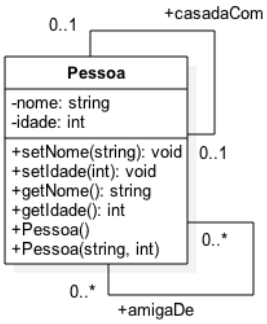
UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
DEPARTAMENTO ACADÊMICO DE ELETRÔNICA
(DAELN)

**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.** Na figura a seguir, um exemplo de Auto-Relacionamento **(Valor: 2,0)**




A seguir, a implementação parcial da classe Pessoa na linguagem a) C++ e b) Java. Nesta questão, você deverá modificar uma dessas implementações C++ ou Java de tal forma que seja possível representar o auto-relacionamento do diagrama UML fornecido.

```cpp
#include <string>

#ifndef PESSOA_H_
#define PESSOA_H_

class Pessoa {
public:
    Pessoa();
    virtual ~Pessoa();
    Pessoa(const std::string, int);

    std::string   getNome();
    int              getIdade();

    void setNome(const std::string);
    void setIdade(int);


private:
    std::string nome;
    int            idade;

protected:


};

#endif /* PESSOA_H_ */
```

```java
public class Pessoa {

    private String nome;
    private int  idade;

    Pessoa(String n, int i)
    { nome = n; idade = i;}

    Pessoa() {}

    public String getNome()
    {return nome;}

    public void setNome(String v)
    { nome = v; }

    public int  getIdade()
    {return idade;}

    public void setIdade( int v)
    {idade = v;}



}
```

**2.** Em um programa C++ (para console), crie uma classe chamada de *Ponto* com um atributo privado inteiro chamado *x* e outro chamado *y*, cada qual com seu respectivo *set* e *get*.Esta classe terá também um método virtual *"virtual void imprime() {draw(x,y,'.')}"* **(Valor: 5,0)**

a) Crie uma classe *Linha_Ortogonal*, derivada de Ponto. Tal classe terá atributos inteiros chamados *xf* e *yf*. Tal classe também terá um atributo do tipo *Vector* da **STL** chamado *conj_Pontos*. Na verdade, cada objeto de Linha_Ortogonal irá se compor por um conjunto de objetos do tipo Ponto alocados dinamicamente. Ainda, classe Linha_Ortogonal deve ter um método chamado *"void imprime()"*. Este método permitirá, a cada objeto de Linha_Ortogonal, desenhar (em tela) uma linha ou semi-reta ortogonal a partir de objetos *Pontos* agregados.

b) Crie uma classe Quadrado, derivada de Ponto, com quatro atributos da classe Linha_Ortogonal Ainda, a classe Quadrado deve ter um método chamado *"void imprime()"*. Tal método permitirá (em tela) desenhar um quadrado a partir do desenho das quatro linhas que a compõem.

c) Em uma função *main* instancie e execute ao menos um objeto de cada classe mencionada.

**3.** Dois alunos, atendendo ao pedido de um professor, forneceram os seguintes códigos em C++ para o mesmo problema dado. **(Valor: 2,0)**

```
#include <iostream>

class Pai
{
  public:
        Pai( ) { }
        virtual ~Pai( ) { }
        virtual void print( )
        { std::cout << " Sou Pai " << std::endl ;
}
};

class Filho : public Pai
{
public :
      Filho ( ) { }
      virtual ~Filho( ) {  }

      void print( )
      {  std::cout << " Sou filho " << std::endl;
}
};

int main( )
{
   Pai *x  = new Pai(  );
   Filho *y = new Filho (  );
   Pai *z  = new Filho ( );

   x->print( );
   y->print( );
   z->print( );

   delete x;  delete y; delete z;
   return 0;
}
```

```
#include <iostream>

class Pai
{
  public:
        Pai( ) { }
        virtual ~Pai( ) { }
        void print( )
        { std::cout << " Sou Pai " << std::endl ;
}
};

class Filho : public Pai
{
public :
      Filho ( ) { }
      virtual ~Filho( ) {  }

      void print( )
      {  std::cout << " Sou filho " << std::endl;
}
};

int main( )
{
   Pai *x  = new Pai(  );
   Filho *y = new Filho (  );
   Pai *z  = new Filho ( );

   x->print( );
   y->print( );
   z->print( );

   delete x;  delete y; delete z;
   return 0;
}
```

Responda:

a) Qual será o resultado impresso da execução desses dois programas?

b) as duas execuções fornecerão a mesma impressão? Se a sua resposta for negativa, justifique-a.

**4.** Em um programa em C++ é certo parametrizar o destrutor para inicializar os atributos desta classe? Por exemplo, *~classeNome(string nome)*. Justifique sua resposta em uma frase apenas. **(Valor: 1,0)**