

Aula 20 - 26.10.2018

Sumário:

Substituição de páginas(continuação).

Bib: OSTEP Caps. 22 e 23

Algoritmos de substituição de páginas

- *Ideal*: remover a página que vai demorar mais tempo a ser usada
- *Random*: escolhe página ao acaso
- *FIFO*: remover a página que está em RAM há mais tempo.
- *LRU* – least recently used.

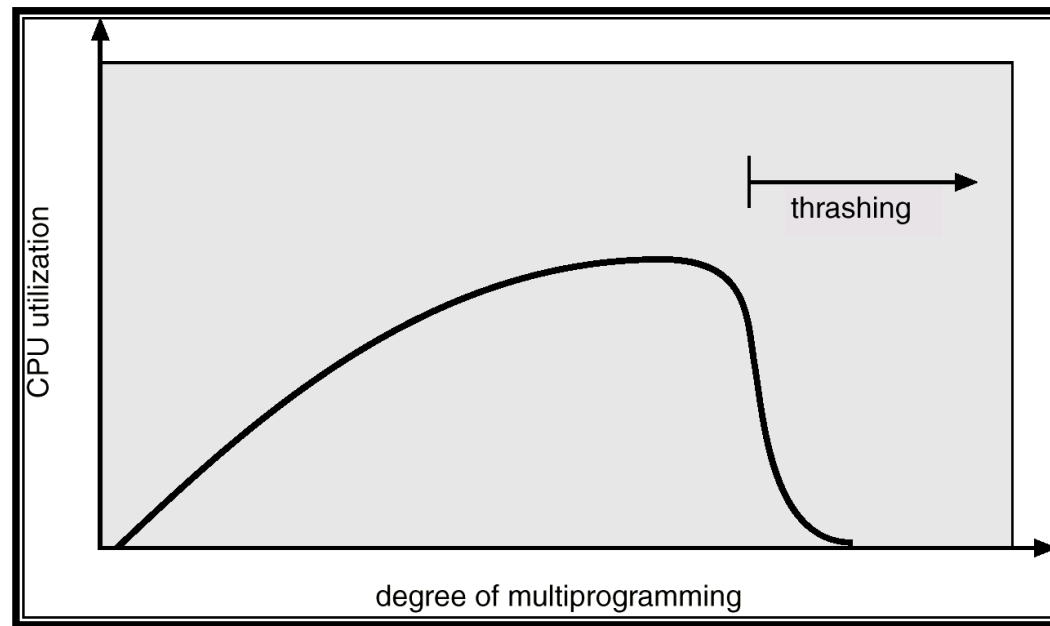
O nº de faltas de página está relacionado com o nº de páginas atribuídas

- Se um processo tem todas as suas páginas em RAM nunca incorre em falta de páginas
- À medida que vai tendo menos páginas, o desempenho degrada-se, porque o processo está sempre no estado WAITING/BLOCKED à espera que chegue uma página.

Thrashing

- Se um processo não tem páginas “suficientes”, o ritmo de faltas de página aumenta muito. Isto é caracterizado por (comando *vmstat*):
 - Baixa taxa de ocupação do CPU.
 - Grande número de operações de I/O sobre o disco de paginação.
 - Poucas “frames” livres.
- **Thrashing** $\equiv$ o processo está praticamente sempre à espera que o SO carregue páginas (page in) e a transferir páginas (page out) de/para o disco.

Thrashing

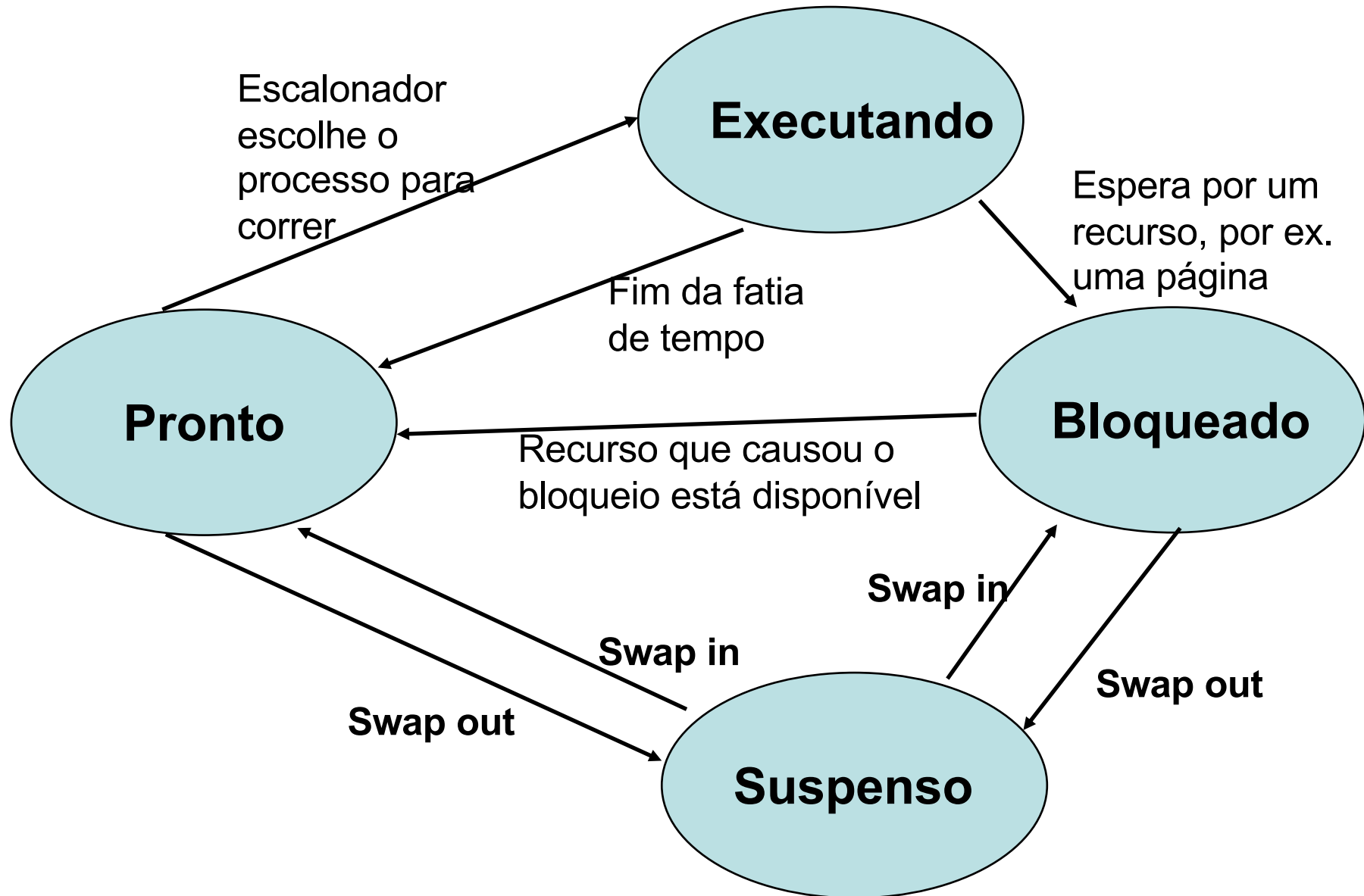


- À medida que o número de processo aumenta, o número de páginas para cada um diminui
- Quando todos os processos têm poucas páginas, há muitos processos em READY/RUNNING e muitos em BLOCKED
- A solução é ter mais RAM

Swapping

- Em situações críticas de escassez de páginas, o SO pode decidir suspender um processo, retirando-lhe todas as páginas em RAM (*swap out*)
- As páginas libertadas são usadas por outros processos que assim acabam mais rapidamente
- Desaparecida a situação de escassez de páginas, o processo suspenso volta a ter páginas em RAM (*swap in*)

Swapping



Reservatório (pool) de páginas livres

- O SO mantém um conjunto de “frames” livres:
 - Quando o algoritmo de substituição escolhe uma frame F que está suja, vai-se buscar uma “frame” ao reservatório que é dada ao processo; isto permite a continuação mais rápida do processo que incorreu na falta
 - O conteúdo da frame F é escrita no disco; quando a operação se completa, a frame F vai para o “pool”
- O SO vai em “background” escrevendo páginas sujas no disco – depois de escritas deixam de estar “dirty”

Reservatório de páginas livres (cont)

- Para todas as “frames” no pool, o conteúdo é preservado e mantém-se informação sobre o par (processo, página virtual) correspondente. Isto permite o reaproveitamento se uma página substituída volta a ser precisa
- Para ser reutilizada por outro processo a “frame” tem de ser preenchida com zeros, por razões de segurança

Reservatório de páginas livres (cont)

- 1: frames retiradas a processos pelo algoritmo de substituição
- 2: frames obtidas na cache, sem necessidade de E/S
- 3: frames “postas a zero” obtidas por um processo
- 4: páginas escritas no disco por um thread de sistema
- 5: páginas preenchidas com 0s por um thread de sistema

