

MC 302EF - Atividade de Laboratório no. 1

Objetivos

- Familiarização com conceitos básicos da Linguagem Java e com o ambiente de desenvolvimento (nos laboratórios o ambiente de desenvolvimento usado será o Eclipse).
- Introdução a orientação a objetos

Descrição do Problema [\[criado em 04/03/16\]](#)

Implementar uma classe que implemente frações ordinárias. Sua classe deve se chamar `Fracao` e um objeto dessa classe deve ter dois atributos numerador e denominador, ambos inteiros e os seguintes métodos públicos:

- construtor: `Fracao(int n, int d);`
- soma de uma fração a outra: `Fracao add(Fracao f);`
- subtração de frações: `Fracao sub(Fracao f);`
- produto de frações: `Fracao mult(Fracao f);`
- divisão de frações: `Fracao div(Fracao f);`
- conversão para float: `float toFloat();`
- conversão para string: `string toString();`
- comparação de frações: `int comparesTo(Fracao f);`
 - retorna -1, zero ou 1 se o resultado da comparação for 'menor', 'igual' ou maior, respectivamente.

Todos os métodos que retornam um objeto `Fracao` devem fazê-lo de forma que a fração representada esteja na forma simplificada ou seja, numerador e denominador devem ser primos entre si.

Exemplo de uso

A classe abaixo mostra um cenário de uso da sua classe `Fracao`:

```
class TestLab1 {
    public static void main(String[] args) {
        Fracao f1 = new Fracao(1,2);
        Fracao f2 = new Fracao(1,3);
        Fracao f3 = f1.add(f2);
        Fracao f4 = f1.sub(f2);
        Fracao f5 = f1.mult(f2);
        Fracao f6 = f1.div(f2);
        System.out.println("f1:"+f1.toString());
        System.out.println("f2:"+f2.toString());
        System.out.println("f3:"+f3.toString());
        System.out.println("f4:"+f4.toString());
        System.out.println("f5:"+f5.toString());
        System.out.println("f6:"+f6.toString());
        System.out.println("float (f3) :"+f3.toFloat());
    }
}
```

Saída esperada (para o exemplo)

```
f1: ( 1 / 2 )
f2: ( 1 / 3 )
f3: ( 5 / 6 )
f4: ( 1 / 6 )
f5: ( 1 / 6 )
f6: ( 3 / 2 )
float(f3): 0.833333
```

Data de entrega

A entrega desta atividade é opcional e não será considerada na avaliação do curso. Caso opte pela entrega, ela deverá ser feita até 12/03/2016

Submissão

A submissão desta atividade será feita através do sistema run.codes. Vocês vão receber um email com indicações quanto à submissão e também quanto à classe que será utilizada para testes.

Desafios (entrega opcional)

1. Escreva um método que dado um *string* representando uma *dízima periódica* retorne a *fração geratriz* correspondente. O seu método deve ser definido como

```
public static Fracao geratriz(String dizima) { ... }
```

O *string* que representa a *dízima periódica* deve ser da forma *xx.xx(ppp)* onde *x* representa um dígito da parte da *dízima* que não se repete e *p* representa um dígito da parte da *dízima* que se repete periodicamente. O número de dígitos entre parêntesis pode ser igual a zero, indicando inexistência da repetição (ex. "0.5", representando 1/2).

Exemplos de uso:

```
Fracao f1 = geratriz("0.(1)"); // 0.1111... = 1/9
Fracao f2 = geratriz("1.(3)"); // 1.3333... = 4/3
Fracao f3 = geratriz("0.(27)"); // 0.2727... = 3/11
Fracao f4 = geratriz("0.8(3)"); // 0.8333... = 5/6
```

2. Escreva o método que, dada uma fração, retorne um *string* contendo a representação da mesma como *dízima periódica*, segundo o padrão definido no item anterior. Seu método deve ser definido como

```
public String dizima() { ... }
```

Este item opcional usa recursos da linguagem ainda não apresentados em sala, como por exemplo operações com *strings*. Caso você tenha aceito o desafio e implementado os métodos solicitados, não precisa incluí-los na submissão ao Suzy – envie-os diretamente ao professor.