

Manual do Usuário - Gerenciador de Memória

1. Programas

Ao extrair o arquivo *.zip*, você verá que foram desenvolvidos dois programas para execução, um possui uma interface gráfica (*GerenciadorMemoria.jar*) e necessita que o JavaFX esteja instalado em seu computador, já o outro (*MainTerminal.jar*) foi desenvolvido para que sua execução ocorra somente através do Terminal. Os passos para executar *GerenciadorMemoria.jar* encontram-se na seção 2. Os passos para executar o programa desenvolvido para o Terminal encontram-se na seção 3.

2. Execução do Programa com interface gráfica

O código foi inteiramente desenvolvido em Java, portanto o seu executável é um arquivo com extensão *.jar*. O nome do arquivo executável é *GerenciadorMemoria.jar*, e para rodar, primeiramente você deve descompactar o *.zip* em uma pasta, logo após isso, basta ir até a pasta em que está o programa e dar um duplo clique no arquivo ou executar o seguinte comando pelo terminal: `java -jar GerenciadorMemoria.jar`

2.2. Tela Inicial

Ao executar o programa, irá abrir uma janela, na qual teremos a nossa tela principal, como pode ser visto na Figura 1. Esta é a tela onde vamos configurar o tamanho da memória, o seu tipo de partição (fixa ou variável) e o tamanho da partição, em caso de partição fixa, ou a política de alocação, em caso de partição variável. A inicialização do programa só será liberada após o usuário realizar a configuração da memória.



Figura 1. Tela Inicial do programa.

2.3. Configurações

Para realizar a configuração da memória, inicialmente o usuário deverá informar o seu tamanho (em potência de 2), conforme a Figura 2, esse tamanho é comum para ambos os tipos de alocação. A seguir, deverá ser informado a política de alocação, conforme a Figura 3, para podermos iniciar a execução das operações.

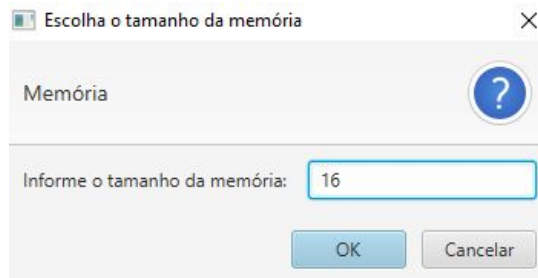


Figura 2. Tela para informar o tamanho da memória.

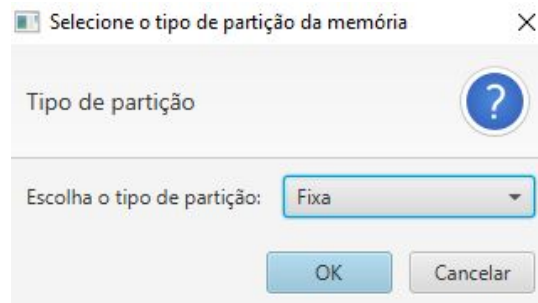


Figura 3. Tela para informar o tipo de partição.

2.3.1 Configuração para partição fixa

No caso de escolhido uma partição de tamanho fixo, o usuário informa o tamanho de cada partição, conforme a Figura 4.

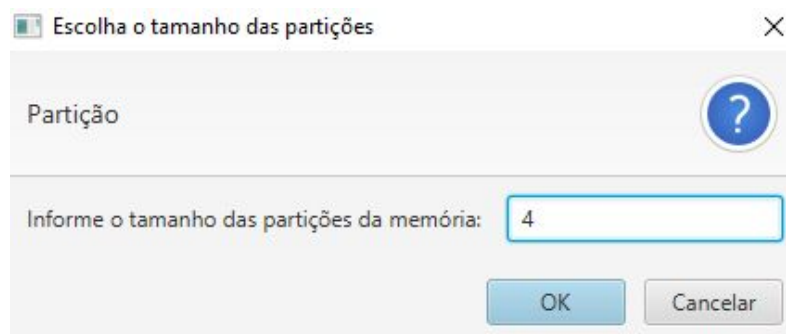


Figura 4. Tela para informar o tamanho da memória.

2.3.2 Configuração para partição variável

No caso de escolhido uma partição variável, o usuário informa a política de alocação da partição: best-fit, first-fit, worst-fit ou next-fit. Conforme Figura 4.

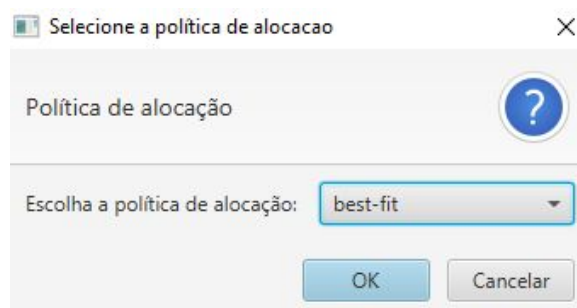


Figura 4. Tela para informar o tamanho da memória.

2.4. Execução

Após realizado a configuração da memória, o usuário poderá iniciar a execução, clicando no botão Iniciar. A tela de execução, conforme Figura 5, possui uma entrada de dados para a realização da simulação, que pode ser carregada através de um arquivo ou digitada manualmente, uma saída dos dados na memória e uma saída para validação da execução. Os resultados podem ser visualizados em uma execução direta ou uma execução passo a passo.



Figura 5. Tela de Execução do programa.

2.4.1. Entrada de dados

Como entrada do programa, pode ser carregado um arquivo *.txt* com as operações através do botão “Selecionar Arquivo” ou digitado manualmente no campo “Entrada”.

2.4.1. Memória

No campo “Memória”, é possível visualizar a configuração da memória, seu espaço total, espaço utilizado, espaço livre e os dados inseridos no passo atual.

Observação: Ao utilizar a execução rápida, o estado da memória visível é o estado após o último comando a ser realizado.

2.4.2. Saída de dados

Por uma execução direta, a saída de dados apresentada no campo “Saída” é o resultado após realizadas todas operações, e por uma execução passo a passo, o campo mostra as informações após cada operação na memória.

3. Execução do Programa através do Terminal

O código foi desenvolvido em Java, portanto a sua execução deve ser através do seguinte comando: `java -jar MainTerminal.jar`. Ao rodar o programa, primeiramente será necessário informar o tamanho da memória (em potência de 2) e logo após informar o tipo de partição (fixa ou variável). Caso a partição seja fixa, você deve informar o tamanho da partição, conforme Figura 6. Caso seja variável, você deve informar a política de de alocação,

conforme Figura 7.

```
Informe o tamanho da memoria. (tamanho equivalente a uma potencia de 2)
16
Informe o tipo de particao. ('fixa' ou 'variavel')
fixa
Informe o tamanho da particao.
4
```

Figura 6. Configuração da Memória de partição fixa.

```
Informe o tamanho da memoria. (tamanho equivalente a uma potencia de 2)
16
Informe o tipo de particao. ('fixa' ou 'variavel')
variavel
Informe o tipo de alocao. ('best-fit', 'worst-fit', 'first-fit', 'next-fit')
best-fit
```

Figura 7. Configuração da Memória de partição variável.

Para a entrada de dados, o código faz leitura do conteúdo do arquivo *entrada.txt*. Para mudar o conjunto de comandos que o programa lê, basta modificar esse arquivo antes de executá-lo.

3.1 Exemplo de Execução

Para o nosso exemplo, iremos utilizar uma memória com tamanho 16 posições, partição fixa e o tamanho da partição de 4 posições. A nossa entrada de dados será a seguinte:

- IN(A, 3)
- IN(B, 2)
- IN(C, 1)
- OUT(B)
- IN(D, 4)
- OUT(A)
- OUT(C)
- OUT(D)

Nessas condições obtemos a execução logo a seguir, ilustrada na Figura 8.

```
Informe o tamanho da memoria. (tamanho equivalente a uma potencia de 2)
16
Informe o tipo de particao. ('fixa' ou 'variavel')
fixa
Informe o tamanho da particao.
4

IN A,3           [16]
IN B,2           [13]
IN C,1           [1][10]
OUT B            [1][2][7]
IN D,4           [5][7]
OUT A            [1][7]
OUT C            [4][7]
OUT D            [4][8]
OUT D            [16]
```

Figura 8. Exemplo de Execução do Programa.