

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
Campus de Curitiba (Brasil) - **Departamento Acadêmico de Eletrônica (DAELN)**.
Disciplina: **Fundamentos de Programação 1- IF61C/Turma: S11. Prof: Jean M. Simão.**
Curso: **Engenharia Industrial Elétrica, ênfase Eletrônica/Telecomunicações.**
Prova sobre linguagem C++ – Prova da 2ª Parcial.

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 equipamentos e seus tratamentos sejam programados em uma classe *Equipamento*. Os atributos relativos aos dados devem ser protegidos e corretamente iniciados quando da criação de objetos, por meio de mecanismo próprio para este fim.

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
```

```
struct Equipamento {
    char nome [150];
    int ano_fabricacao, antiguidade;
};
```

```
void DescobreAntiguidade ( struct Equipamento* equip, int ano_atual )
{
    equip->antiguidade = ano_atual - ( equip->ano_fabricacao );
}
```

```
void main ( )
{
    struct Equipamento robo, prensa;

    robo.ano_fabricacao = 1996;
    robo.antiguidade = -1;
    strcpy ( robo.nome, "robô transportador");

    prensa.ano_fabricacao = 1986;
    prensa.antiguidade = -1;
    strcpy ( prensa.nome, "prensa leve");

    DescobreAntiguidade ( &robo, 2008 );
    DescobreAntiguidade ( &prensa, 2008 );

    printf ( " O %s tem %d anos de fabricação. \n ", robo.nome, robo.antiguidade);
    printf ( " O %s tem %d anos de fabricação. \n ", prensa.nome, prensa.antiguidade);

    system ( "pause" );
}
```

Obs.: Utilizar apenas comandos de saída e (mesmo) de entrada efetivamente do C++ (i.e. *cout* e *cin*).

(Questão - 2) Evolua o programa C++, desenvolvido na primeira questão, de maneira a minimizar o uso da função *main*. Isto é, crie uma classe *Principal* e instancie um objeto dela na função *main*. Certamente, a classe *Principal* terá entre seus atributos os objetos *robo* e *prensa* (da classe *Equipamento*). A classe *Principal* terá também uma função-membro (i.e. um método) chamada *executar*, onde será chamada as funções-membro *DescobreAntiguidade* desses objetos.

(Questão - 3) Elabore uma classe *Lista_Equipamentos* para ponteiros de objetos da classe *Equipamento*. Os objetos da classe *Lista_Equipamentos* poderão ter um número variável de elementos inclusos. Salienta-se ainda que cada *Equipamento* pode participar de diversas listas.

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

Obs. 2: A classe *Lista_Equipamentos* 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_Equipamentos*. A classe *Principal* deverá ter ainda um método para listar os *Equipamentos*, bem como outro para incluí-los. Estes métodos basicamente chamarão métodos do objeto de *Lista_Equipamentos*, passando os parâmetros necessários (quando necessários).

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

(Questão - 5) Elabore uma classe *Carro*, derivada da classe *Equipamento*, normalmente elaborada na questão 1. Esta classe *Carro* terá dois atributos privados chamado *chassi* e *placa* que deverão (nos seus objetos) ser acessíveis de alguma maneira, por algum método. Ainda, instancie (i.e. crie) três objetos da classe *Carro* na classe *Principal*.