

Simulador de Escalonamento de Processos com Gerência de Memória

Gabriel Silva & Thamires Bessa
Sistemas Operacionais I

Introdução

Introdução

Simular um sistema operacional que implementa a política de escalonamento Round Robin com feedback e com gerência de memória por paginação.

Embasamento Teórico

—

Conceitos importantes

- ❑ Gerência de Memória por paginação de memória virtual
- ❑ Tabela de páginas
- ❑ Estados dos processos

Premissas

- ❑ Política de escalonamento: Round Robin com feedback e filas de prioridade;
- ❑ estratégia de realocação: LRU (Least Recently Used) e tratamento de swapping;
- ❑ tempos de serviço, a quantidade de páginas virtuais de cada processo, e a quantidade de saídas para I/O de cada processo foram definidas de forma aleatória, utilizando uma distribuição uniforme;
- ❑ A memória principal será dividida em 64 frames;
- ❑ O working set limit para todos os processos é de 4 páginas virtuais por processo e este WSL é uma lista com apenas 4 posições;
- ❑ a fatia de tempo para o escalonador é 5 u.t;
- ❑ os tempos de I/O para impressora, disco e fita são respectivamente 15 u.t., 1u.t. e 5u.t.;

Premissas

- ❑ Spawn (criação) de processos a cada 3 segundos;
- ❑ número máximo de 18 processos;
- ❑ Cada processo pode ter no máximo 64 páginas;

Implementação



Dados técnicos

- ❑ Linguagem de programação escolhida: Python
- ❑ Biblioteca gráfica: matplotlib
- ❑ Bibliotecas adicionais: NumPy, SciPy, Pandas



Simulação

```
Desktop -- bash -- 138x44

fila de IO = []
p on processador = 6514
p on io = [0, 0, 1]
Tempo : 1803
PID(6514) - u.t.(2) - pagina referenciada = 19 - page fault ? S
Memória Principal = [ 6514, 6514, 6514, 6514, 6498, 6498, 6498, 6498, 0, 0, 0, 0, 6499, 6499, 6499, 6499, 6511, 6511, 6511, 6511
, 6505, 6505, 6505, 6505, 0, 0, 0, 0, 0, 0, 0, 6500, 6500, 6500, 6500, 0, 0, 0, 0, 6503, 6503, 6503, 6503, 6509, 6509, 6509, 6509, 0, 0
, 0, 0, 6513, 6513, 6513, 6513, 6502, 6502, 6502, 6497, 6497, 6497, 6497]
fila de BP = [ 6509(P), 6498(P), 6497(P), 6500(P), 6502(P), 6513(P), 6499(P), 6505(P), 6511(P), 6503(P)]
fila de AP = []
fila de IO = []
p on processador = 6514
p on io = [0, 0, 1]
Tempo : 1804
PID(6514) - u.t.(3) - pagina referenciada = 7 - page fault ? S
Memória Principal = [ 6514, 6514, 6514, 6514, 6498, 6498, 6498, 6498, 0, 0, 0, 0, 6499, 6499, 6499, 6499, 6511, 6511, 6511, 6511
, 6505, 6505, 6505, 6505, 0, 0, 0, 0, 0, 0, 0, 6500, 6500, 6500, 6500, 0, 0, 0, 0, 6503, 6503, 6503, 6503, 6509, 6509, 6509, 6509, 0, 0
, 0, 0, 6513, 6513, 6513, 6513, 6502, 6502, 6502, 6497, 6497, 6497, 6497]
fila de BP = [ 6509(P), 6498(P), 6497(P), 6500(P), 6502(P), 6513(P), 6499(P), 6505(P), 6511(P), 6503(P)]
fila de AP = []
fila de IO = []
p on processador = 6514
p on io = [0, 0, 1]
Tempo : 1805
PID(6514) - u.t.(4) - pagina referenciada = 21 - page fault ? S
Memória Principal = [ 6514, 6514, 6514, 6514, 6498, 6498, 6498, 6498, 0, 0, 0, 0, 6499, 6499, 6499, 6499, 6511, 6511, 6511, 6511
, 6505, 6505, 6505, 6505, 0, 0, 0, 0, 0, 0, 0, 6500, 6500, 6500, 6500, 0, 0, 0, 0, 6503, 6503, 6503, 6503, 6509, 6509, 6509, 6509, 0, 0
, 0, 0, 6513, 6513, 6513, 6513, 6502, 6502, 6502, 6497, 6497, 6497, 6497]
fila de BP = [ 6509(P), 6498(P), 6497(P), 6500(P), 6502(P), 6513(P), 6499(P), 6505(P), 6511(P), 6503(P)]
fila de AP = []
fila de IO = []
p on processador = 6514
p on io = [0, 0, 1]
Tempo : 1806
PID(6514) - u.t.(5) - pagina referenciada = 16 - page fault ? S
Processo 6514 volta para a fila de BP e estado Pronto
Processo 6509 entrou em execução via BP
Memória Principal = [ 6514, 6514, 6514, 6514, 6498, 6498, 6498, 6498, 0, 0, 0, 0, 6499, 6499, 6499, 6499, 6511, 6511, 6511, 6511
, 6505, 6505, 6505, 6505, 0, 0, 0, 0, 0, 0, 0, 6500, 6500, 6500, 6500, 0, 0, 0, 0, 6503, 6503, 6503, 6503, 6509, 6509, 6509, 6509, 0, 0
, 0, 0, 6513, 6513, 6513, 6513, 6502, 6502, 6502, 6497, 6497, 6497, 6497]
fila de BP = [ 6498(P), 6497(P), 6500(P), 6502(P), 6513(P), 6499(P), 6505(P), 6511(P), 6503(P), 6514(P)]
fila de AP = []
fila de IO = []
p on processador = 6509
```

```

PID(6514) - u.t.(4) - pagina referenciada = 1 - page fault ? N
Memória Principal = [ 6499, 6499, 6499, 6499, 6496, 6496, 6496, 6496, 6506, 6506, 6506, 6506, 6507, 6507, 6507, 6507, 6511, 6511
, 6511, 6513, 6513, 6513, 6513, 6497, 6497, 6497, 6497, 6514, 6514, 6514, 6514, 6510, 6510, 6510, 6510, 6504, 6504, 6504, 6504, 6505
, 6505, 6505, 6505, 6515, 6515, 6515, 6515, 6500, 6500, 6500, 6500, 6501, 6501, 6501, 6501, 6509, 6509, 6509, 6509, 6503, 6503, 6503, 6503
]
fila de BP = [ 6498(K), 6508(K), 6502(K), 6497(P), 6511(P), 6510(P), 6507(P), 6504(P), 6505(P), 6515(P), 6501(P), 6509(P), 6500(
P), 6499(P), 6496(P), 6503(P), 6513(P), 6506(P)]
fila de AP = []
fila de IO = []
p on processor = 6514
p on io = [0, 0, 1]
Tempo : 206
PID(6514) - u.t.(5) - pagina referenciada = 17 - page fault ? S
Processo 6514 volta para a fila de BP e estado Pronto
Processo 6497 Pronto/Suspensão
Processo 6498 na MP[24]
Processo 6498 entrou em execução via BP
Memória Principal = [ 6499, 6499, 6499, 6499, 6496, 6496, 6496, 6496, 6506, 6506, 6506, 6506, 6507, 6507, 6507, 6507, 6511, 6511
, 6511, 6511, 6513, 6513, 6513, 6513, 6498, 6498, 6498, 6498, 6514, 6514, 6514, 6514, 6510, 6510, 6510, 6510, 6504, 6504, 6504, 6504, 6505
, 6505, 6505, 6505, 6515, 6515, 6515, 6515, 6500, 6500, 6500, 6500, 6501, 6501, 6501, 6501, 6509, 6509, 6509, 6509, 6503, 6503, 6503, 6503
]
fila de BP = [ 6508(K), 6502(K), 6497(K), 6511(P), 6510(P), 6507(P), 6504(P), 6505(P), 6515(P), 6501(P), 6509(P), 6500(P), 6499(
P), 6496(P), 6503(P), 6513(P), 6506(P), 6514(P)]
fila de AP = []
fila de IO = []
p on processor = 6498
p on io = [0, 0, 1]
Tempo : 207
PID(6498) - u.t.(1) - pagina referenciada = 13 - page fault ? S
Processo 6498 foi para IO
Processo 6498 colocou o IO[0] de disco na fila
Processo 6498 Bloqueado
Processo 6511 Pronto/Suspensão
Processo 6508 na MP[16]
Processo 6508 entrou em execução via BP
Memória Principal = [ 6499, 6499, 6499, 6499, 6496, 6496, 6496, 6496, 6506, 6506, 6506, 6506, 6507, 6507, 6507, 6507, 6508, 6508
, 6508, 6508, 6513, 6513, 6513, 6513, 6498, 6498, 6498, 6498, 6514, 6514, 6514, 6514, 6510, 6510, 6510, 6510, 6504, 6504, 6504, 6504, 6505
, 6505, 6505, 6505, 6515, 6515, 6515, 6515, 6500, 6500, 6500, 6500, 6501, 6501, 6501, 6501, 6509, 6509, 6509, 6509, 6503, 6503, 6503, 6503
]
fila de BP = [ 6502(K), 6497(K), 6511(K), 6510(P), 6507(P), 6504(P), 6505(P), 6515(P), 6501(P), 6509(P), 6500(P), 6499(P), 6496(
P), 6503(P), 6513(P), 6506(P), 6514(P)]
fila de AP = []
fila de IO = []
p on processor = 6508

```

```
    fila de IO = []
p on processor = 6514
p on io = [0, 0, 1]
Tempo : 1003
    PID(6514) - u.t.(2) - pagina referenciada = 19 - page fault ? S
    Memória Principal = [ 6514, 6514, 6514, 6514, 6498, 6498, 6498, 6498, 0, 0, 0, 0, 6499, 6499, 6499, 6499, 6511, 6511, 6511, 6511
, 6505, 6505, 6505, 6505, 0, 0, 0, 0, 0, 0, 0, 0, 6500, 6500, 6500, 6500, 0, 0, 0, 0, 6503, 6503, 6503, 6503, 6509, 6509, 6509, 0, 0
, 0, 0, 6513, 6513, 6513, 6513, 6502, 6502, 6502, 6502, 6497, 6497, 6497, 6497]
    fila de BP = [ 6509(P), 6498(P), 6497(P), 6500(P), 6502(P), 6513(P), 6499(P), 6505(P), 6511(P), 6503(P)]
    fila de AP = []
    fila de IO = []
p on processor = 6514
p on io = [0, 0, 1]
Tempo : 1004
    PID(6514) - u.t.(3) - pagina referenciada = 7 - page fault ? S
    Memória Principal = [ 6514, 6514, 6514, 6514, 6498, 6498, 6498, 6498, 0, 0, 0, 0, 6499, 6499, 6499, 6499, 6511, 6511, 6511, 6511
, 6505, 6505, 6505, 6505, 0, 0, 0, 0, 0, 0, 0, 0, 6500, 6500, 6500, 6500, 0, 0, 0, 0, 6503, 6503, 6503, 6503, 6509, 6509, 6509, 0, 0
, 0, 0, 6513, 6513, 6513, 6513, 6502, 6502, 6502, 6502, 6497, 6497, 6497, 6497]
    fila de BP = [ 6509(P), 6498(P), 6497(P), 6500(P), 6502(P), 6513(P), 6499(P), 6505(P), 6511(P), 6503(P)]
    fila de AP = []
    fila de IO = []
p on processor = 6514
p on io = [0, 0, 1]
Tempo : 1005
    PID(6514) - u.t.(4) - pagina referenciada = 21 - page fault ? S
    Memória Principal = [ 6514, 6514, 6514, 6514, 6498, 6498, 6498, 6498, 0, 0, 0, 0, 6499, 6499, 6499, 6499, 6511, 6511, 6511, 6511
, 6505, 6505, 6505, 6505, 0, 0, 0, 0, 0, 0, 0, 0, 6500, 6500, 6500, 6500, 0, 0, 0, 0, 6503, 6503, 6503, 6503, 6509, 6509, 6509, 0, 0
, 0, 0, 6513, 6513, 6513, 6513, 6502, 6502, 6502, 6502, 6497, 6497, 6497, 6497]
    fila de BP = [ 6509(P), 6498(P), 6497(P), 6500(P), 6502(P), 6513(P), 6499(P), 6505(P), 6511(P), 6503(P)]
    fila de AP = []
    fila de IO = []
p on processor = 6514
p on io = [0, 0, 1]
Tempo : 1006
    PID(6514) - u.t.(5) - pagina referenciada = 16 - page fault ? S
    Processo 6514 volta para a fila de BP e estado Pronto
    Processo 6509 entrou em execução via BP
    Memória Principal = [ 6514, 6514, 6514, 6514, 6498, 6498, 6498, 6498, 0, 0, 0, 0, 6499, 6499, 6499, 6499, 6511, 6511, 6511, 6511
, 6505, 6505, 6505, 6505, 0, 0, 0, 0, 0, 0, 0, 0, 6500, 6500, 6500, 6500, 0, 0, 0, 0, 6503, 6503, 6503, 6503, 6509, 6509, 6509, 0, 0
, 0, 0, 6513, 6513, 6513, 6513, 6502, 6502, 6502, 6502, 6497, 6497, 6497, 6497]
    fila de BP = [ 6498(P), 6497(P), 6500(P), 6502(P), 6513(P), 6499(P), 6505(P), 6511(P), 6503(P), 6514(P)]
    fila de AP = []
    fila de IO = []
p on processor = 6509
```

Conclusão



Conclusão

Os processos seguiram a política correta de mudança de estados devido às saídas para I/O e por preempção e também seguindo a ordem circular e suas respectivas filas. Já a gerência de memória executou como esperado fazendo as mudanças de estados pertinentes e respeitando as restrições impostas pelo projeto da simulação.

Dúvidas e
perguntas?

Referências

—

Referências

- ❑ Operating Systems: Internals and Design Principles, de William Stallings
- ❑ <https://docs.scipy.org/doc/numpy/reference/generated/numpy.random.uniform.html>
- ❑ <https://docs.python.org/2/library/collections.html>
- ❑ <https://docs.python.org/2/tutorial/inputoutput.html#reading-and-writing-files>

Obrigado