

## TAD Pilha sequencial/estática (atividade em dupla ou trio)

### Objetivos

- Verificar o entendimento e dificuldades com o TAD pilha sequencial.
- Resolver um problema com o uso de pilha.

### Instruções

- A atividade deve ser resolvida usando a linguagem Java e deve usar uma versão adaptada da implementação do TAD pilha sequencial/estática realizada durante as aulas.
- A solução não deve usar a classe `Stack` oferecida pela linguagem Java.
- Inclua a identificação do grupo (nome completo e RA de cada integrante) no início de cada arquivo de código, como comentário.
- Inclua todas as referências (livros, artigos, sites, vídeos, entre outros) consultadas para solucionar a atividade como comentário no arquivo `.java` que contém a `main()`. Caso use ChatGPT ou similar, inclua, em um documento PDF, o histórico da interação (prompt + respostas).

### Problema proposto

Escreva um programa que usa pilha para verificar e informar se uma expressão (string informada pelo usuário do programa) possui a correspondência e ordem correta dos símbolos `()`, `[]`, `{}` e `<>`.

Por exemplo, os símbolos na string `"{([{<>}])}"` estão corretos, enquanto que na string `"{([{{<>}]})}"` o primeiro `}` e o `>` estão na ordem errada.

Alguns exemplos que o programa deve informar “expressão OK”:

```
"(Hello([W]o)r)ld!"  
"<html>"  
"(a && b) || (c && d)"
```

E alguns exemplos que o programa deve informar erro de correspondência:

```
"[x + y]"  
"<body"  
"(( [ ] ) ) }"
```

### Entrega

Compacte o código-fonte (somente arquivos `*.java`) no formato `zip`.

**Atenção:** O arquivo `zip` não deve conter arquivos intermediários e/ou pastas geradas pelo compilador/IDE (ex. arquivos `*.class`, etc.).

**Prazo de entrega:** via link do Moodle até 08/03/2024 23:59.

## Critérios de avaliação

A nota da atividade é calculada de acordo com os critérios da tabela a seguir.

ITEM AVALIADO	PONTUAÇÃO MÁXIMA
[1] Implementação da operação de inserção da pilha ( <b>push</b> ).	1,5
[2] Implementação da operação de remoção da pilha ( <b>pop</b> ).	1,5
[3] Implementação da operação de consulta da pilha ( <b>top</b> ).	1,0
[4] Implementação das operações extras da pilha (construtor, <b>size</b> , <b>count</b> , <b>isEmpty</b> , <b>isFull</b> , <b>clear</b> ).	0,5/operação (máx. 3,0)
[5] Uso correto da pilha para resolução do problema.	2,0
[6] Funcionamento geral do programa, de acordo com o enunciado.	1,0

Tabela 1 - Critérios de avaliação.

A tabela a seguir contém critérios de avaliação que podem **reduzir** a nota final da atividade.

ITEM INDESEJÁVEL	REDUÇÃO DE NOTA
O projeto é cópia de outro projeto.	Projeto é zerado
O projeto usa a classe <b>Stack</b> oferecida pela linguagem Java.	Projeto é zerado
Há erros de compilação e/ou o programa trava durante a execução <sup>1</sup> .	-1,0
Não há identificação do grupo. Não há indicação de referências. Arquivos enviados em formatos incorretos. Arquivos e/ou pastas intermediárias que são criadas no processo de compilação ou pela IDE foram enviadas junto com o código-fonte.	-1,0

Tabela 2 - Critérios de avaliação (redução de nota).

O código-fonte será compilado com o compilador **javac** (21.0.2) na plataforma Windows da seguinte forma:

```
> javac *.java -encoding utf8
```

O código compilado será executado com **java** (21.0.2) na plataforma Windows da seguinte forma:

```
> java <Classe>
```

Sendo que **<Classe>** deve ser substituído pelo nome da classe que contém o método **public static void main(String[] args)**.

<sup>1</sup> Sobre erros de compilação: considere apenas erros. Não há problema se o projeto tiver *warnings* (embora *warnings* podem avisar sobre possíveis travamentos em tempo de execução, como loop infinito, divisão por zero, etc.).