

Instruções:

Este trabalho **deve ser feito em dupla** e consiste no desenvolvimento de uma solução para uma calculadora baseada em pilha para resolver operações lidas de um arquivo com o seguinte formato:

```
4
12
/
5
*
```

No exemplo acima, o resultado do processamento da expressão é 15.

As operações que serão colocadas no arquivo são as seguintes:

- **inteiro**: neste caso o número inteiro deve ser inserido na calculadora e torna-se disponível;
- **+**, *****: as operações de soma e multiplicação são executadas com os dois últimos números disponíveis;
- **-**, **/**: as operações de subtração e divisão são realizadas com os dois últimos números disponíveis. Atente para a ordem das operações: se no arquivo constar

2
3
/

 o resultado deve ser 1.5.

2
3
-

 o resultado deve ser 1.
- **pop**: descarta o último resultado (número) da calculadora;
- **dup**: repete o último resultado da calculadora;
- **swap**: troca de ordem os dois últimos resultados (se houver apenas um resultado na pilha, nada acontece);
- **chs**: troca o sinal do último resultado;
- **sqr**: calcula a raiz quadrada do último resultado.

Os números de entrada serão inteiros, mas em todas as operações o resultado deve ser calculado como *double* e devolvido para a calculadora para ser usado em novas operações.

Portanto, primeiro **deve ser implementada uma pilha usando, obrigatoriamente, estruturas encadeadas**. Esta pilha deverá ter **exatamente os seguintes métodos**: **push(e)**, **pop()**, **top()**, **size()**, **isEmpty()** e **clear()**.

Depois, esta pilha deverá ser usada para processar as operações da calculadora.

Quando a calculadora estiver pronta, deve ser executada com os arquivos disponíveis no Moodle.

A solução a ser implementada deve ler o arquivo e apresentar o resultado do processamento de cada um, considerando que alguns deles poderão ter erro de sintaxe (por exemplo, ter mais operadores do que operandos).

Além de apresentar o resultado das operações e informar quando ocorre um erro, também deverá ser processado o tamanho máximo atingido pela pilha para resolver a expressão. Um exemplo do que deve ser apresentado para um arquivo está ilustrado a seguir:

Arquivo:

```
10
20
/
2
+
5
*
```

Resultado armazenado no topo da pilha: 20.0

Tamanho máximo da pilha: 2

No final deve ser feito um relatório descrevendo o algoritmo implementado para a calculadora. É **obrigatório apresentar o algoritmo da calculadora, em linguagem algorítmica**, além de incluir um parágrafo comentando

o seu funcionamento e desempenho (notação O). Comentários sobre facilidade ou dificuldade encontrada para o desenvolvimento do trabalho também devem ser incluídos. Além disso, este relatório deve ter o *print* (impressão do resultado da execução do trabalho) do resultado do processamento de cada arquivo que será disponibilizado no Moodle (o valor ou se havia um erro de sintaxe), e o tamanho máximo atingido pela pilha para o seu cálculo.

Tarefas:

- Implementar uma pilha usando estruturas encadeadas.
- Implementar a calculadora especificada usando a pilha desenvolvida.
- Ler e avaliar as operações dos arquivos fornecidos.
- Escrever o relatório.

Entrega:

- Cada dupla deverá **entregar somente o relatório no formato pdf e o código fonte da implementação feita (apenas os arquivos .java)**.
- Deve ser feito o *upload* deste arquivo através do *Moodle* **até a data e horário especificado**.

Avaliação:

Os seguintes critérios de avaliação serão utilizados:

- **Implementação da solução:** será averiguada se a solução está completa, eficiente e correta, e a qualidade e clareza do código implementado.
- **Relatório com a descrição da solução:** será avaliada a escrita e a explicação de como o problema foi solucionado, e os resultados obtidos para cada expressão.

Observações:

- Os trabalhos que NÃO FOREM ENTREGUES através do Moodle **segundo as orientações aqui descritas**, até o dia e horário especificado, não serão avaliados!
- Trabalhos que apresentarem ERRO DE COMPILAÇÃO NÃO SERÃO CONSIDERADOS.
- Trabalhos que apresentarem CÓPIAS DAS SOLUÇÕES de outros colegas resultarão em NOTA ZERO para todos os alunos envolvidos.

Um exemplo de como fazer a leitura do arquivo:

```
Path path1 = Paths.get("arquivo.txt");
try(Scanner sc=new Scanner(Files.newBufferedReader(path1,Charset.defaultCharset())))
{
    sc.useDelimiter("[;\n]"); // separadores: ; e nova linha
    while (sc.hasNext()) {
        System.out.println(sc.next());
    }
} catch (IOException x) {
    System.err.format("Erro de E/S: %s%n", x);
}
```

Caso precise remover quebras de linhas, usar, por exemplo:

```
String s = s.replaceAll("\n", "");
s = s.replaceAll("\r", "");
s = s.replaceAll("\t", "");
```