
LABORATÓRIO 4

FUNÇÕES MEMBRO

EXERCÍCIOS DE REVISÃO

VOCÊ DEVE RESPONDER PARA REVISAR OS CONCEITOS IMPORTANTES

1. Para que serve o operador de resolução de escopo?

2. Em que locais o operador de resolução de escopo não deve ser usado?

3. Existem duas formas de criar um método inline em C++. Quais são elas?

EXERCÍCIOS DE FIXAÇÃO

VOCÊ DEVE FAZER OS EXERCÍCIOS PARA FIXAR O CONTEÚDO

1. Considerando a classe Jogo mostrada abaixo.

```
class Jogo
{
private:
    string nome;           // nome do jogo
    float preco;           // preço do jogo
    int horas;             // quantidade de horas jogadas
    float custo;           // valor por hora jogada

    void calcular()        // calcular custo da hora jogada
    {
        if (horas > 0)
            custo = preco / horas;
    }

public:
    void adquirir(const string& titulo, float valor);
    void atualizar(float valor);
    void jogar(int tempo);
    void exibir();
};
```

- a. Implemente a função calcular() como uma função inline fora da declaração da classe. Mova a implementação para um arquivo fonte (Jogo.cpp) e teste essa modificação com um programa que chame os métodos jogar ou atualizar.
 - b. Separe a implementação da classe em dois arquivos diferentes, de forma que o método jogar fique em um arquivo (JogoX.cpp) e o método atualizar fique em outro arquivo (JogoY.cpp). Para que o código funcione, teste as duas soluções vistas na aula:
 - i. Duplica a definição do método calcular
 - ii. Coloque o método calcular no arquivo de inclusão
2. Modifique a implementação do método exibir de forma que os valores em Reais sejam exibidos sempre com 2 casas depois da vírgula. A modificação não deve alterar a interface da classe. Os programas que já utilizavam a classe devem continuar funcionando depois da modificação do método exibir.

```
void Jogo::exibir()
{
    cout << nome << " R$"
         << preco << " "
         << horas << "h = R$"
         << custo << "/h\n";
}
```

EXERCÍCIOS DE APRENDIZAGEM

VOCÊ DEVE ESCREVER PROGRAMAS PARA REALMENTE APRENDER

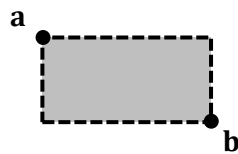
1. O tamanho de um objeto na memória é definido apenas pelo espaço ocupado pelos seus atributos ou inclui também os métodos? Construa um programa para fazer o teste e obter a resposta pelo uso do operador sizeof sobre um objeto.
2. Modifique a classe Point mostrada abaixo de forma a introduzir métodos para retornar os valores dos atributos x e y.

```
class Point
{
private:
    int x;
    int y;

public:
    void MoveTo(int px, int py);
    void Translate(int dx, int dy);
};
```

Construa e teste 3 soluções diferentes:

- a) Uma com métodos inline implementados dentro da classe
 - b) Uma com métodos inline implementados no arquivo .cpp
 - c) Uma com métodos inline implementados no arquivo .h
3. Dada uma classe Rect que define um retângulo por dois pontos, como mostrado na figura abaixo, construa a implementação dos métodos Create e Translate de forma a manter o tamanho do retângulo inalterado pela operação de movimentação.



```
class Rect
{
private:
    Point a;
    Point b;

public:
    void Create(int ax, int ay, int bx, int by);
    void Translate(int dx, int dy);
};
```