FUNDAÇÃO OSWALDO ARANHA

CENTRO UNIVERSITÁRIO DE VOLTA REDONDA

CURSO DE GRADUAÇÃO EM SISTEMAS DE INFORMACÃO

TRABALHO DE PROGRAMAÇÃO ORIENTADA A OBJETOS

DANILO DUARTE RIBEIRO

SEGUNDA CHAMADA DE P.O.O.

VOLTA REDONDA

2020

# 

# 1.0 INTRODUÇÃO

O presente trabalho tem como objetivo apresentar a solução “SegundaChamada” com os formulários *Windows Forms* “CadastroDePessoas”, “PessoaFisica” e “PessoaJuridica”, além das classes “Pessoa”, “Fisica” e “Juridica”; como um trabalho para a segunda chamada da disciplina de Programação Orientada a Objetos do curso de Sistemas de Informação do UniFOA (Centro Universitário de Volta Redonda).

# 2.0 DESENVOLVIMENTO DA APLICAÇÃO

A aplicação tem como objetivo o cadastro de pessoas físicas e jurídicas e uma listagem com ambas, e consiste em um formulário principal chamado “CadastroDePessoas” que possui dois botões para chamar os formulários filhos “PessoaFisica” e “PessoaJurididca”, respectivamente, e uma *listbox* para exibir as pessoas cadastradas.

Ao escolher uma das opções, um dos formulários filhos mencionados anteriormente será exibido para preencher os dados que, por sua vez, são propriedades das classes “Pessoa”, “Fisica” e “Juridica” e se organizam da seguinte maneira: “Nome”, “Email” e “Telefone” pertencem a classe “Pessoa” e são herdados pelas classes “Fisica” e “Juridica”; o Cpf pertence a classe “Fisica” e o “Cnpj” pertence à classe “Juridica”.

Após preencher os dados, eles são gravados na interface “fisica”, para dados de pessoa física; e na interface “juridica”, para dados de pessoa jurídica, respectivamente, ambas baseando-se nas propriedades de suas devidas classes; além de chamar método polimórfico “TipoPessoa()” que define a variável “Tipo” com o respectivo tipo de pessoa para a qual se está realizando o cadastro.

Encerrando os formulários filhos, são chamados os métodos “GravarPessoaFisica()”, para pessoas físicas; e “GravarPessoaJuridica()”, para pessoas jurídicas, respectivamente. Ambos os métodos são responsáveis por extrair os valores das interfaces e inseri-los na *Listbox* “lstPessoas”.

# 3.0 APLICAÇÕES DOS ITENS SOLICITADOS

Abaixo seguem as aplicações de cada um dos itens solicitados:

## 3.1 Interfaces

Foram utilizadas como “intermediadores” de *Lists* para as classes “Fisica” e “Juridica”. Encontram-se no arquivo “CadastroDePessoas.cs” entre as linhas 15 a 18.

## 3.2 Classes Abstratas

Foi utilizada a classe abstrata “Pessoa” para reunir propriedades comuns entre as classes “Fisica” e “Juridica”.

## 3.3 Classe Concretas

As classes “Fisica” e “Juridica” são classes concretas baseadas na classe abstrata “Pessoa”.

## 3.4 Herança

As propriedades “Nome”, “Email” e “Telefone” da classe “Pessoa” são herdadas pelas classes “Fisica” e “Juridica”.

## 3.5 Polimorfismo

O método polimórfico “TipoPessoa()” foi implementado como *virtual* na classe “Pessoa” e como *override* nas classes “Fisica” e “Juridica”, sendo substituído e funcionando de maneira diferente para cada maneira por meio do qual é chamado.

# 4.0 DIAGRAMA DE CLASSES

A figura 1 mostra o diagrama de classes contendo as classes “Pessoa”, “Fisica” e “Juridica”.

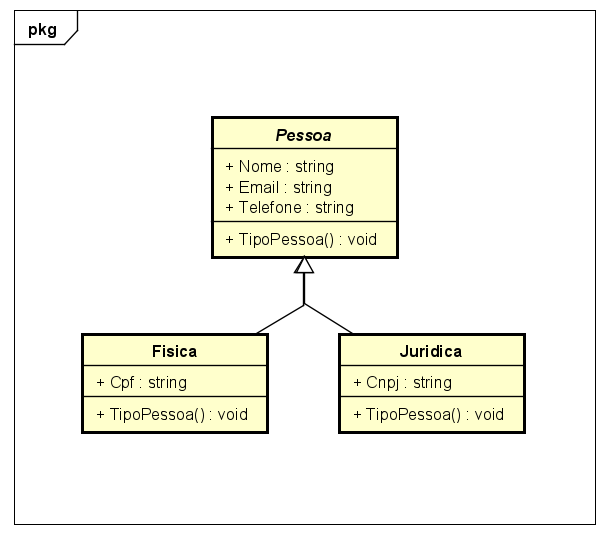


Figura 1: Diagrama de Classes

Fonte: Desenvolvido pela equipe

# 5.0 CONCLUSÃO

Baseando-se nos diversos itens e aplicações acima, conclui-se que o desenvolvimento de aplicações em *C#* e, mais especificamente, *Windows Forms*, detém a capacidade de serem neles utilizados diversos artifícios fundamentais na programação, tais como classes, herança, polimorfismo, e etc.; tornando tais tecnologias ótimas alternativas para o ramo de desenvolvimento de aplicações *Windows*.