
EP3

MAC 422

Sistemas Operacionais

Jonas Arilho
Pedro Bortolli

20/11/2017

Estrutura do Código

Shell

- Implementação do Shell do EP3 parecida com a do EP1
 - Comandos readline e history
-

Gerenciamento de Espaço Livre

Alocação

- Implementação de três algoritmos
 - Best Fit
 - Worst Fit
 - Quick Fit
-

Alocação

- Bitmap implementado com vector (C++)
- Best/Worst fit fazem uma varredura em toda a memória em $O(n)$
- Quick fit armazena os 3 tamanhos mais frequentes de processos e guarda as posições da memórias em três sets (C++)

Substituição de Páginas

Páginas

-
- Implementação de 4 algoritmos
 - Ótimo
 - Fila (FIFO)
 - Menos Recentemente Usado versão 2 (LRU 2)
 - Menos Recentemente Usado versão 4 (LRU 4)
-

Páginas

- Tabela de Páginas como um vector (c++) de páginas virtuais
- Tabela de Quadros de Página como um vector (c++) de frames
- Mapeamento de Páginas de um Processo como um vector (c++) com o id das páginas de cada processo

Páginas

- Page_Fault separada em 2 tipos:
 - Nova Página
 - Página já existente
- Criação de Páginas implica adicionar uma nova entrada em todas as tabelas
- Carregar Página atualiza as tabelas
- Suspende Página atualiza as tabelas

Detalhes

- Simulação ocorre em ordem cronológica: primeiro chegada de processos, depois acessos a memória, e finalmente fim de um processo. Não foi simulado o tempo real de fato (como feito no EP2)
 - Arquivos binários gerados usando fseek e fwrite
-

Arquivos Temporários

/tmp

- Os arquivos são criados dentro do diretório /tmp do linux e são abertos como binários e removidos depois
- Durante a alocação de memória, é modificado o arquivo /tmp/ep3.vir
- Quando uma página é substituída, é modificado o arquivo /tmp/ep3.mem

Resultados (gráficos)

Termologia dos Resultados

Obtendo os resultados

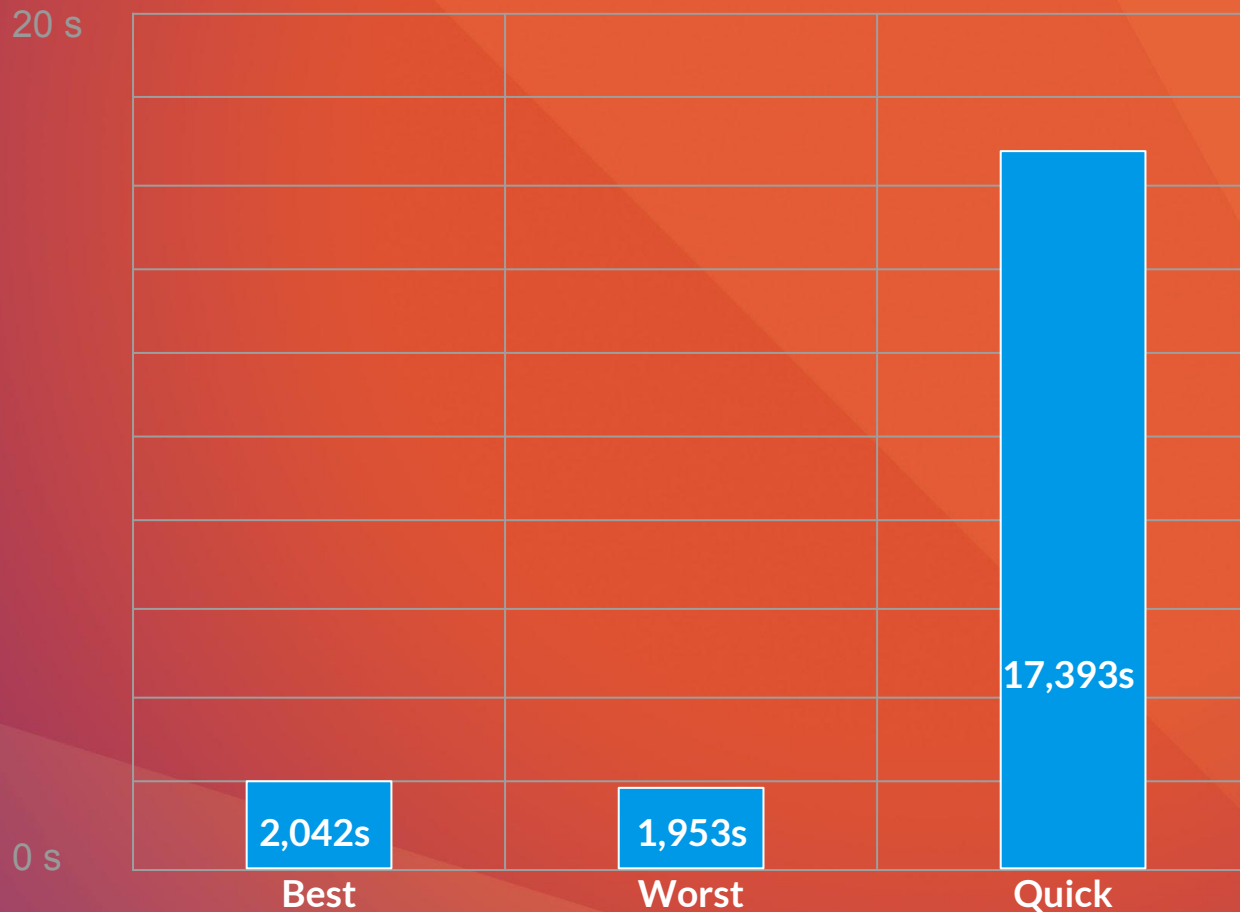
- Tempo de execução medido em segundos com o comando time
 - Contador no código para contar os page faults
-

Configuração do Arquivo Trace

- memória virtual: 1024 bytes
 - memória física: 256 bytes
 - unidade de alocação: 2 bytes
 - tamanho da página: 8 bytes
 - Número de processos: 10.000
-

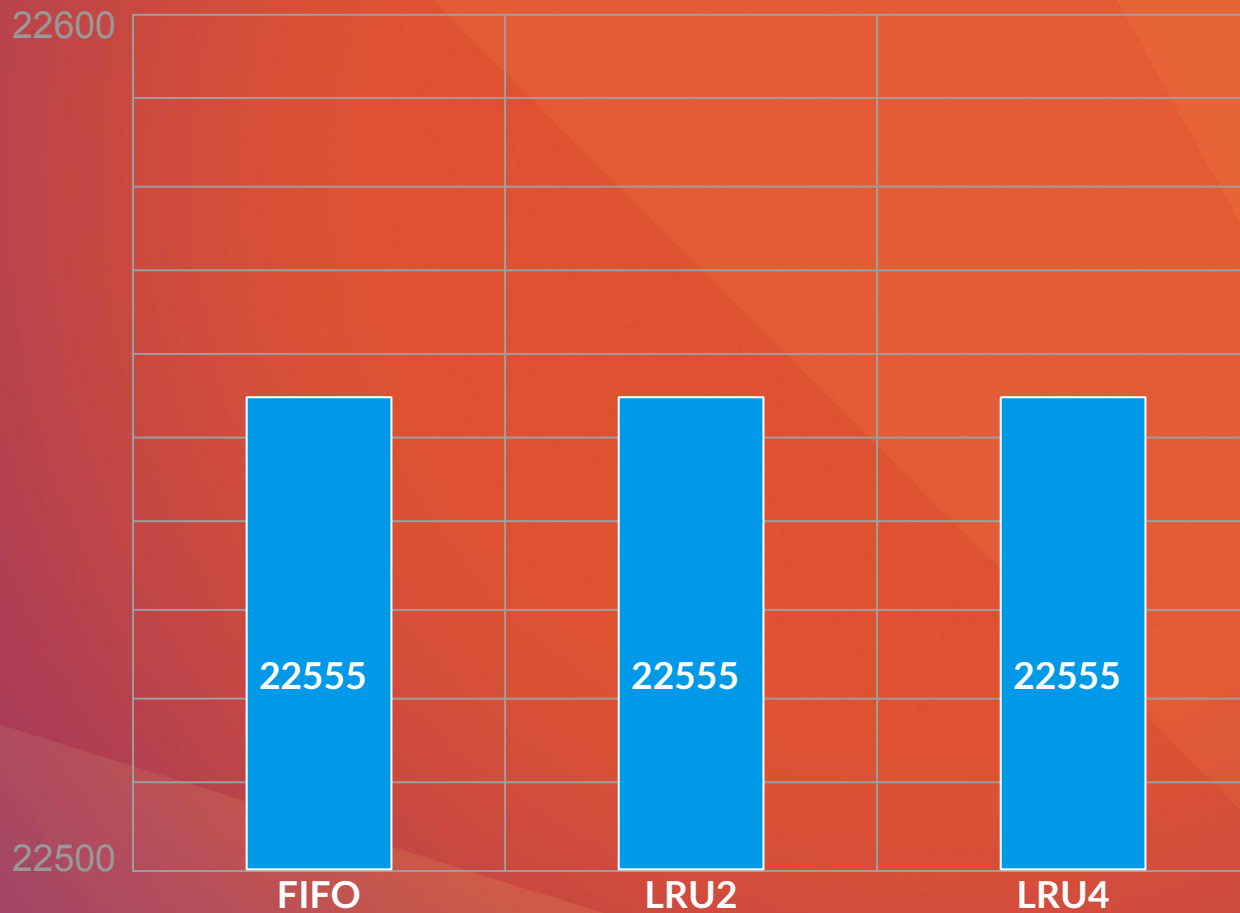
Gráficos

Tempo para
encontrar
espaço livre na
memória



Gráficos

Número de Page Faults



Conclusões

- Quick fit apresentou um tempo de execução maior do que o Best/Worst fit. Apesar de ir contra o esperado, acreditamos que o modo como foi implementado o algoritmo resultou nisso

- Best/Worst fit apresentaram tempo de consumo muito parecidos, o que é esperado: ambos varrem a memória inteira para fazer as mudanças
 - Mesmo número de page faults para os algoritmos de paginação, o que era esperado
-