

Modelo de Relatório do Projeto Simulador de Gestão de Processos Escalonamento e Gestão de Memória

André Ribeiro - 39686

José Costa - 39428

~~Diogo Nave - 34519~~

URL do projeto no GitHub - <https://github.com/VanSheep7/SO-Projeto.git>

URL do projeto no GitLab - <https://gitlab.com/vansheep/so/->

O PROJETO:

Temos vindo a desenvolver um simulador cujo código fonte está escrito na linguagem Python.

O simulador tem a capacidade de aceder e ler os ficheiros de texto progenitor e filho e interagir com os mesmos.

Faz escalonamento e também tem implementadas funções de gestão de memória.

Recebemos a notícia que um membro do nosso grupo por motivos próprios já não trabalha neste projeto. O projeto desenvolvido é da autoria de André Ribeiro e José Costa.

O programa consiste num simulador que interage com ficheiros, gere memória e executa processos de forma organizada por meio de operações de escalonamento, criação de processos, o seu término e permuta.

São exemplos das suas funcionalidades:

- . Leitura de ficheiros de texto
- . Acompanhamento e Alteração de estado dos processos
- . Organização de processos e o seu controle (PCB)
- . Gestão de memória e tempo

Conteúdo (resumido) de cada file

simulador1.py – Process Control Block (PCB);First Come First Served (FCFS)

simulador2.py – Priority Establishment

simulador3.py - Shortest Job First (SJF)

simulador4.py – Gerenciamento de memória

Algoritmos Implementados

Rate-monotonic scheduling (RMS)

First Come First Served (FCFS)

Observações e Informação adicional

O programa foi escrito em Python3.

Os ficheiros de texto utilizados estão presentes nas plataformas github e gitlab.

No código estão presentes alguns testes que ajudaram no tweaking do projeto, são irrelevantes ao funcionamento.

Exemplo da Execução do programa e verificação

```
andre@andrevm:~/Documentos$ python3 simulador4.py
1.025
Tempo--> 0
P1-V1
Tempo--> 1
P1-V1
Tempo--> 2
P2-V1
Tempo--> 3
P2-V1
Tempo--> 4
P2-V1
Tempo--> 5
P1-V2
Tempo--> 6
P1-V2
Tempo--> 7
P2-V1
Tempo--> 8
```

output do simulador4.py

O ficheiro simulador4.py tem as funcionalidades de memória. Este programa por meio do Bloco de controlo de processos armazena informações dos processos e também os divide em classes.

Está também neste ficheiro contido o algoritmo que escolhemos implementar, o Rate-monotonic scheduling (RMS), um processo com menor duração de funcionamento terá uma prioridade mais alta e será priorizado.

```
def Rate_Monotonic(x):
    global Tempo
    global Memory
    while(Tempo<x):
        #print("xxxx")
        verr()
        print("Tempo--> %d" % (Tempo))
        if(len(Memory)!=0):
            print("P%d-V%d"%(Memory[0].num,Memory[0].versao))
            Memory[0].tempo=Memory[0].tempo-1
            #print(Memory[0].tempo)
            if(Memory[0].tempo==0):
                Memory.pop(0)
            Tempo=Tempo+1
        print("Tempo--> %d" % (Tempo))
```

Excerto do código do simulador4.py

OUTPUTS OBTIDOS NA EXECUÇÃO DOS FICHEIROS

```

andre@andrevm:~/Documentos$ python3 simulator1.py
0
1
2
3
Execução---> 1 M 100 None
4
Execução---> 1 A 19 None
5
Execução---> 1 A 20 None
6
Execução---> 1 S 12 None
7
Execução---> 1 A 1 None
8
Execução---> 1 A 4 None
9
Execução---> 1 C 2 None
Execução---> 1 L 0 filho1
Execução---> 4 M 200 None
Execução---> 4 A 19 None
Execução---> 4 T 0 None
validação
4
Final no processo_concorrentes
12
Execução---> 1 C 2 None
Execução---> 1 L 0 filho2
Execução---> 5 M 300 None
Execução---> 5 S 12 None
Execução---> 5 A 5 None
Execução---> 5 T 0 None
validação
5
Final no processo_concorrentes
16
Execução---> 1 T 0 None
validação
1
17
Execução---> 2 M 200 None
18
Execução---> 2 A 19 None
19

```

```

Execução---> 2 T 0 None
validação
2
Tempo-> 20
Execução-> None
Prontos a executar-> 3
Terminados-> 4
Terminados-> 5
Terminados-> 1
Terminados-> 2
20
Execução---> 3 M 300 None
21
Execução---> 3 S 12 None
22
Execução---> 3 A 5 None
23
Execução---> 3 T 0 None
validação
3
24
fim
FCFS-->Final
14
3
0
13
10
10
9
5
5

```

simulator1.py

simulador2.py

```

andre@andrevm:~/Documentos$ python3 simulador2.py
0
1
2
3
Execução---> 1 M 100 None
4
Execução---> 1 A 19 None
5
Execução---> 1 A 20 None
6
Execução---> 1 S 12 None
7
Execução---> 2 M 200 None
8
Execução---> 2 A 19 None
9
Execução---> 2 T 0 None
10
Execução---> 1 A 1 None
11
Execução---> 1 A 4 None
12
Execução---> 1 C 2 None
Execução---> 1 L 0 filho1
Execução---> 4 M 200 None
Execução---> 4 A 19 None
Execução---> 4 T 0 None
Final no processo_concorrentes
12
Execução---> 1 C 2 None
Execução---> 1 L 0 filho2
Execução---> 5 M 300 None
Execução---> 5 S 12 None
Execução---> 5 A 5 None
12
Execução---> 4 M 200 None
Execução---> 4 A 19 None
Execução---> 4 T 0 None
Final no processo_concorrentes
Execução---> 1 T 0 None
16
Execução---> 3 M 300 None
17
Execução---> 3 S 12 None
18

```

```

Execução---> 3 A 5 None
19
Execução---> 3 T 0 None
Tempo-> 20
Execução-> None
Terminados-> 2
Terminados-> 4
Terminados-> 1
Terminados-> 3
20
fim
13
2
3
0
5
1

```

```

andre@andrevm:~/Documentos$ python3 simulador3.py
0
1
2
3
Execução---> 1 M 100 None
4
Execução---> 1 A 19 None
5
Execução---> 1 A 20 None
6
Execução---> 1 S 12 None
7
Execução---> 2 M 200 None
8
Execução---> 2 A 19 None
9
Execução---> 2 T 0 None
10
Execução---> 1 A 1 None
11
Execução---> 1 A 4 None
12
Execução---> 1 C 2 None
Execução---> 1 L 0 filho1
Execução---> 4 M 200 None
Execução---> 4 A 19 None
Execução---> 4 T 0 None
Final no processo_concorrentes
12
Execução---> 1 C 2 None
Execução---> 1 L 0 filho2
Execução---> 5 M 300 None
Execução---> 5 S 12 None
Execução---> 5 A 5 None
Execução---> 5 T 0 None
Final no processo_concorrentes
12

```

simulador3.py

```

Execução---> 1 C 2 None
Execução---> 1 L 0 filho2
Execução---> 5 M 300 None
Execução---> 5 S 12 None
Execução---> 5 A 5 None
Execução---> 5 T 0 None
Final no processo_concorrentes
12
Execução---> 4 M 200 None
Execução---> 4 A 19 None
Execução---> 4 T 0 None
Final no processo_concorrentes
Execução---> 5 M 300 None
Execução---> 5 S 12 None
Execução---> 5 A 5 None
Execução---> 5 T 0 None
Final no processo_concorrentes
Execução---> 1 T 0 None
Tempo-> 20
Execução-> None
Prontos a executar-> 3
Terminados-> 2
Terminados-> 4
Terminados-> 5
Terminados-> 1
20
Execução---> 3 M 300 None
21
Execução---> 3 S 12 None
22
Execução---> 3 A 5 None
23
Execução---> 3 T 0 None
24
fin
17
6
3
0
9
5

```

```

andre@andrevm:~/Documentos$ python3 simulador4.py
1.025
Tempo--> 0
P1-V1
Tempo--> 1
P1-V1
Tempo--> 2
P2-V1
Tempo--> 3
P2-V1
Tempo--> 4
P2-V1
Tempo--> 5
P1-V2
Tempo--> 6
P1-V2
Tempo--> 7
P2-V1
Tempo--> 8

```

simulador4.py