

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)

Prova - I

Obs:. Em ambas as questões, você deve garantir o encapsulamento das classes.

1. Crie uma classe em C++ chamada ContaDePoupanca. Essa classe contém duas constantes indicando a taxaDeJurosAnual e a taxaDeSaque, com os valores respectivamente 3 (que representa 3%) e 2 (que representa 2%). Além disso, essa classe também possui dois atributos, um do tipo string que representa o nome do titular da conta e outro atributo que é número real (esse número sempre será ≥ 0) indicando a quantia que o poupador atualmente tem em depósito. Forneça os seguintes métodos públicos:

- **Inicializar** - Um construtor que receba como argumentos um nome (texto) e um valor (número real), e inicializa as variáveis do objeto.
- **calculeRendimentoMensal** - que calcule o rendimento mensal multiplicando o saldo pela taxaDeJurosAnual dividida por 12; este rendimento deve ser somado ao saldoDaPoupanca.
- **sacar** - no qual o usuário poderá retirar valores da quantia atual da poupança (saldoDaPoupanca), lembre-se que todo saque da conta poupança, além do valor solicitado, também deve ser debitado o valor da taxaDeSaque em relação ao valor solicitado, por exemplo: O usuário deseja sacar 110 reais, sendo a taxaDeSaque 2%, será retirado da poupança 112,2 reais(110 solicitado e 2,2 da taxa de saque).
- **toString** - retorna em forma de texto todas as informações do objeto, por exemplo: "Titular da conta: Leandro, saldo: r\$ 1000".
- Métodos **getters** e **setters** para todos os atributos da classe.

Escreva um programa de teste que instância dois objetos diferentes contaDePoupanca , poupador1 (nome=Leandro Almeida) e poupador2 (nome=Any Caroliny), com saldos de R\$ 2000,00 e R\$ 300,00, respectivamente. Realize o saque de 200 reais para o poupador1 e 300 reais para o poupador2. Em seguida, calcule o rendimento mensal dos dois objetos e imprima as informações atuais dos dois objetos usando o método toString.

2. Codifique a seguinte classe:

```
class Aluno {
    public:
        Aluno(string nome);
        void addNota(int unidade, float valor);
        bool alterarNota(int unidade, valor);
        void calculoMedia();
        string toString();
        string getNome();
        void setNome(string nome);
    private:
        string nome;
        float notas[3];
};
```

A disciplina possui 3 unidades, sendo assim cada aluno possui 3 notas que serão representadas em vetor de notas e cada índice indica a nota de uma unidade da disciplina (por exemplo, notas[0] guarda a nota da unidade 1 da disciplina). Além disso, leve em conta as seguintes observações:

- O construtor Aluno(string nome) atualiza a variável nome e inicializa o vetor de notas, para isso, inicialize todas as notas com o valor -1 (indicando que nenhuma nota foi atribuída ainda)
- Um aluno só pode ter no máximo 3 notas, caso o usuário tente adicionar mais de uma nota, o sistema deve informar que não é possível.
- Uma nota só pode ser alterada se ela já foi adicionada. Caso a alteração seja possível, atualize o valor e retorne true para o programa principal. Se a nota ainda não foi cadastrada, retorne false.
- A média só pode ser calculada se o aluno tiver todas as notas cadastradas. O método calcular média deve imprimir "aprovado" se o valor da média, com duas casas decimais, dos alunos for média ≥ 70 e mostra a média dele. Por outro lado, se o aluno obter uma média ≥ 40 e < 70 , o sistema deve imprimir a mensagem "Apto para o Exame Final" e mostra a nota mínima que o aluno precisa tirar na prova final*. Por fim, se a média do aluno for menor que 40, o sistema deve imprimir a mensagem "Infelizmente o aluno foi reprovado...".
- Uma nota só pode ser $0 \leq \text{nota}$ ou $\text{nota} \leq 100$.
- O método toString deve retornar as informações atuais do objeto em forma de texto, por exemplo: "Leandro, nota 1: 100, nota 2: 80, nota 3: 90. Média: 90.00". Lembre-se, se as notas estiverem incompletas, a média não deve ser exibida, por exemplo: "Leandro, nota 1: 100, nota 2: 80, nota 3: -1. Média: -".

Faça um programa principal e importe o arquivo Aluno.h. Crie dois alunos e teste todos os métodos da classe aluno.

* Para saber a nota mínima a ser tirada no Exame final, pegue o valor 100 e subtraia a média das unidades, a diferença representa a nota mínima a ser tirada para aprovação.