

Aluno: Rubens Pessoa de Barros Filho  
Disciplina: Teste de Software  
Período: 2016.1

## Especificação do Sistema

### O sistema

O sistema será desenvolvido na linguagem de programação Java, em sua versão 1.8 utilizando a ferramenta *Maven* para gerenciamento de projeto.

Consistirá em um software para gerenciamento da estrutura de dados fila (ou *queue*) e esta fila carregará elementos do tipo de dados **inteiro** (internamente representados pela classe *Integer* da biblioteca da linguagem Java).

Inicialmente, este software receberá um número inteiro que representará internamente o tamanho máximo de elementos que nossa estrutura de dados poderá carregar. Este número deverá estar associado à estrutura de dados desde o momento de sua criação, ou seja, este inteiro que representa a capacidade máxima deverá ser passado como parâmetro do construtor na etapa de inicialização da estrutura de dados.

Posteriormente, o software rodará enquanto uma variável booleana *isRunning* tiver valor *True*. No caso desta variável ter o valor *True*, nosso software poderá receber alguns comandos, tais quais devem ser escritos em *lowercase* seguindo estritamente as regras da especificação (inclusive espaços em branco).

Desconsiderar aspas simples e os caracteres "<" e ">". Os nomes sublinhados serão substituídos de acordo com explicação à direita do comando.

São os comandos:

- "add <Inteiro>"
  - Inteiro poderá assumir qualquer valor inteiro, seja negativo, positivo ou neutro;
  - Este comando chamará um método similar presente na implementação da fila.
  - Não terá nenhuma forma de retorno;
  - Lançará *Exception* proveniente de método pertencente à fila.
- "remove"
  - Este comando chamará um método similar presente na implementação da fila.
  - Retornará o elemento do tipo inteiro que for removido da fila.
  - Lançará *Exception* proveniente de método pertencente à fila.
- "element"
  - Este comando chamará um método similar presente na implementação da fila.

- Retornará um elemento do tipo inteiro da fila de acordo com especificação dos métodos da fila.
  - Lançará *Exception* proveniente de método pertencente à fila.
- “size”
  - Este comando chamará um método similar presente na implementação da fila.
  - Retornará um número inteiro que representará a quantidade de elementos adicionados e presentes na fila no instante da invocação do comando. Desta forma, este número só poderá assumir valores positivos ou 0.
- “print”
  - Este comando chamará um método similar presente na implementação da fila.
  - Retornará uma String que será a representação da fila no momento da invocação do comando. Após sua invocação, este comando apagará todos os elementos da fila.
  - Lançará *Exception* proveniente de método pertencente à fila.
- “quit”
  - Este comando mudará o valor da variável booleana *isRunning* para False e desta forma, encerrará a execução do programa.

## A Estrutura de Dados (Fila)

Nosso software utilizará por debaixo dos panos a seguinte estrutura de dados abstrata: fila. Esta estrutura utiliza o raciocínio FIFO (*first in first out*) de maneira similar às filas que enfrentamos no cotidiano. Desta maneira, faremos uso da abstração proveniente do paradigma de programação baseado em Orientação a Objetos. Sendo Fila (*Queue*) uma classe e esta, em seu lugar, carregará em si alguns atributos e métodos que serão descritos a seguir:

Ressaltarei que a fila que será implementada deverá utilizar o conceito de Generics da linguagem Java, de forma a poder ser utilizada para qualquer tipo de dados, inclusive os criados por terceiros. Esta abordagem facilita a reutilização da nossa pilha em futuras aplicações.

### Atributos (Tipo - Nome da Variável : Especificação)

- `Object[] queue` : Array (nativo da linguagem Java. Proibido a utilização da biblioteca *Collections*) com elementos do tipo `Object` e de tamanho fixo determinado pelo usuário como descrito da especificação do sistema.
- `int first`: Inteiro que marcará o índice do primeiro elemento da fila.
- `int last`: Inteiro que marcará o índice do último elemento da fila.
- `int currentSize`: Inteiro que marcará a quantidade de elementos contidos na fila.

A forma que será manipulada as inserções e remoções na fila será com o índice do primeiro e último elementos da mesma. Desta forma acredito que seja uma maneira mais eficiente do que se fizéssemos uma moção de todos os elementos uma casa à frente sempre que um elemento fosse removido.

### Métodos (Assinatura do método : Especificação)

- `public QueueImpl(int size)` : Construtor da fila que aceita como parâmetro um inteiro que representa a capacidade máxima de elementos que a fila poderá carregar. Nesta função, o Array do tipo `Object[]` deverá ser inicializado com esse valor.
- `public void enqueue(T element)`: Método de inserção de elementos genéricos na fila. Deverá inserir sempre na primeira posição livre após o último elemento da fila. Caso se tente inserir mais elementos do que a fila pode carregar, deverá ser lançada a Exceção "`QueueIsFullException`". Ao fim de cada inserção, lembrar de atualizar os valores dos atributos (`first`, `last` e `currentSize`) de acordo com suas especificações. Esta função não retorna nenhum valor.
- `public int size()` : Método que retorna o valor do atributo *currentSize*.
- `public T element()` : Método que retorna o primeiro elemento da fila. Lança exceção em caso de a fila estar vazia. Não retorna nenhum valor.
- `public T dequeue()` : Método de remoção de elemento da fila. Deverá remover sempre o primeiro elemento da fila. Caso se tente remover elementos quando a fila já estiver vazia, deverá ser lançada uma exceção "`QueueIsEmptyException`". Ao fim de cada remoção, lembrar de atualizar os valores dos atributos (`first`, `last` e

currentSize) de acordo com suas especificações. Esta função retorna o elemento removido.

- `public String toString()` : Método para visualização dos elementos da fila. Deverá ser retornado uma `String` no seguinte formato: "Queue = {elemento1, elemento2, ... elementon}"
- `public void resetQueue()` : Método para reinicialização da fila. Excluirá todos os elementos da fila. Lançará exceção caso a fila esteja vazia. Esta função não retorna nenhum valor.