**Tarefas POO – Talles N1**

1- Tarefa 25/02/2025: Construa um tipo abstrato de dados que contenha pelo menos 2 atributos e 2 métodos que vão

manipular os atributos. Ou seja, definir uma classe com 2 métodos e 2 atributos. Além disso, construir um

programa em linguagem C++ para implementar o tipo abstrato de dados (TAD).

2- Tarefa 25/02/2025: Construir um programa em linguagem C++ para implementar o tipo abstrato de dados (TAD).

3- Tarefa 11/03/2025: Tarefa POO - Modificar o exemplo aluno3.cpp

(Classrom), incluindo mais 3 atributos e dois método.

#include <iostream>

#include <string>

class Aluno {

private:

std::string nome; // Atributo

public:

// Construtor

Aluno(const std::string& nome) : nome(nome) {}

// Método para exibir o nome do aluno

void exibirNome() const {

std::cout << "Nome: " << nome << std::endl;

}

};

int main(int argc, char\* argv[]) {

if (argc != 2) {

std::cerr << "Uso: " << argv[0] << " <nome\_do\_aluno>" << std::endl;

return 1;

}

// Obtendo o nome do aluno a partir dos argumentos da linha de comando

std::string nome = argv[1];

// Criando um objeto Aluno

Aluno aluno1(nome);

// Exibindo o nome do aluno

aluno1.exibirNome();

return 0;

}

4- Tarefa 25/03/2025: Definir uma classe para representar o cliente de um banco. Pelo menos 4 atributos e 4 métodos.

(banco), tem que ter o saldo e identificador do cliente e o identificador não pode ser o nome do cliente.

Não é pra usar o "construtor" e também definir no mínimo 4 métodos para manipular os atributos de cada cliente. Ex.:

método saque, tem que trabalhar saldo e subtrair e parâmetro que estou passando. Se estou criando um depósito tem que somar.

5- Tarefa 01/04/2025: Escreva um programa com duas classes que apresentem um relacionamento por herança. O programa deverá apresentar os métodos:

construir, destruir, getter, setter. No exemplo, deverão ser explorados os modificadores de tipos para atributos.

protected - Dentro da classe e das classes filhas (Permitir herança com controle)

private - Somente dentro da própria classe (Você quer esconder os dados de fora)

public - Livre acesso de fora da classe (Quando precisa ser acessado externamente)

6- Tarefa 08/04/2025: Produzir exemplos de programas que utilizam relacionamento de agregação e composição entre duas classes. Além da definição das

classes inserir métodos construtores, destrutores e getters/setters. Verificar e analisar as diferenças quanto a execução dos

exemplos de relacionamentos em C++.

7- Tarefa 15/04/2025: Tarefa POO -

item 1 - Defina exemplos de associações com diferentes cardinalidades.

Ou seja, 1:1, 1:N, N:N. Fazer o código em C++ e o diagrama de classes representando

suas associações.

Item 2. Da mesma forma que o item 1, explore o conceito de herança múltipla.