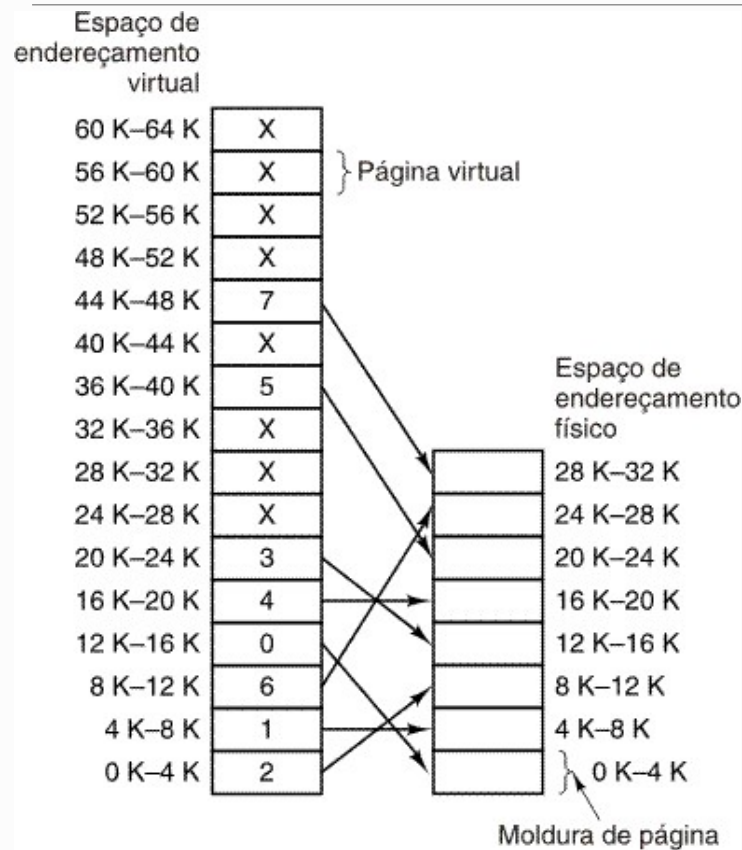
move REG, 8192

O endereço 0 (virtual) pertence a página 2 da memória real.

Tamanho da pagina = 4k;

endereço = $(1024 * 2 * 4)$

Conversão utilizada pela MMU



Já a instrução

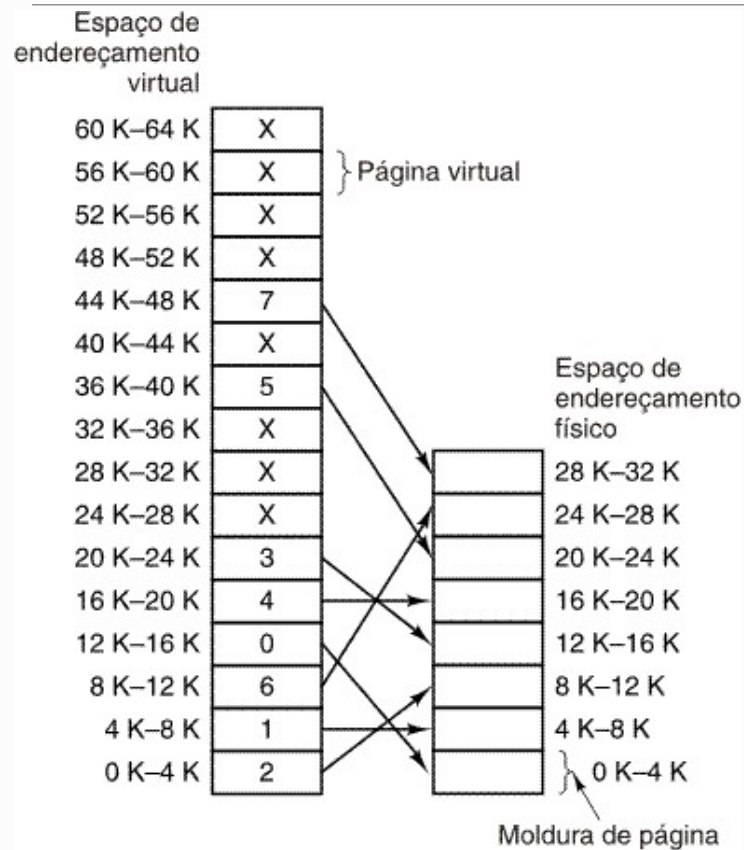
move REG, 8192

É convertida pela MMU para

move REG, 24576

O endereço 8192 pertence a página 6 da memória real (24K até 28K)

Conversão utilizada pela MMU



Limitação

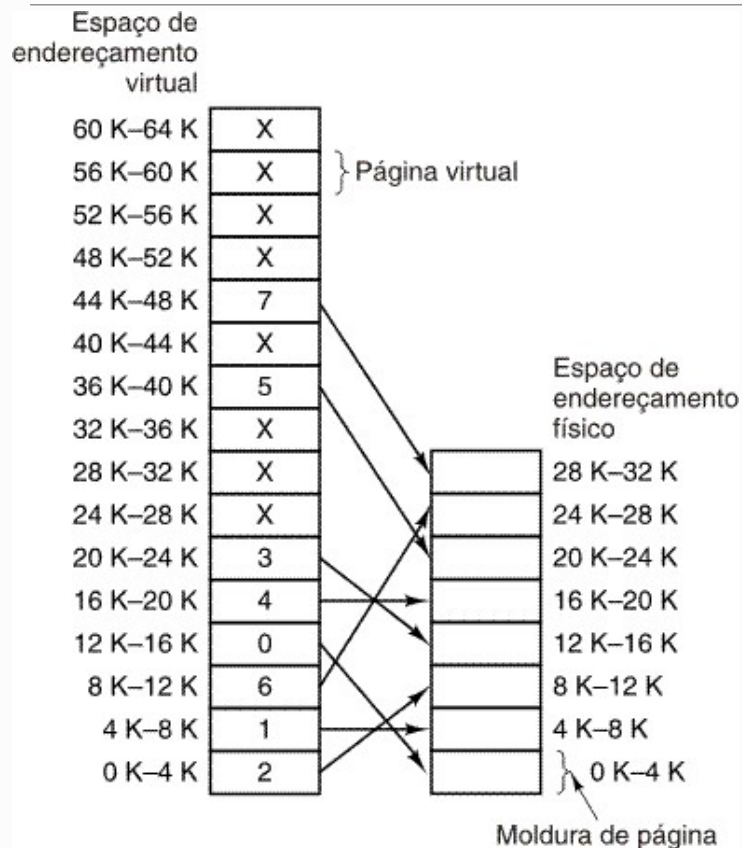
A memória virtual é maior do que a memória física primária.

O que acontece ao invocarmos

move REG, 32780 ?

12º byte da 8ª página da memória virtual

Conversão utilizada pela MMU



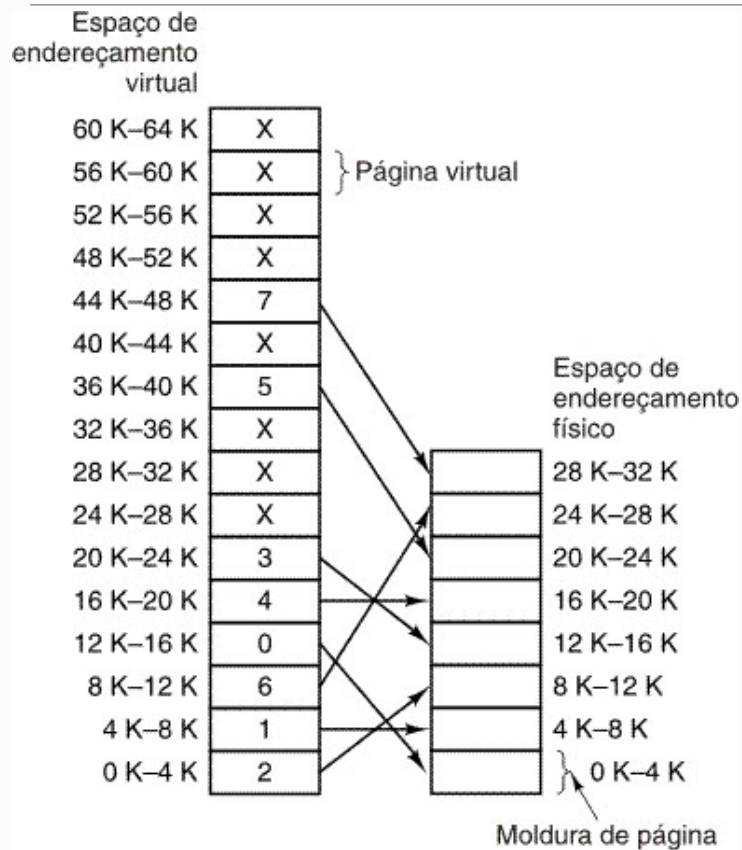
move REG, 32780 ?

A MMU encontra um indicador “X”. Isso significa que o conteúdo não está na memória principal. Está no disco.

Isso é conhecido como page fault (falta de página);

O que a MMU deve fazer?

Conversão utilizada pela MMU



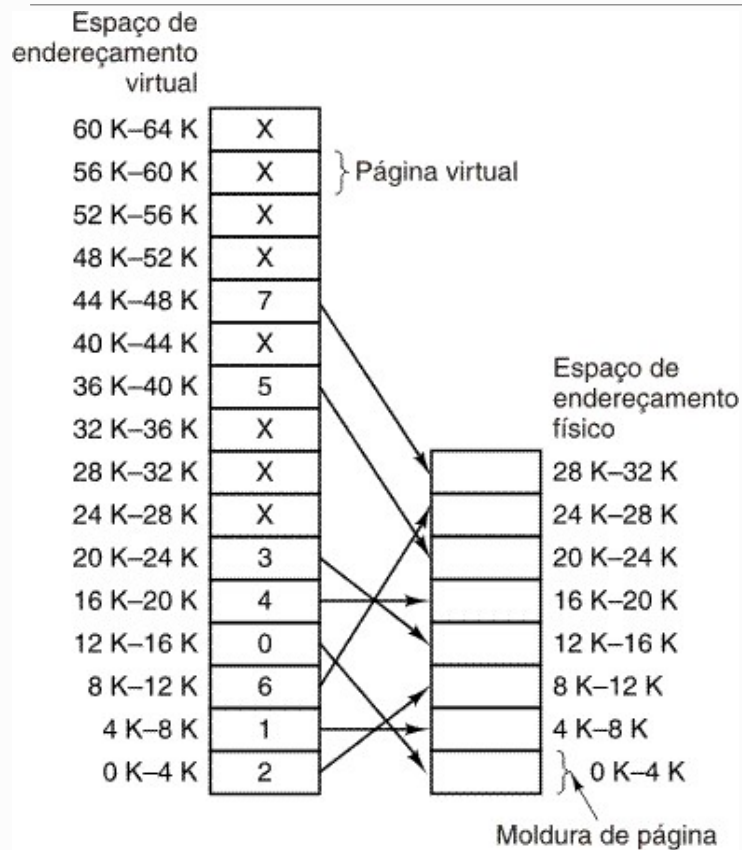
Page fault:

A MMU deve trazer a página do disco para a memória primária.

Mas todas as páginas da memória primária estão sendo referenciadas pela memória virtual.

O que fazer?

Conversão utilizada pela MMU

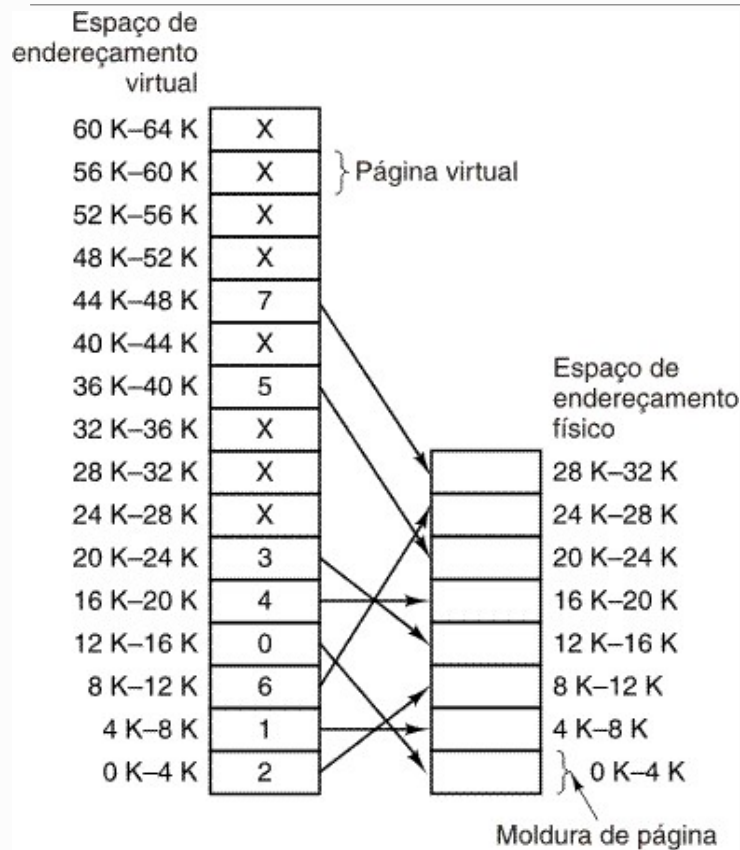


Page fault:

Neste caso, a MMU deve executar um algoritmo para substituição de página.

Elimina uma das páginas da memória física (mas a armazena no disco) e traz a página requisitada para a posição liberada.

Conversão utilizada pela MMU



Page fault:

Problema: Qual página vai ser eliminada da memória?

Existem diferentes algoritmos

Não usada recentemente.

FIFO.

Menos recentemente utilizada (MRU).

Dentro outros.

Substituição de Página

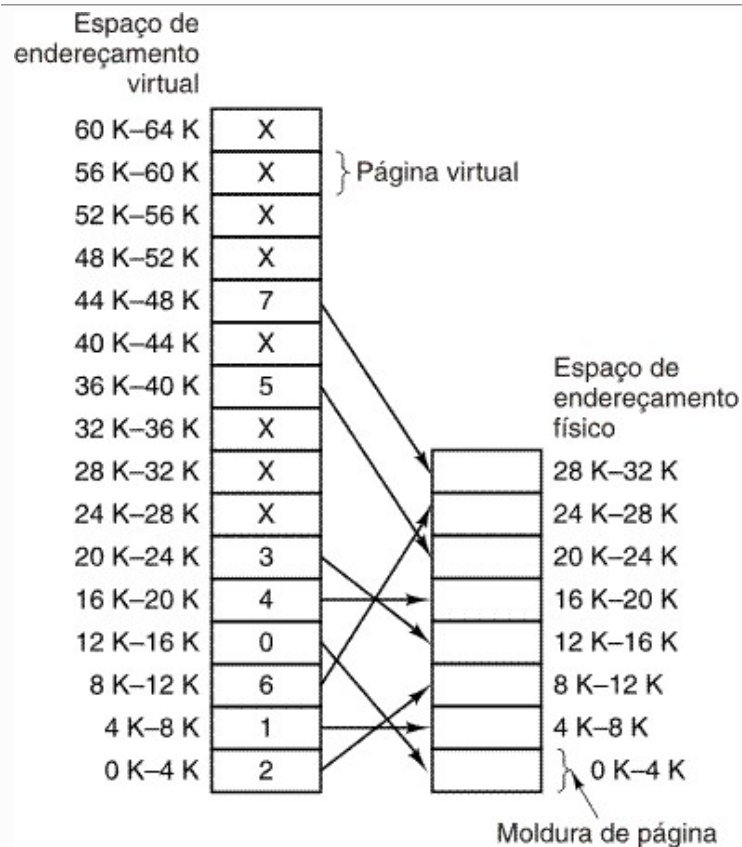
Decisão do Sistema Operacional:

- Qual página deve ser removida?

É importante notar que este problema se repete em diferentes áreas da Computação:

- Cache de servidores web.
 - Quais páginas devem estar no cache do servidor Apache?
- Cache do proxy;
- Cache do Youtube.

Substituição de Página



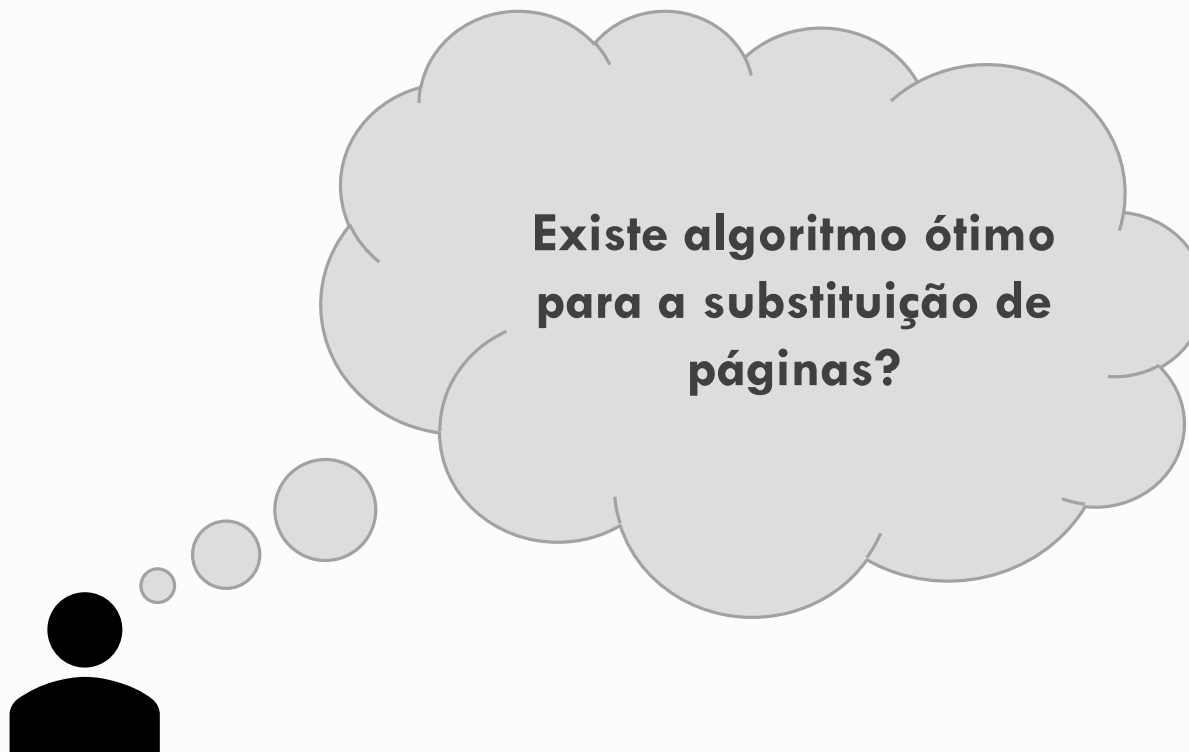
Diferentes algoritmos podem ser implementados para manutenção de caches.

Substituição Aleatória:

- Escolha aleatória de uma moldura de página na memória principal para ser substituída.
- Fácil implementação.
- Desempenho frustrante.

Substituição de Página

Antes de comentar outros algoritmos devemos responder:



Substituição de Página

Existe.

Mas é impossível de implementar em um S.O. de uso geral!

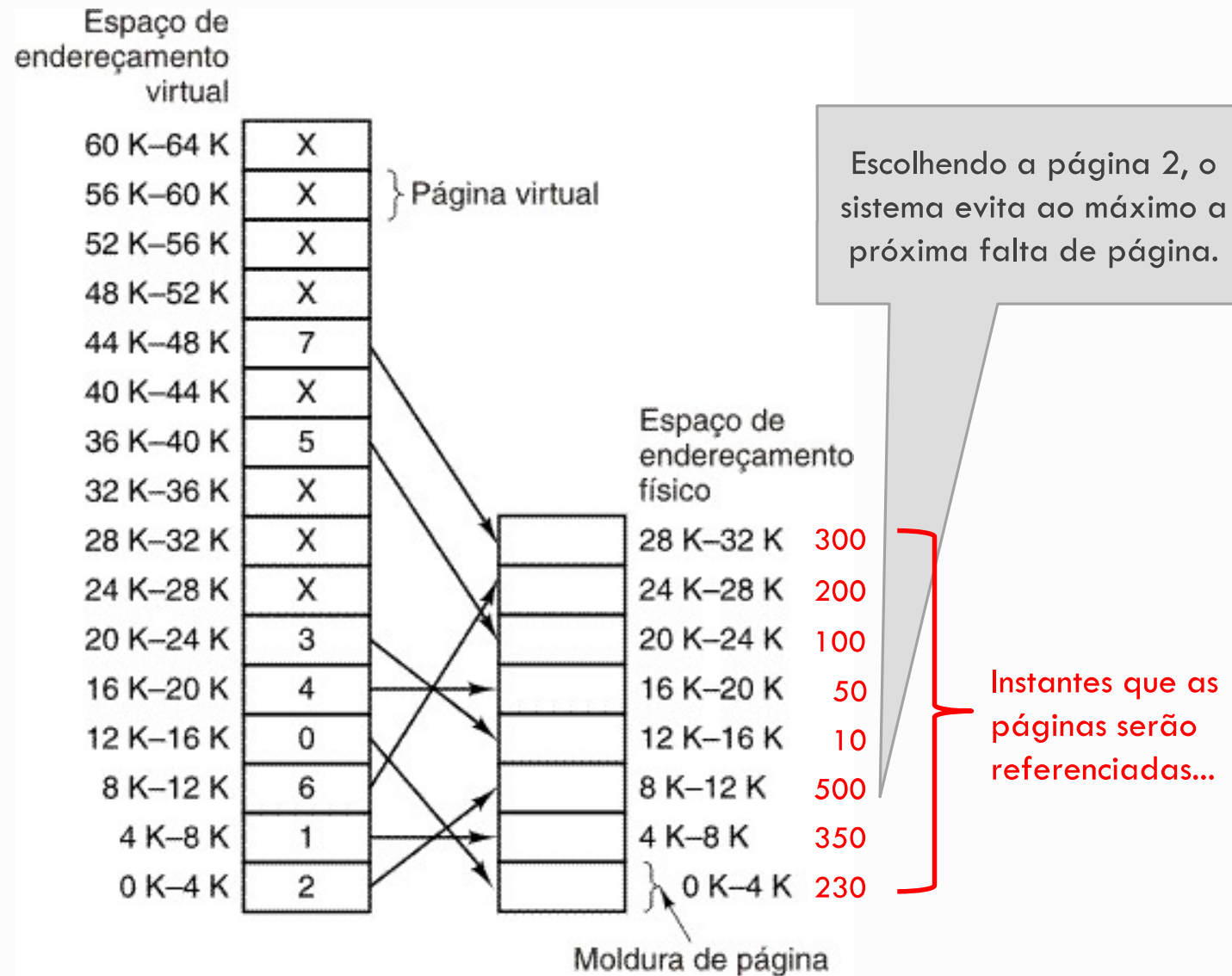
Então, é como se não existisse.

Algoritmo:

Remover a página que mais tarde será referenciada.

Ou seja, evita ao máximo futuras faltas de página

Substituição de Página



Algoritmos de Substituição de Página

Não Usada Recentemente (NUR)

Não Usada Recentemente (NUR)

A tabela de páginas precisa de dois bits para cada registro de página:

R – indica se a página foi referenciada.

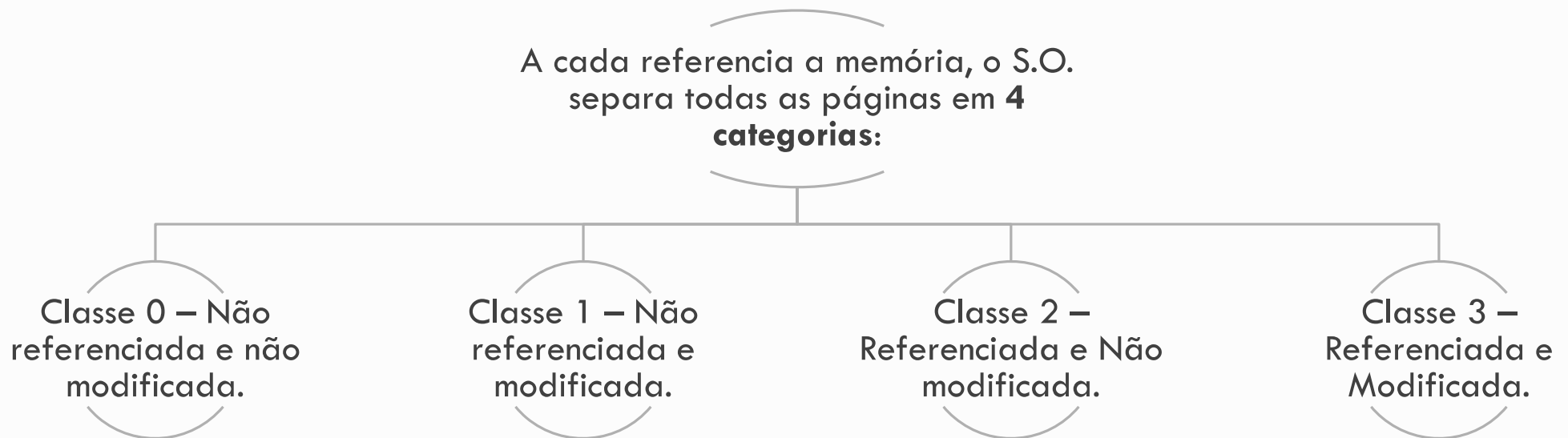
M – indica se a página foi modificada.

O número 1 (um) indica verdadeiro.

O número 0 (zero) indica falso.

Não Usada Recentemente (NUR)

É importante observar que **os bits devem ser atualizados a cada referencia a memória.**



Não Usada Recentemente (NUR)

O Algoritmo NUR remove aleatoriamente uma página da classe de mais baixa ordem que não esteja vazia.

As classes embutem informações sobre as páginas mais “importantes” e mais utilizadas.

E faz uma escolha “inteligente” sobre a eliminação de uma página.

Algoritmos de Substituição de Página

First In, First Out (FIFO)

First In, First Out (FIFO)

A primeira página da fila é a próxima a deixar a memória principal na próxima falta de página.

Fácil Implementação e baixo custo computacional.

- Operações em $\Theta(1)$ para remover e inserir na lista.

Não tem garantia alguma de qualidade e sua implementação pura é raramente encontrada na implementação de caches.

Algoritmos de Substituição de Página

Segunda Chance

Segunda Chance

É uma simples modificação do algoritmo FIFO.

Utiliza a Fila, mas evita eliminar uma página recentemente utilizada.

Para analisar esta informação, é utilizado o bit R (referenciada).

Segunda Chance

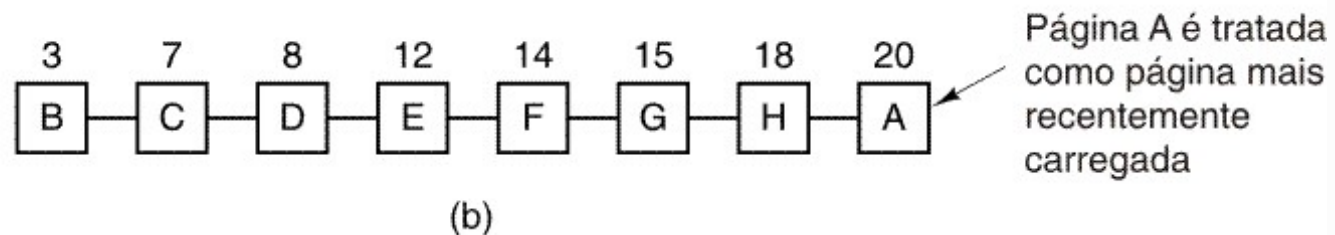
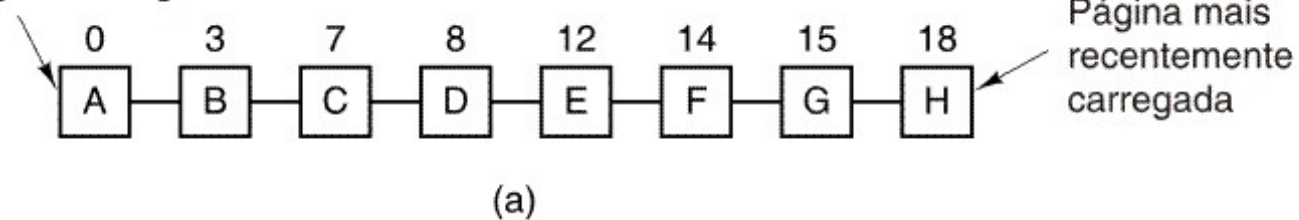
Se a página foi lida ou referenciada recentemente, ela não é removida da lista.

Vai para o fim da lista, e seu bit R (recentemente referenciada) passa a ser 0.

Mescla a ideia do FIFO, mas inserindo informações de popularidade.

Segunda Chance

Primeira página carregada



Operação do algoritmo segunda chance

a) Lista de páginas em ordem FIFO;

b) Estado da lista em situação de falta de página no instante 20, com o bit R da página A em 1 (números representam instantes de carregamento das páginas na memória).

Algoritmos de Substituição de Página

Relógio

Relógio

Melhoria do Algoritmo de Segunda Chance.

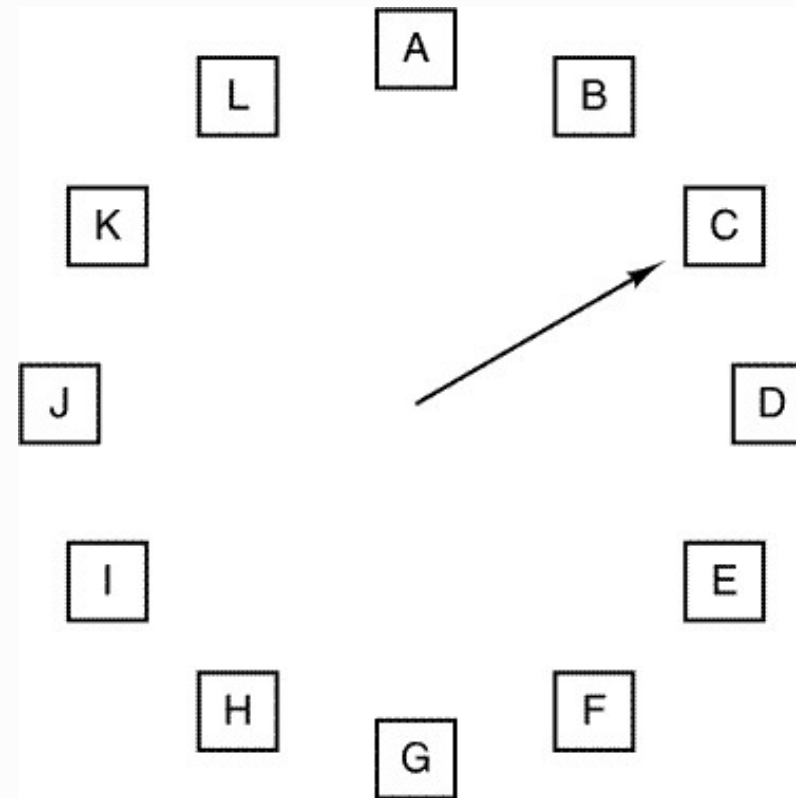
Ao invés de implementar uma fila, implementa uma lista circular:

Evita remover elementos do início da fila, e inserir no fim da fila
(evita alocação de memória dinâmica).

Apenas move seu apontador do primeiro da “fila”.

Só é eficiente porque a memória virtual tem um tamanho fixo.

Relógio



Quando ocorre uma falta de página, a página indicada pelo ponteiro é inspecionada. A ação executada depende do bit R:

R = 0: Remover a página

R = 1: Zerar R e avançar o ponteiro

Relógio

Diferencia do algoritmo da Segunda chance somente em implementação.

Não existe nenhuma diferença com relação a quantidade de páginas substituídas ao final de sua execução.

Algoritmos de Substituição de Página

Menos Recentemente Usada (MRU)

Menos Recentemente Usada (MRU)

Utiliza da hipótese:

- Páginas referenciadas intensamente nas ultimas instruções provavelmente serão de novo referenciadas de maneira intensa nas próximas instruções.

Obviamente, remove a página a mais tempo não referenciada.

Observe que isso não acontece no algoritmo de segunda chance.

Menos Recentemente Usada (MRU)

Para implementar, é necessário manter uma lista encadeada de todas as páginas na memória;

Vantagem?

Na ocorrência de uma falta de página, a página a mais tempo sem utilização é a primeira da lista.

Desvantagem?

A lista tem que ser gerenciada a cada referência à memória.