



Curso: Engenharia Eletrônica, **Disciplina:** Fundamentos de Programação II (IF62C), **Turma:** S__
Professores: Hermes Del Monego (), Jean Simão (), Robinson Vida () **Data:** __/__/____.

Aluno: _____ **Código:** _____, **Início** __:__:__ **Fim** __:__:__

1ª Prova

1) [3,0 pt] Definir uma classe base Pessoa. Esta classe deverá conter:¹

- a) como atributos privados o nome e a idade
- b) um construtor por default, sem parâmetros, que inicializa os atributos com "Indefinido" e 0, bem como um destrutor (vazio)
- c) um construtor alternativo, com parâmetros, que receba dados para os atributos definidos na classe
- d) métodos públicos sets e gets para acesso aos membros de dados privados
- e) uso de const conforme apropriado
- f) um método virtual chamado informa que imprime em tela os valores dos atributos.

Escrever um programa no qual haja:

- a) uma classe Sistema com construtor sem parâmetros e destrutor vazio, bem como dois objetos da classe Pessoa como atributos protegidos
- b) na classe Sistema um método protegido cadastrar que cadastra endereços/apontamentos de objetos do tipo Pessoa (citados no tópico anterior) em um vector<Pessoa*> chamado de V
- c) na classe Sistema um método listar que chama o método informa de objetos Pessoa apontados no vector<Pessoa*> V
- d) na função main um objeto apropriado (i.e. instância) da classe Sistema que chame de alguma forma o método listar.

2) [4,0 pt] Baseado na questão anterior, definir uma classe Professor. Essa classe deve conter:

- a) derivação da classe base Pessoa definida na questão anterior, mediante implementação de um mecanismo de herança pública

¹Questões inspiradas de: - Prof. Vitor E. Silva Souza, Curso Rápido de C++ - Lista de Exercícios. Programação III Inf 09331 2013/1 UFES CT DI.

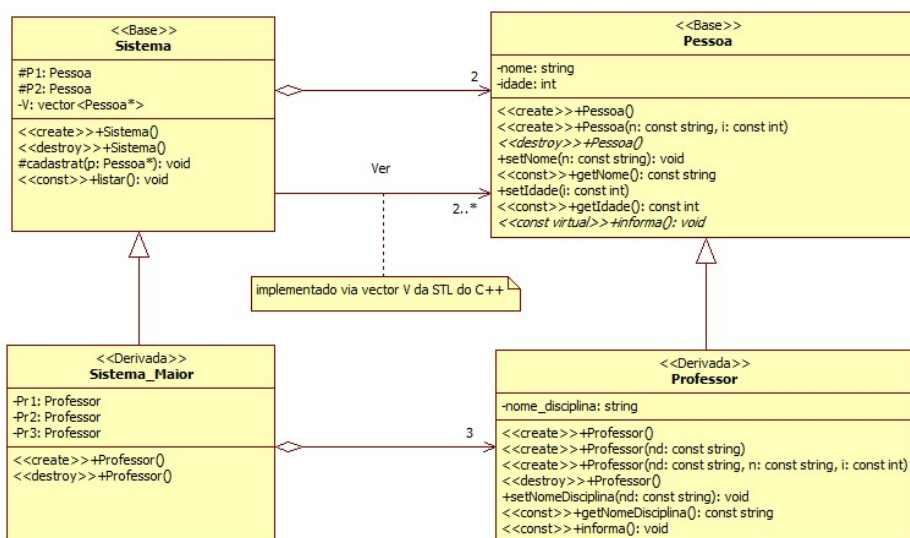
- b) um atributo privado nome_disciplina com respectivo métodos públicos set e get, bem como uso de const conforme apropriado
- c) um construtor por default que permita invocar explicitamente o construtor por default da classe base Pessoa
- d) um segundo construtor que permita criar objeto fornecendo apenas o nome da disciplina que leciona o professor e permita invocar implicitamente o construtor por default da classe base Pessoa
- e) um terceiro construtor que permita criar o objeto fornecendo o nome da disciplina que leciona o professor e demais dados necessários ao construtor com parâmetro da classe base Pessoa
- f) um método que redefina apropriadamente o método informa, bem como um método destrutor vazio.

Escrever um programa no qual haja:

- a) uma classe Sistema_Maior, derivado de Sistema (questão anterior), com construtor sem parâmetro e destrutor vazio
- b) na classe Sistema_Maior três objetos da classe Professor, como atributos privados, cujos endereços\apontamentos são incluídos em V citado na questão anterior
- c) via construtor da classe Sistema_Maior, cada objeto da classe Professor sendo criado com um construtor distinto (conforme os três tipos de construtores definidos na classe Professor)
- d) na função main um objeto apropriado da classe Sistema_Maior que chame de alguma forma o método listar

Observações para as duas questões acima:

1. Abaixo segue um diagrama de classes para fins de auxílio.
2. Cada item vale 10% da sua questão.



3) [3,0pt] Analise o código a seguir:

```
class Pai{
public:
    Pai(){}
    virtual ~Pai(){}
    void metodoComParametro(char *v){}
    virtual void metodoVirtualComParametro(char
*v){}
protected:
    void metodo1(){}
    virtual void metodo2(){}
};
class Filha : public Pai
{
public:
    Filha(){}
    virtual ~Filha(){}
    void metodoComParametro(int v){}
    void metodoVirtualComParametro(int v){}
protected:
    void metodo2(int v){}
    void metodo3(){}
    virtual void metodo4(){}
};
class Neta : public Filha
{
public:
    Neta(){}
    virtual ~Neta(){}
};
```

Um programador escreveu no método main, a seguinte instrução:

Filha *q = new Neta()

Marque V (verdadeiro) ou F (false) na lista a seguir, quais métodos podem ser chamados pelo objeto q.

- void metodoComParametro(char *)
- void metodoComParametro(int)
- void metodoVirtualComParametro(char *)
- void metodoVirtualComParametro(int)
- void metodo1()
- void metodo2()
- void metodo2(int)
- void metodo3 ()
- void metodo4()