

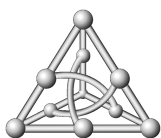
Prova Optativa 2

Algoritmos e Programação Orientada a Objetos II — 2018

Instruções para a realização da prova:

1. A prova contém **3 questões**, totalizando **10 pontos**;
2. A prova tem duração de **1 hora e 50 minutos**;
3. **Não altere a disposição do mobiliário da prova**;
4. **Coloque seu celular** em cima do gabinete, desligado. Caso o professor ouça o celular vibrando ou tocando durante a aula, o aluno receberá **nota zero**;
5. **Faça a prova em silêncio**; não converse durante a prova;
6. **Não tente plagiar a prova do(a) seu(sua) colega**, você pode prejudicar você e seu(sua) colega;
7. **Utilize apenas o que foi ensinado em sala de aula**. O uso de qualquer estrutura de programação ou estrutura de dados que não foi ensinada em sala de aula anulará a sua questão.
8. Utilize a IDE ou editor de texto de sua preferência;
9. Crie uma pasta com o seu nome dentro do diretório *Documentos*, e coloque o seu projeto dentro desta pasta.

BOA PROVA!



Questões

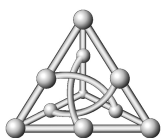
1. [3.50] Escreva um algoritmo recursivo que, dado um número n como entrada, calcula a seguinte soma:

$$\text{soma}(n) = 1 + \frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{n}$$

Seu método deve receber como parâmetro **APENAS** o inteiro n , e a recursão deve ser utilizada como estratégia para o acúmulo da soma, dispensando a necessidade de variáveis acumuladoras/contadores.

Para fins de teste, seguem abaixo valores esperados para o método $\text{soma}(n)$.

$n = 1,$	$\text{soma}(n) = 1.0$
$n = 2,$	$\text{soma}(n) = 1.5$
$n = 3,$	$\text{soma}(n) = 1.8333334$
$n = 4,$	$\text{soma}(n) = 2.0833335$
$n = 5,$	$\text{soma}(n) = 2.2833335$
$n = 6,$	$\text{soma}(n) = 2.4500003$
$n = 7,$	$\text{soma}(n) = 2.5928574$
$n = 8,$	$\text{soma}(n) = 2.7178574$
$n = 9,$	$\text{soma}(n) = 2.8289685$
$n = 10,$	$\text{soma}(n) = 2.9289684$



2. [3.50] *Patrulha Canina* é um seriado infantil que conquistou a criançada, sendo estrelado por 6 queridos cachorrinhos: *Marshall*, *Rubble*, *Chase*, *Rocky*, *Zuma* e *Skye*. O líder dessa patrulha é o menino *Ryder*.



A *Patrulha Canina* está recrutando novos membros para a trupe, e para isso, precisa de um sistema que permita cadastrar e gerenciar candidatos a patrulheiros caninos, e você modelará esse sistema utilizando orientação a objetos.

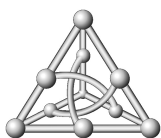
Para todos os candidatos, *Ryder* deseja saber o nome, idade e a lista de vacinas já tomadas. *Ryder* está interessado apenas em candidatos das raças *Labrador* e *Poodle*.

Para os candidatos da raça *labrador*, *Ryder* deseja saber o Q.I. (número) do cão (labradores são famosos por sua inteligência). Para os candidatos da raça *Poodle*, *Ryder* deseja saber quantas almofadas ele já destruiu até hoje (*Poodles* são famosos por serem bagunceiros e brincalhões).

Uma das missões de um futuro patrulheiro canino é salvar o dia, junto com a *Patrulha Canina*. Todos os candidatos a Patrulheiros devem ter condições de salvar o dia, cada um da sua maneira. Um *Labrador* salva o dia fazendo cálculos probabilísticos que indicam a chance de sucesso em uma missão da *Patrulha Canina*. Já os *Poodles* salvam o dia através da sua velocidade e agilidade nas missões.

Modele o problema acima utilizando todos os conceitos de orientação a objetos vistos em sala de aula. A atitude de "salvar o dia" pode ser modelada através de uma simples impressão na tela, que expressa o que o cãozinho faz ao salvar o dia.

Nesse exercício você não precisa criar um método *main* para testar as classes criadas. Você será avaliado pela capacidade de abstrair um contexto real utilizando orientação a objetos.



3. [3.50] Gustavo está organizando uma grande festa de ano novo em sua casa, onde receberá toda sua família. Para isso, Gustavo está pedindo que cada familiar leve um prato de comida para compor a mesa de jantar.

O grupo de *WhatsApp* da família está alvoroçado com o evento, e todos desejam saber se podem ou não podem levar um determinado prato. Gustavo já tem em mãos a lista do que cada pessoa disse que levará até o momento, no seguinte formato:

```
nome_do_familiar, nome_do_prato
```

Abaixo segue um exemplo dessa lista:

```
Vanessa, Brigadeiro  
Renata, Creme de Milho  
Neto, Churrasco  
Loide, Bobó  
Samuel, Brigadeiro  
Kathiuze, Creme de Milho
```

Crie um programa chamado *Reveillon*, que a partir das informações da lista fornecida por Gustavo, oferece dois relatórios referentes à festa de Reveillon da família:

- (a) Listagem dos pratos já informados por algum familiar;
- (b) Quantidade de pessoas que levarão cada prato.

O arquivo Algoritmos3.java (disponibilizado juntamente com o arquivo da prova) contém classes e métodos que efetuam a leitura do arquivo de entrada, o devolvendo já em um formato apropriado.

Neste exercício, você deve utilizar as coleções genéricas ArrayList, Map e Set.