

Trabalho Estrutura de Dados

IFMG - Formiga

Denise Ferreira Garcia Rezende

Aluno: Gabriel Guimarães de Almeida

Curso: Bacharelado em Ciência da Computação

Período: 2º

1. Execução do Sistema

O sistema foi construído para ser executado em vários sistemas operacionais, pois o tal não utiliza comandos de um S.O. específico. Mas vale salientar que tal projeto não teve seu executável gerado e, para rodá-lo normalmente, deve-se utilizar a IDE *IntelliJ* juntamente com o *Java Development Kit* (JDK) na versão 1.8, preferencialmente.

2. Descrição das Classes

Foram desenvolvida classes para expressar cada elemento do contexto do trabalho de um modo que se torne muito mais fácil e produtivo e, para isso, foi usado a famosa Orientação à Objetos (O.O). Dentre elas estão:

- **Main:** Classe principal a qual possui a função *main*, a que é responsável pela inicialização de todo o sistema;
- **Constantes:** Classe que contém valores que são utilizados em todo o sistema como uma espécie de configuração;
- **Porto:** Classe que guarda uma lista de áreas de atracamento. Ela é, basicamente, o objeto essencial para que o contexto do trabalho seja suprido;
- **Navio:** Classe responsável por representar um navio real perante ao seu tempo de espera na fila de atracamento e suas pilhas de contêineres;
- **Container:** Representa os contêineres que são retirados dos navios e empilhados em travessas e, assim, depositadas em um pátio de armazenamento;
- **AreaAtracamento:** Visa interpretar a área que possui a fila de espera de navios os quais terão seus contêineres desempilhados;
- **ElementoNavio:** Elementos que a compõem as filas de navio que estão presentes nas áreas de atracamento;

- ElementoContainer: Elementos que compõem as pilhas de contêineres que estão presentes nas áreas de atracamento e nos navios;
- FilaNavio: Fila de navios que serão submetidos a um algoritmo para calcular quanto cada um irá levar para ter seus contêineres desempilhados;
- PilhaContainer: Pilha de contêineres que serão desempilhados (navio) e empilhados logo em seguida (área de atracamento).

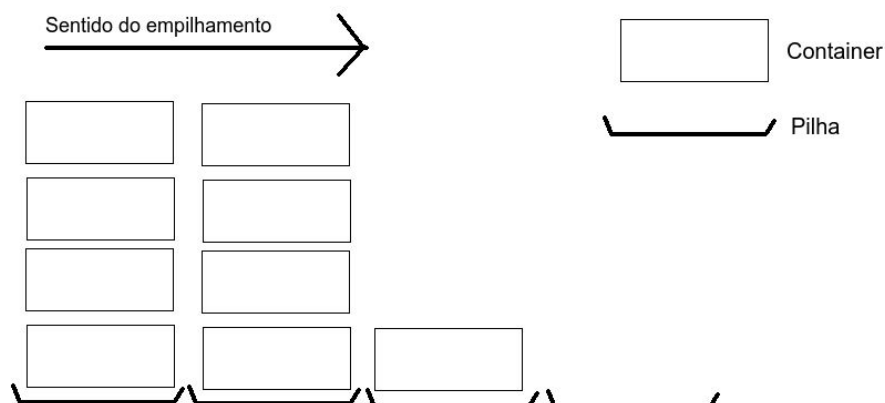
3. Funcionamento

O trabalho se encontra no seguinte endereço: <https://github.com/dealmeida-gabriel-1945/trabalho_pratico_ED_2020>, e teve dois viés de funcionamentos desenvolvidos os quais estão contidos nas *branches*: *main* e *adicionando-navio-a-cada-tic*.

- *main*:

Nessa *branch* foi utilizado a seguinte uma regra de negócio que se dá pelo fato de quando se iniciar o sistema, cada área de atracamento receberá um número aleatório de navios os quais também receberão um número randômico que corresponde a quantos contêineres cada um possuirá dentre suas quatro pilhas. Assim que tal valor se consolidar, o algoritmo irá começar a empilhar a quantidade correta de contentores dentro dos navios respeitando a seguinte regra:

Cada pilha poderá conter até 4 contêineres e para que uma comece a receber contentores a anterior deverá estar cheia.



Assim que tudo foi populado corretamente é trabalhado o porto e não é adicionado novos navios a cada “tic” de tempo e, no antes de finalizar, é mostrado o estado do porto após o trabalho.

- *adicionando-navio-a-cada-tic*:

Nessa *branch* está um algoritmo muito semelhante ao anterior, mas aqui é adicionado de zero à três navios na área de atracamento a cada “tic” de tempo até que o tempo de espera ultrapasse um máximo de tempo especificado pelo usuário. Esse limite foi necessário por conta do grande tempo que o sistema levava para se concluir quando o número de navios era muito grande.

Após cada navio ser desempilhado é gerado um *worklog*, que se dá em uma string ditando o identificador dele e seu tempo de espera e é concatenado ao *worklog* da área de atracamento, Assim, no final do código, todos esses *reports* serão listados.