

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ (UTFPR)
 Campus de Curitiba-PR (Brasil) – DAELN - **Departamento Acadêmico de Eletrônica.**

Curso: **Engenharia Industrial Elétrica, ênfase Eletrônica/Telecomunicações.**

Disciplina: **Fundamentos de Programação 1 – IF62C – Turma: S11.**

Nome do Aluno: _____

Horário de Começo: _____ Horário de Fim: _____

Leia toda a prova antes de começar, pois os enunciados estão completados uns nos outros.

- Há cinco questões na prova, sendo que o aluno deverá responder corretamente 4 delas para obter 100% de aproveitamento. Ressalta-se que caso o aluno responda as cinco corretamente, sua nota será ainda 100% de aproveitamento.

(Questão 1) Transforme o programa em linguagem C abaixo em um programa C++ orientado a objetos, onde os dados de pessoas e seus tratamentos sejam programados em uma classe *Pessoa*. Os dados devem ser protegidos e corretamente iniciados quando da criação de objetos, por meio de mecanismo próprio para este fim. A classe *Pessoa* deverá ter, adicionalmente, um atributo nome.

```
#include <stdio.h>
struct Pessoa
{
    int dia;
    int mes;
    int ano;
    int idade;
};

int Calc_Idade ( struct Pessoa p, int ano )
{
    int idade = ano - p.ano;
    return idade;
}

int main ( )
{
    struct Pessoa Einstein, Newton;

    Einstein.dia = 14;
    Einstein.mes = 3;
    Einstein.ano = 1879;

    Newton.dia = 4;
    Newton.mes = 1;
    Newton.ano = 1643;

    Einstein.idade = Calc_Idade ( Einstein, 2007 );
    Newton.idade = Calc_Idade ( Newton, 2007 );

    printf ( "A idade de Einstein seria %d \n", Einstein.idade );
    printf ( "A idade de Newton seria %d \n", Newton.idade );

    getch();
    return 0;
}
```

(Questão - 2) Considerando o cabeçalho da classe abaixo, responda objetivamente as perguntas:

```
class Principal
{
private:
    Pessoa Einstein;
    Pessoa Newton;
public:
    Principal ();
    void Executar ();
};
```

A – Por que se aconselha ter uma classe *Principal*?

B – *Einstein* e *Newton* são *classes*, *objetos* ou *ponteiros* definidos no escopo da classe *Principal*?

C – O relacionamento de *Einstein* ou *Newton* para com a classe *Principal* é de associação, agregação ou herança?

D – Qual é a principal diferença entre elementos públicos e privados em geral?

E – Qual é a função do método *Principal()*? Como métodos desta natureza se chamam?

(Questão - 3) Elabore uma classe *Lista_Pessoas* para ponteiros de objetos da classe *Pessoa*. Os objetos da classe *Lista_Pessoas* poderão ter um número variável de elementos inclusos. Salienta-se ainda que cada *Pessoa* pode (potencialmente) participar de diversas listas no sistema.

Obs.1: Utilizar alocação dinâmica de memória para implementar a lista encadeada.

Obs. 2: A classe *Lista_Pessoas* deverá ter um método para listar bem como outro para incluir.

(Questão - 4) Faça com que a classe *Principal* tenha (um atributo privado que será) um objeto da classe *Lista_Pessoas*. A classe *Principal* deverá ter ainda um método para listar as *Pessoas*, bem como outro para incluí-las. Estes métodos basicamente chamarão métodos do objeto de *Lista_Pessoas*, passando os parâmetros necessários (quando necessários).

Obs.1: Para a questão ser corrigida, é necessário que pelo menos o cabeçalho (.h) da classe *Lista_Pessoas* tenha sido elaborado.

(Questão - 5) Elabore uma classe *Aluno*, derivada da classe *Pessoa*, normalmente elaborada na questão 1. Esta classe *Aluno* terá um atributo privado chamado RA (Registro Acadêmico) e, evidentemente, os respectivos métodos *get* e *set*.