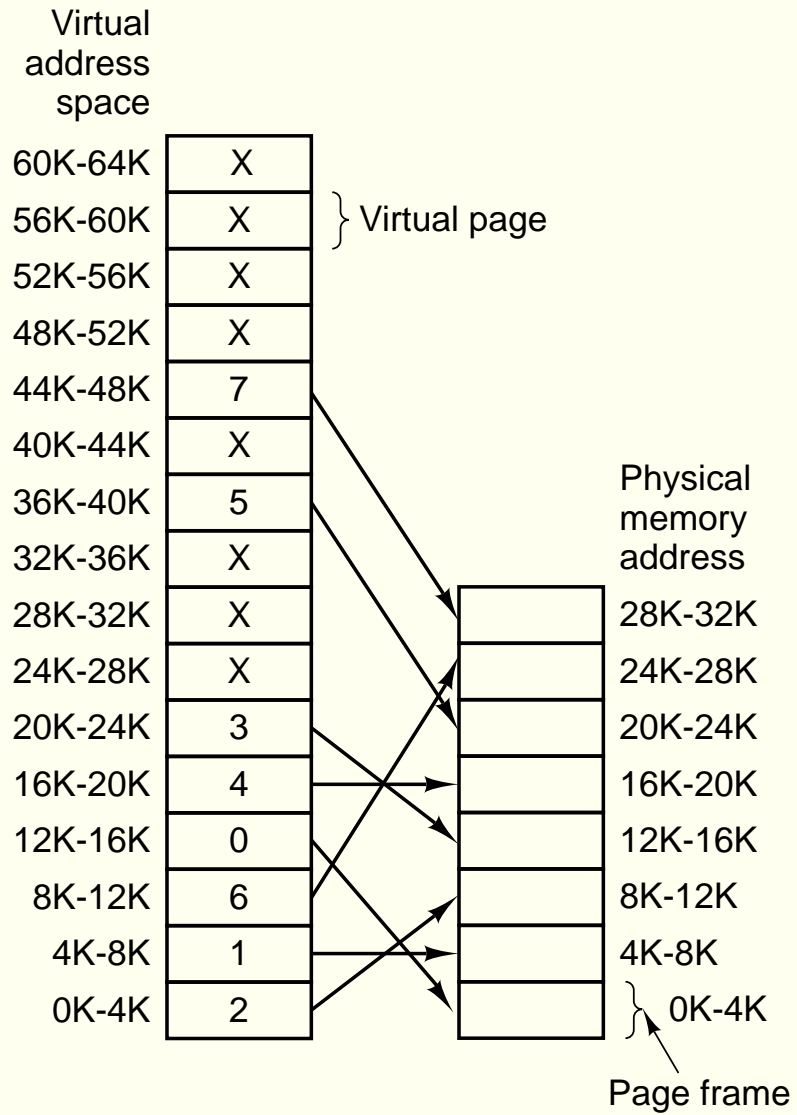


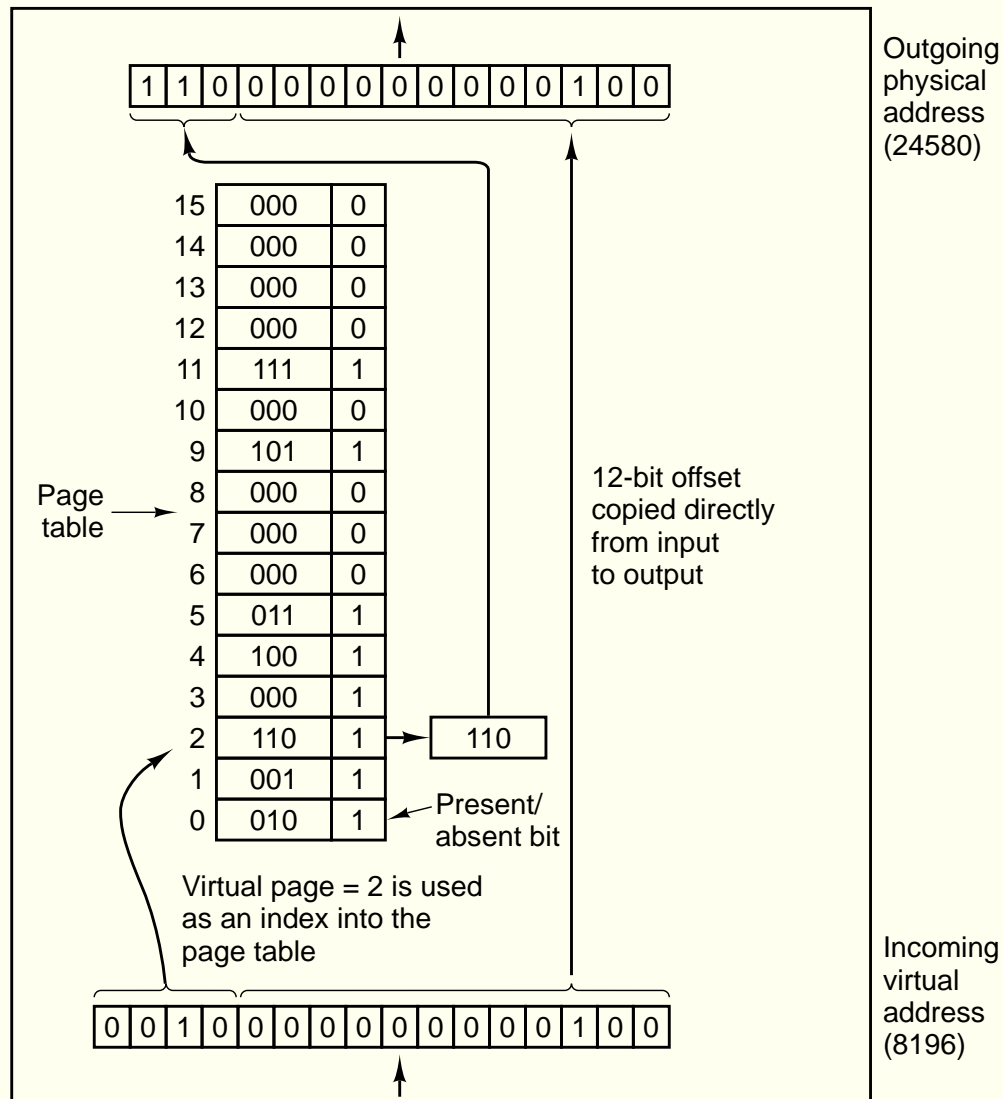
**MC514–Sistemas Operacionais: Teoria e Prática**  
1s2009

## **Gerenciamento de Memória - 1**

# Paginação

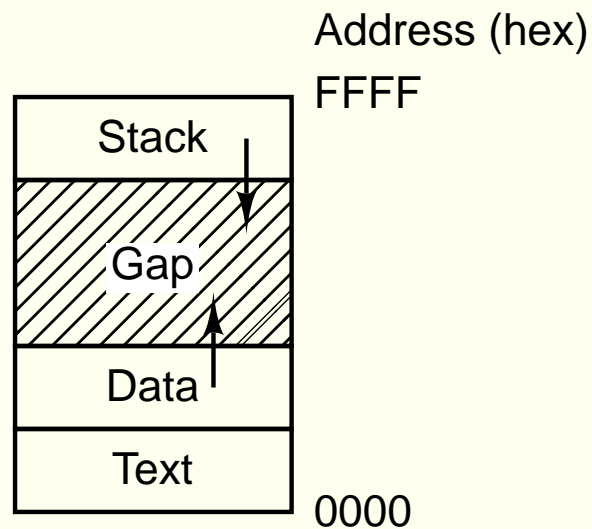


# Mapeamento dos endereços



# Espaço de endereçamento

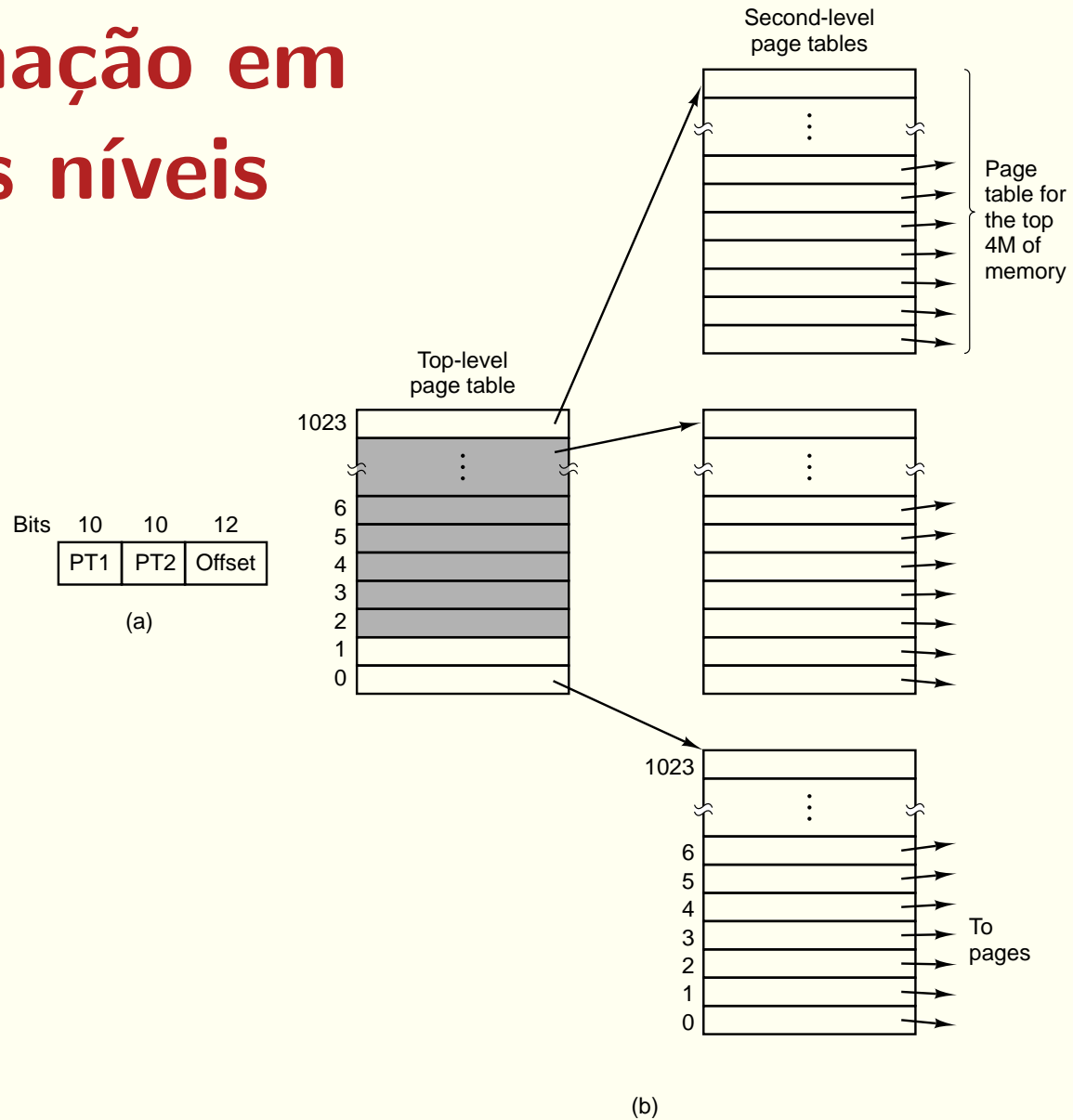
- Nem todas as páginas precisam ser mapeadas
- Veja o código `sbrk.c`



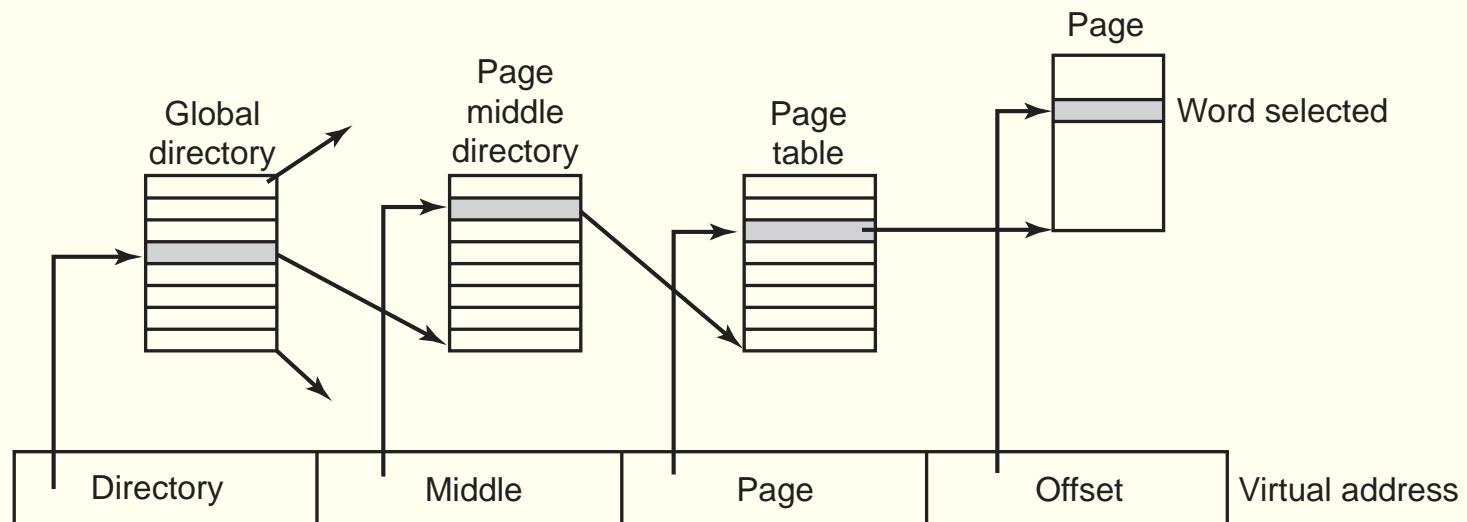
# Paginação - Exemplo

- 32 bits de endereço
- páginas de 4k
- $2^{20}$  entradas na tabela?  
e apenas algumas seriam utilizadas...
- Outras opções
  - paginação em vários níveis
  - tabelas invertidas

# Paginação em vários níveis

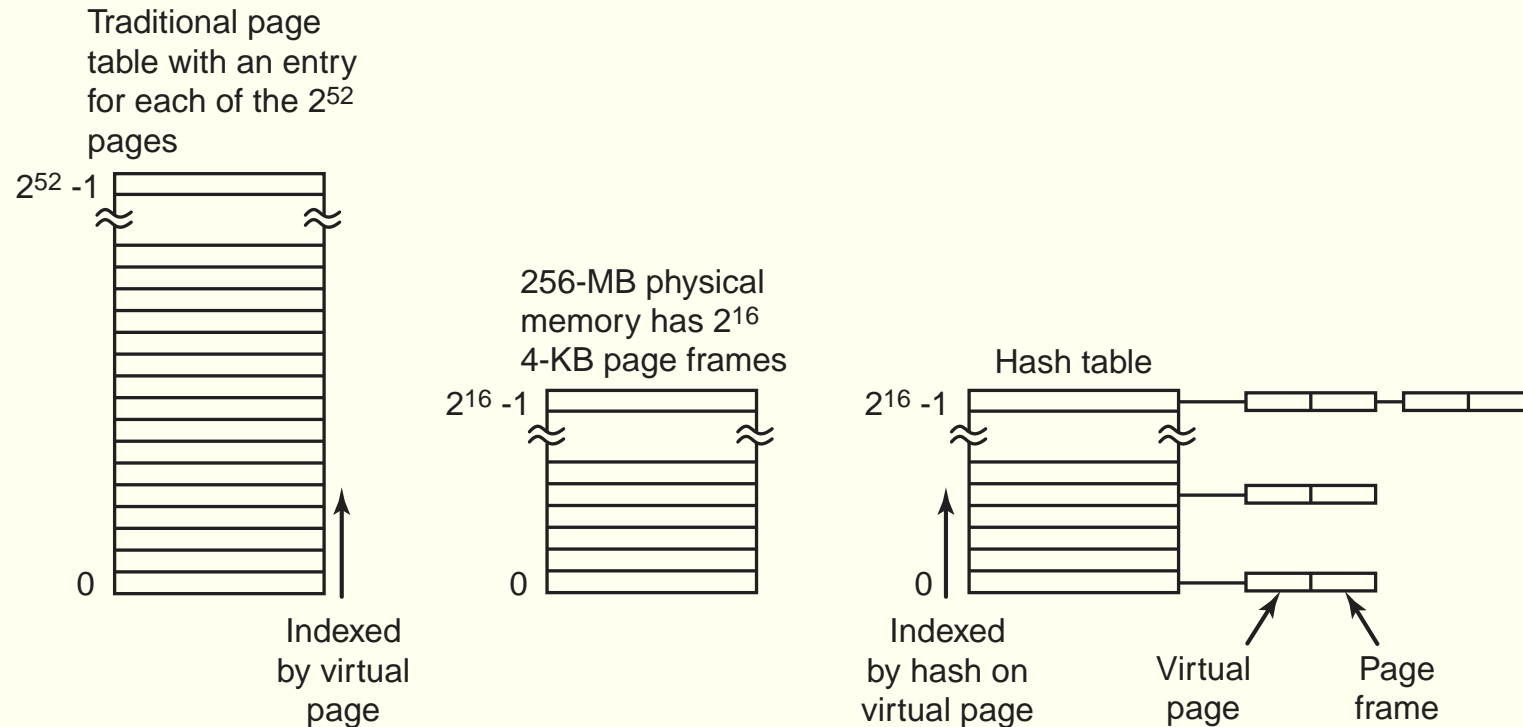


# Tabela em três níveis



- Linux?

# Tabela de páginas invertida



- Uma entrada por página na memória física



# Translation Look Aside Buffers (TLBs)

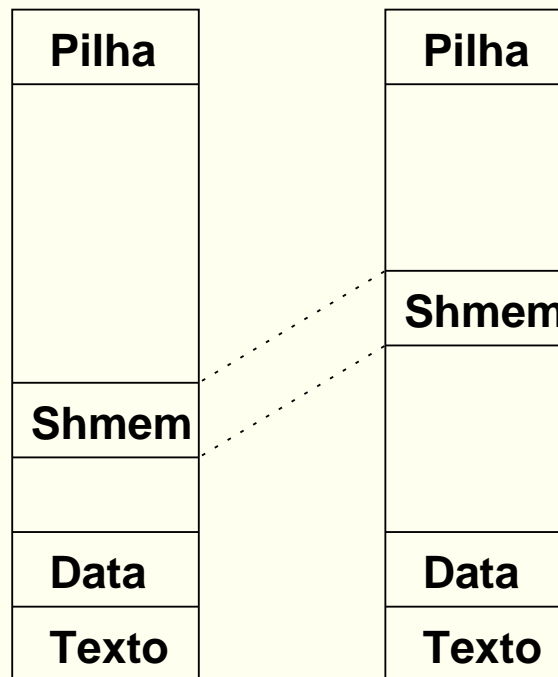
Valid	Virtual page	Modified	Protection	Page frame
1	140	1	RW	31
1	20	0	R X	38
1	130	1	RW	29
1	129	1	RW	62
1	19	0	R X	50
1	21	0	R X	45
1	860	1	RW	14
1	861	1	RW	75

Vamos analisar os exemplos pag1.c e loop\_malloc.c

# Memória compartilhada

Processo A

Processo B



# Memória compartilhada

```
int shmget(key_t key, size_t size, int shmflg);  
void *shmat(int shmid,  
            const void *shmaddr, int shmflg);
```

- Veja os exemplos: sh1.c sh2.c sh\_fork.c sh\_server.c e sh\_client.c

# Substituição de páginas

Algoritmo ótimo:

- Baseado no uso futuro de uma página
- Impossível de ser implementado
- Pode ser simulado (segunda execução do mesmo processo com a mesma entrada)
- Útil para medidas de desempenho

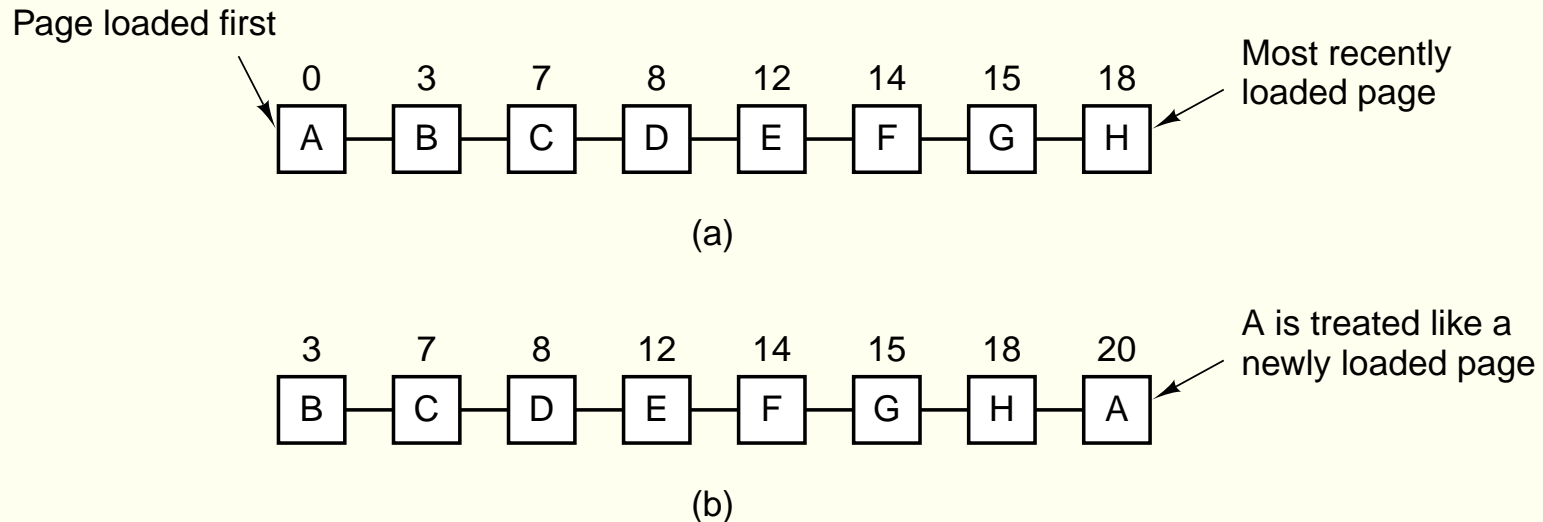
## Não usada recentemente

- Classe 0: não referenciada, não modificada
- Classe 1: não referenciada, mas modificada
- Classe 2: referenciada, mas não modificada
- Classe 3: referenciada e modificada

## First In, First Out

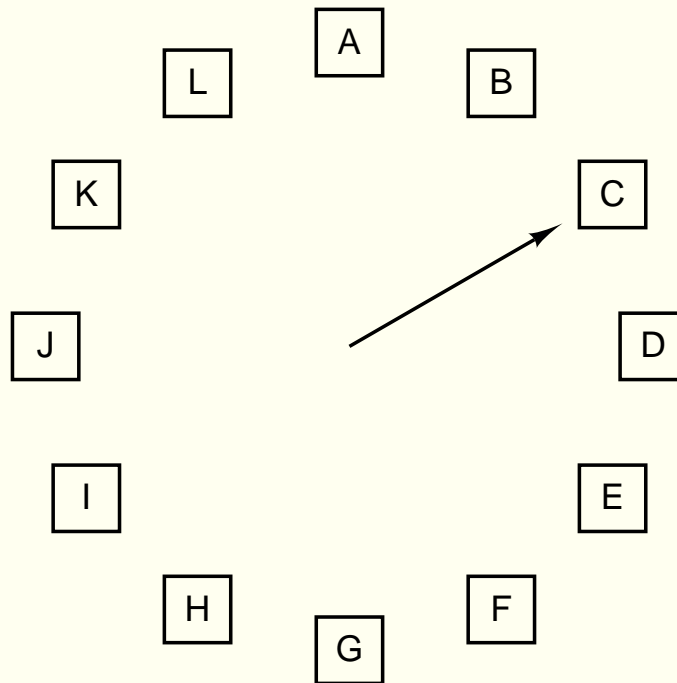
- Simplementes coloca as páginas em uma fila
- Pode remover páginas importantes

## Segunda chance



- Se o bit  $R == 0$ , a página é substituída, senão
- bit  $R$  é limpo e a página é colocada no final da fila

# Relógio



When a page fault occurs,  
the page the hand is  
pointing to is inspected.  
The action taken depends  
on the R bit:

R = 0: Evict the page

R = 1: Clear R and advance hand

- Implementação circular da segunda chance



## Política de limpeza

- Gravar as páginas modificadas na última hora pode ser pouco eficiente
- *Paging Daemon*
  - Varre periodicamente a memória
  - Tenta manter um número de *frames* livres
  - kswapd

# Controle da carga

- *Thrashing*
  - O Sistema Operacional só se ocupa das tarefas de paginação e escalonamento
  - Todos os processos precisam de mais memória
  - Swap (como escolher quais processos vão para o disco?)