

Disciplina	Curso	Turno	Período
Algoritmos e Estruturas de Dados I	Ciência da Computação	Manhã	1º
Professora: Rosilane Mota			

Lista 08 - Classes

1. Você está desenvolvendo um programa para armazenar e exibir dados pessoais. Cada pessoa é representada por um objeto da classe Pessoa, que possui atributos como nome, idade e altura. Seu programa deve ser capaz de receber um número inteiro N e, em seguida, ler os dados de N pessoas. Após a leitura, deve imprimir os dados de cada pessoa.

EXEMPLO ENTRADA

4

João 25 1.75

Maria 30 1.65

Carlos 22 1.80

Ana 28 1.70

SAÍDA ESPERADA:

Dados da pessoa:

Nome: João

Idade: 25 anos

Altura: 1.75 metros

Dados da pessoa:

Nome: Maria

Idade: 30 anos

Altura: 1.65 metros

Dados da pessoa:

Nome: Carlos

Idade: 22 anos

Altura: 1.8 metros

Dados da pessoa:

Nome: Ana

Idade: 28 anos

Altura: 1.7 metros

2. Você está desenvolvendo um programa para realizar o controle de tempo em um sistema. Cada operação é marcada com um horário específico, representado por um objeto da classe Relógio. A classe Relógio possui métodos para definir o horário, obter o horário formatado e avançar o horário para o próximo segundo.

Sua tarefa é ler um número inteiro N ($1 \leq N \leq 1000$) que representa a quantidade de operações a serem realizadas. Para cada operação, você deve receber um horário inicial e imprimir o horário após um segundo.

EXEMPLO ENTRADA

2
23 59 59
0 0 0

SAÍDA ESPERADA:

Horário inicial: 23:59:59
Novo horário: 00:00:00

Horário inicial: 00:00:00
Novo horário: 00:00:01

3. Você está desenvolvendo um sistema de controle de combustível para uma frota de carros. Cada carro é representado por um objeto da classe Carro, que possui características como capacidade do tanque, consumo de combustível, quantidade atual de combustível e distância percorrida.

A classe Carro também possui métodos para abastecer e mover o carro, atualizando a quantidade de combustível e a distância percorrida, respectivamente.

Sua tarefa é receber informações sobre dois carros e simular o movimento de cada um deles. Para cada carro, você deve ler o combustível inicial e a distância que será percorrida. Após a simulação, você deve imprimir a distância total percorrida e a quantidade de combustível restante para cada carro.

Entrada:

A entrada contém as seguintes informações:

Um número inteiro representando o combustível inicial do Carro 1.

Um número inteiro representando o combustível inicial do Carro 2.

Um número inteiro representando a distância que o Carro 1 irá percorrer.

Um número inteiro representando a distância que o Carro 2 irá percorrer.

Considere que as capacidades dos tanques dos carros são **50 litros** e que o **consumo de combustível** é o mesmo para ambos, **15 km/l**.

EXEMPLO ENTRADA

20
30
200
400

SAÍDA ESPERADA:

Carro 1:

Distância percorrida: 200

Combustível restante: 7

Carro 2:

Distância percorrida: 400

Combustível restante: 4

4. Você está desenvolvendo um sistema de biblioteca. Crie duas classes: Autor e Livro. A classe Autor deve armazenar o nome do autor, e a classe Livro deve conter informações sobre o título, ano de publicação e um ponteiro para um objeto da classe Autor.

Escreva um programa principal (main) que solicita ao usuário informações sobre um autor e um livro. Em seguida, imprima os detalhes do livro, incluindo o título, ano de publicação e nome do autor.

EXEMPLO ENTRADA

Mark Johnson

Mistérios Subaquáticos

2021

SAÍDA ESPERADA:

Detalhes do Livro:

Título: Mistérios Subaquáticos

Ano de Publicação: 2021

Autor: Mark Johnson

5. A PUC Minas Praça da Liberdade decidiu modernizar seus elevadores, e você foi designado para desenvolver um sistema de controle para os dois elevadores que estarão em operação. Sua tarefa é implementar a classe Elevador e criar um programa principal (main) que utiliza as funcionalidades desses elevadores.

Especificações:

Implemente a classe Elevador com os seguintes atributos privados:

andarAtual: o andar atual do elevador.

totalAndares: o total de andares no prédio da PUC Minas Praça da Liberdade.

capacidade: a capacidade máxima de pessoas no elevador.

personasPresentes: o número atual de pessoas no elevador.

A classe deve ter um construtor padrão e um construtor que recebe a capacidade e o total de andares como parâmetros.

Implemente o método inicializa que recebe a capacidade e o total de andares como parâmetros e inicializa os atributos do elevador.

Implemente os métodos entra e sai para simular a entrada e saída de pessoas do elevador, respectivamente. Lembre-se de verificar se há espaço ou pessoas suficientes para realizar a operação.

Implemente os métodos sobe e desce para simular o movimento do elevador para cima e para

baixo, respectivamente. Certifique-se de que o elevador não ultrapasse os limites do prédio. Implemente métodos de acesso (getAndarAtual, getTotalAndares, getCapacidade, getPessoas-
Presentes) para obter informações sobre o estado atual do elevador.

Programa Principal:

O programa principal deve realizar o seguinte:

Solicitar ao usuário a capacidade do elevador e o total de andares do prédio da PUC Minas
Praça da Liberdade.

Criar dois objetos Elevador com base nas informações fornecidas e inicializar os elevadores.

Solicitar ao usuário a quantidade de ações a serem simuladas.

Para cada ação, solicitar ao usuário a operação desejada (entrar, sair, subir, descer) e executar a
operação correspondente em um dos elevadores.

Após cada ação, imprimir o andar atual e o número de pessoas presentes no elevador correspon-
dente.

**Observação: Por favor, faça o código de forma descente para que o elevador do meio
pare de travar toda hora.**

EXEMPLO ENTRADA

```
8 10
5
entrar
subir
sair
descer
entrar
```

SAÍDA ESPERADA:

```
Andar atual: 0
Pessoas presentes: 1
Andar atual: 1
Pessoas presentes: 1
Andar atual: 1
Pessoas presentes: 0
Andar atual: 0
Pessoas presentes: 0
Andar atual: 0
Pessoas presentes: 1
```