

IFPB – Campus Campina Grande

Curso de Bacharelado em Engenharia de Computação

Disciplina: **Programação Orientada a Objetos (Teoria e prática) – Semestre 2021.1**

Professor: Leandro de Almeida Melo (leandro.melo@ifpb.edu.br)

Lista de Exercícios - II

1. Crie uma classe Agenda que armazena até 10 pessoas e que seja capaz de realizar operações como as que estão descritas a seguir.

Para a sua implementação da classe Agenda, pode ser usada esta implementação da classe pessoa.

```
class Agenda{
    Pessoa Povo[10];
public:
    void armazenaPessoa(string nome, int idade, float altura, string cidade, string bairro, in
numero);
    void removePessoa(string nome);
    int buscaPessoa(string nome); // informa em que posição da agenda está a pessoa
    void imprimePovo(); // imprime todos os dados de todas as pessoas da agenda
    void imprimePessoa(int i); // imprime os dados da pessoa que está na posição 'i' da
agenda
};
```

Na classe Pessoa, os atributos são nome, idade e Endereço. Observe que Endereço também é uma classe e seus atributos são cidade (string), bairro(string) e número(inteiro). Ambas as classes terão construtor, a classe pessoa terá um construtor que receberá e inicializará todas as variáveis de Pessoa e Endereço, para isso o construtor de Pessoa chamará o construtor de Endereço. Para essas duas classes, faça os métodos getters e setters que serão utilizados no sistema, ou seja, sem criar Sets e Gets desnecessários.

Faça um programa de teste (com main) que utilize a sua classe agenda.

2. Crie uma classe TurmaDeAlunos capaz de representar os dados referentes a uma turma de 10 alunos matriculados em uma disciplina.

A classe turma deve possuir, pelo menos, os seguintes atributos:

- lista de alunos: capaz de armazenar os dados de cada uma dos alunos matriculados. Cada aluno deve ser representado por um objeto da classe Pessoa (especificação abaixo);
- os dados do professor: objeto da classe Pessoa, capaz de armazenar os dados do professor;
- recursos: objeto da classe Recursos que deve indicar que recursos são necessários para esta turma. Veja abaixo a descrição da classe recursos.

```
class Pessoa {
```

```

private:
    string nome;
    int idade;
    float altura;
public:
    void setNome(string nome);
    void setIdade(int idade);
    void setAltura(float altura);
    string getNome();
    int getIdade();
    float getAltura();
    string toString();
};

```

A classe Recurso deve representar a necessidade ou não de um certo recurso para o grupo de alunos. A classe deve possuir um atributo do tipo bool para cada recurso possível e os respectivos sets e gets. Se um certo atributo contiver valor false o recurso não é necessário, caso contrário o recurso é necessário. A classe Recurso possui um construtor vazio e deve ser chamado junto com o construtor da classe TurmaDeAlunos.

```

class Recurso
{
    public:
        \\metodos
    private:
        int computadores, projetor, arcondicionado;
};

```

3. Implemente a classe Veículo. O veículo é composto por várias partes: um motor, um tanque de combustível e 4 pneus.

O funcionamento do veículo depende das suas partes, da seguinte forma:

Motor

- Possui uma potência (em hp), uma taxa fixa de consumo (em km/litro) e um tanque de combustível (ver abaixo).
- Possui um comando para avançar uma determinada quantidade de km. Se a pressão de mais de um pneu (ver abaixo) estiver abaixo de 20 lb, o consumo do veículo aumenta em 30%. Se não houver combustível suficiente, o veículo avança até o tanque esvaziar.

Tanque de Combustível

- Possui uma determinada capacidade e quantidade atual de combustível (ambos em litros).
- Pode ser abastecido com uma certa quantidade de combustível, limitado à sua capacidade máxima.

Pneu

- Cada pneu possui uma determina pressão (em lb).
- Pode ser calibrado com determinada pressão informada (positiva ou negativa, sendo somada à atual).

Considere que as partes do veículo devem ser modeladas como atributos. Quando aos métodos, implemente-os conforme as operações a serem realizadas com as partes do veículo, sem criar Sets e Gets desnecessários.

No programa principal, faça as seguintes operações:

- Instancie um veículo cujo motor tem 71 hp, consumo de 12 km/litro, tanque com capacidade para 50 litros, pneus dianteiros com 27 lb e traseiros com 23 lb.
- Abasteça o tanque com 30 litros.
- Exiba na tela as informações sobre cada componente do veículo.
- Avance 300 km.
- Exiba na tela as informações sobre cada componente do veículo.
- Reduza a pressão do pneu traseiro esquerdo para 17 lb.
- Exiba na tela as informações sobre cada componente do veículo.
- Avance 100 km.
- Exiba na tela as informações sobre cada componente do veículo.
- Abasteça 20 litros.
- Exiba na tela as informações sobre cada componente do veículo.
- Reduza a pressão do pneu dianteiro direito para 18 lb.
- Exiba na tela as informações sobre cada componente do veículo.
- Abasteça mais 10 litros.
- Exiba na tela as informações sobre cada componente do veículo.
- Avance 200 km.

A cada operação de movimento, mostre na tela a distância percorrida e a quantidade de combustível restante.

Referências

- Programação C/C++. Prof. Márcio Sarroglia Pinho.
<https://www.inf.pucrs.br/~pinho/PRGSWB/OO/oocpp.html>.
 - <https://www.inf.pucrs.br/~pinho/CPP/Exercicios/Classes/ExercicioVeiculo.html>
 - <https://www.inf.pucrs.br/~pinho/CPP/Exercicios/Classes/ExercicioAgenda.html>