

MC202 - Estruturas de Dados

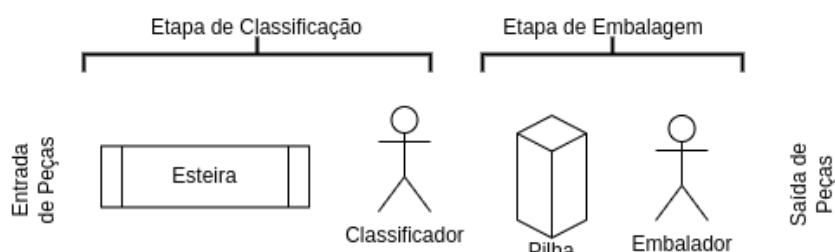
Lab 07 - Filas e Pilhas

Data da Primeira Chance: 02 de outubro de 2023

Link da atividade: <https://classroom.github.com/a/43yFX3rR>

Peso: 3

Antonio tem uma fábrica com tornos. As duas últimas etapas do processo são a etapa de **classificação** e **embalagem**. Essas duas etapas ainda são manuais, ou seja, feitas por pessoas e levam um tempo de acordo com cada peça. Segue o diagrama das etapas:



O tempo despendido pelo classificador e pelo embalador depende das características de cada peça. Para ajudar, Antonio criou uma tabela:

Tamanho e Tipo das Peças		Tempo Classificação	Tempo Embalagem
Pequena	1	1 minuto	1 minuto
	2		2 minutos
Média	1	2 minutos	2 minutos
	2		3 minutos
Grande	1	3 minutos	2 minutos
	2		3 minutos

O funcionamento da linha se dá conforme:

1. O **embalador**:
 - a. Embala a peça conforme o tempo da tabela.
 - b. Sempre que estiver livre e existir uma peça na pilha, ele a pegará para embalar
 - c. O embalador pode liberar uma peça e logo em seguida já pegar a próxima.
2. A **pilha**:
 - a. Recebe peças do classificador e alimenta o processo de embalagem

- b. Tem um tamanho limite, e não pode receber peças caso esteja cheia
 - c. O embalador sempre tentará retirar uma peça antes de o classificador colocar uma peça na pilha
 - d. As peças sempre passam pelo menos 1 minuto na pilha (o embalador não pegará uma peça no mesmo minuto que o classificador depositar a peça na pilha).
3. O **classificador**:
- a. Faz a classificação da peça conforme a tabela, e libera peças classificadas para a pilha
 - b. Sempre que estiver livre e existir uma peça na esteira, ele a pegará para classificar
 - c. O classificador pode liberar uma peça e logo em seguida já pegar a próxima.
4. A **esteira** (fila) de peças:
- a. Recebe peças do restante da linha de produção e alimenta o processo de classificação
 - b. Tem um tamanho limite, e não pode receber peças caso esteja cheia
 - c. É suficientemente rápida, significando que se a esteira estiver vazia, uma peça que entra na esteira pode ser classificada no minuto seguinte.
 - d. Caso já existam peças na esteira, elas são enfileiradas uma após a outra.

Atenção: Cada etapa descrita acima pode fazer com que a etapa anterior atrase. A cada momento que uma peça precisa ser colocada na esteira e ela esteja cheia, teremos 1 minuto de atraso.

Antonio te contratou para criar um simulador. Ele tem um relatório que diz a cada minuto qual peça entrou na esteira. Antonio deseja saber como seriam as entradas e saídas de peças na esteira e na pilha minuto a minuto, e também quantos minutos a esteira atrasou a entrada de peças.

Entrada

Na primeira linha da entrada, temos 2 inteiros a , b , onde a representa o tamanho da esteira e b representa o tamanho da pilha. Esses inteiros são maiores que 0.

Na segunda linha de entrada, temos 1 inteiro x , representando a quantidade de entradas no relatório.

Nas x linhas seguintes, temos a entrada de cada linha do relatório:

- Uma **string**, representando o tamanho da peça (podendo ser “pequena”, “media” ou “grande”).
- Um **inteiro**, representando a classe da peça (podendo ser ‘1’ ou ‘2’).

Obs.: Cada linha do relatório equivale a 1 minuto, e a primeira linha é o minuto 1!

Sempre haverá ao menos uma entrada de peças no relatório.

Saída

No final das suas simulações, Antonio precisa de um relatório contendo algumas as informações da simulação:

- Imprima uma linha contendo: ##### Simulação
- A cada minuto de simulação imprima os eventos que aconteceram nas duas etapas, a seguinte forma:
 - Min. %d: Classificação (+%d,-%d) | Embalagem (+%d,-%d).
Você deve imprimir quantas peças entraram (+%d) e quantas saíram (-%d) naquele minuto. As peças entram na etapa quando são adicionadas na esteira ou na pilha. E as peças saem da etapa no *minuto seguinte* ao que o classificador/embalador terminou seu processo.
- Na última linha, imprima o tempo total de atraso (que a esteira está cheia sem poder receber peças, lembre-se que se a esteira está parada e não tiver peça nova no minuto, não temos atraso): Tempo atraso: %d.

Exemplos

Exemplo 1:

Entrada

```
2 2
10
grande 1
grande 2
media 2
media 1
grande 1
media 1
grande 1
grande 1
grande 1
grande 1
```

Saída

```
##### Simulação
Min. 1: Classificação (+1,-0) | Embalagem (+0,-0).
Min. 2: Classificação (+1,-0) | Embalagem (+0,-0).
Min. 3: Classificação (+1,-0) | Embalagem (+0,-0).
Min. 4: Classificação (+0,-0) | Embalagem (+0,-0).
Min. 5: Classificação (+1,-1) | Embalagem (+1,-0).
Min. 6: Classificação (+0,-0) | Embalagem (+0,-0).
Min. 7: Classificação (+0,-0) | Embalagem (+0,-0).
Min. 8: Classificação (+1,-1) | Embalagem (+1,-1).
Min. 9: Classificação (+0,-0) | Embalagem (+0,-0).
Min. 10: Classificação (+1,-1) | Embalagem (+1,-0).
Min. 11: Classificação (+0,-0) | Embalagem (+0,-0).
Min. 12: Classificação (+1,-1) | Embalagem (+1,-1).
Min. 13: Classificação (+0,-0) | Embalagem (+0,-0).
Min. 14: Classificação (+0,-0) | Embalagem (+0,-0).
```

```
Min. 15: Classificação (+1,-1) | Embalagem (+1,-1).
Min. 16: Classificação (+0,-0) | Embalagem (+0,-0).
Min. 17: Classificação (+1,-1) | Embalagem (+1,-1).
Min. 18: Classificação (+0,-0) | Embalagem (+0,-0).
Min. 19: Classificação (+0,-0) | Embalagem (+0,-1).
Min. 20: Classificação (+1,-1) | Embalagem (+1,-0).
Min. 21: Classificação (+0,-0) | Embalagem (+0,-1).
Min. 22: Classificação (+0,-0) | Embalagem (+0,-0).
Min. 23: Classificação (+0,-1) | Embalagem (+1,-1).
Min. 24: Classificação (+0,-0) | Embalagem (+0,-0).
Min. 25: Classificação (+0,-0) | Embalagem (+0,-0).
Min. 26: Classificação (+0,-1) | Embalagem (+1,-1).
Min. 27: Classificação (+0,-0) | Embalagem (+0,-0).
Min. 28: Classificação (+0,-0) | Embalagem (+0,-0).
Min. 29: Classificação (+0,-1) | Embalagem (+1,-1).
Min. 30: Classificação (+0,-0) | Embalagem (+0,-0).
Min. 31: Classificação (+0,-0) | Embalagem (+0,-0).
Min. 32: Classificação (+0,-0) | Embalagem (+0,-1).
Tempo atraso: 10.
```

Exemplo 2:

Entrada

```
5 5
10
grande 1
grande 2
media 2
media 1
grande 1
media 1
grande 1
grande 1
grande 1
grande 1
```

Saída

```
##### Simulação
Min. 1: Classificação (+1,-0) | Embalagem (+0,-0).
Min. 2: Classificação (+1,-0) | Embalagem (+0,-0).
Min. 3: Classificação (+1,-0) | Embalagem (+0,-0).
Min. 4: Classificação (+1,-0) | Embalagem (+0,-0).
Min. 5: Classificação (+1,-1) | Embalagem (+1,-0).
Min. 6: Classificação (+1,-0) | Embalagem (+0,-0).
Min. 7: Classificação (+1,-0) | Embalagem (+0,-0).
Min. 8: Classificação (+1,-1) | Embalagem (+1,-1).
Min. 9: Classificação (+0,-0) | Embalagem (+0,-0).
Min. 10: Classificação (+1,-1) | Embalagem (+1,-0).
Min. 11: Classificação (+0,-0) | Embalagem (+0,-0).
Min. 12: Classificação (+1,-1) | Embalagem (+1,-1).
Min. 13: Classificação (+0,-0) | Embalagem (+0,-0).
```

```
Min. 14: Classificação (+0,-0) | Embalagem (+0,-0).
Min. 15: Classificação (+0,-1) | Embalagem (+1,-1).
Min. 16: Classificação (+0,-0) | Embalagem (+0,-0).
Min. 17: Classificação (+0,-1) | Embalagem (+1,-1).
Min. 18: Classificação (+0,-0) | Embalagem (+0,-0).
Min. 19: Classificação (+0,-0) | Embalagem (+0,-1).
Min. 20: Classificação (+0,-1) | Embalagem (+1,-0).
Min. 21: Classificação (+0,-0) | Embalagem (+0,-1).
Min. 22: Classificação (+0,-0) | Embalagem (+0,-0).
Min. 23: Classificação (+0,-1) | Embalagem (+1,-1).
Min. 24: Classificação (+0,-0) | Embalagem (+0,-0).
Min. 25: Classificação (+0,-0) | Embalagem (+0,-0).
Min. 26: Classificação (+0,-1) | Embalagem (+1,-1).
Min. 27: Classificação (+0,-0) | Embalagem (+0,-0).
Min. 28: Classificação (+0,-0) | Embalagem (+0,-0).
Min. 29: Classificação (+0,-1) | Embalagem (+1,-1).
Min. 30: Classificação (+0,-0) | Embalagem (+0,-0).
Min. 31: Classificação (+0,-0) | Embalagem (+0,-0).
Min. 32: Classificação (+0,-0) | Embalagem (+0,-1).
Tempo atraso: 2.
```

Exemplo 3:

Entrada

```
10 10
10
grande 1
grande 1
grande 1
grande 1
grande 1
pequena 2
pequena 2
pequena 2
pequena 2
pequena 2
```

Saída

```
#### Simulação
Min. 1: Classificação (+1,-0) | Embalagem (+0,-0).
Min. 2: Classificação (+1,-0) | Embalagem (+0,-0).
Min. 3: Classificação (+1,-0) | Embalagem (+0,-0).
Min. 4: Classificação (+1,-0) | Embalagem (+0,-0).
Min. 5: Classificação (+1,-1) | Embalagem (+1,-0).
Min. 6: Classificação (+1,-0) | Embalagem (+0,-0).
Min. 7: Classificação (+1,-0) | Embalagem (+0,-0).
Min. 8: Classificação (+1,-1) | Embalagem (+1,-1).
Min. 9: Classificação (+1,-0) | Embalagem (+0,-0).
Min. 10: Classificação (+1,-0) | Embalagem (+0,-0).
Min. 11: Classificação (+0,-1) | Embalagem (+1,-1).
Min. 12: Classificação (+0,-0) | Embalagem (+0,-0).
```

```
Min. 13: Classificação (+0,-0) | Embalagem (+0,-0).
Min. 14: Classificação (+0,-1) | Embalagem (+1,-1).
Min. 15: Classificação (+0,-0) | Embalagem (+0,-0).
Min. 16: Classificação (+0,-0) | Embalagem (+0,-0).
Min. 17: Classificação (+0,-1) | Embalagem (+1,-1).
Min. 18: Classificação (+0,-1) | Embalagem (+1,-0).
Min. 19: Classificação (+0,-1) | Embalagem (+1,-0).
Min. 20: Classificação (+0,-1) | Embalagem (+1,-1).
Min. 21: Classificação (+0,-1) | Embalagem (+1,-0).
Min. 22: Classificação (+0,-1) | Embalagem (+1,-1).
Min. 23: Classificação (+0,-0) | Embalagem (+0,-0).
Min. 24: Classificação (+0,-0) | Embalagem (+0,-1).
Min. 25: Classificação (+0,-0) | Embalagem (+0,-0).
Min. 26: Classificação (+0,-0) | Embalagem (+0,-1).
Min. 27: Classificação (+0,-0) | Embalagem (+0,-0).
Min. 28: Classificação (+0,-0) | Embalagem (+0,-1).
Min. 29: Classificação (+0,-0) | Embalagem (+0,-0).
Min. 30: Classificação (+0,-0) | Embalagem (+0,-1).
Tempo atraso: 0.
```

Dicas

Fique à vontade para criar quantas estruturas ou TADs forem necessários. Fique atento à ordem das operações que estão no enunciado. Trabalhe com uma operação por vez.

Regras e Avaliação

Neste laboratório, você deve usar a estrutura de dado “Pilha” para representar a pilha, e “Fila” para representar a esteira. Implemente as funções de Pilha e Fila em TADs. Crie um TAD também para representar a peça. Fique a vontade para criar um TAD para representar tanto o classificador quanto o embalador (eles podem ser o mesmo TAD com um parâmetro tipo). **Para as Pilha, utilize listas ligadas, já para Fila utilize vetores.**

Seu código será avaliado não apenas pelos testes do GitHub, mas também pela qualidade. Dentre os critérios subjetivos de qualidade de código analisaremos neste laboratório:

- A escolha de bons nomes de funções e variáveis;
- A ausência de trechos de código repetidos desnecessariamente;
- O uso apropriado de funções;
- A ausência de vazamentos de memória;
- A eficiência dos algoritmos propostos;
- A correta utilização das Estruturas de Dados;
- **Esse laboratório deve conter o uso de Pilhas e Filas!**

Note, porém, que essa não é uma lista exaustiva, pois outros critérios podem ser analisados dependendo do código apresentado visando mostrar ao aluno como o código poderia ser melhor.

Submissão

Você deverá submeter no repositório criado no aceite da tarefa. O arquivo principal deve ser nomeado `simulador.c`. Cada TAD deve ser implementada em arquivos diferentes. Não se esqueça de dar `git push` !

Atenção: O repositório da sua atividade conterà alguns arquivos iniciais. Fica **estritamente proibido** ao aluno alterar os arquivos já existentes, tais como o testador existente ou demais arquivos de configuração do laboratório.

Lembre-se que sua atividade será corrigida automaticamente na aba “Actions” do repositório. Confirme a correção e o resultado, já que o que vale é o que está lá e não na sua máquina.

Após a correção da primeira entrega, será aberta uma segunda chance, com prazo de entrega apropriado.