

# 3º. Trabalho de POO

**Data de Entrega:** 23/11/2021

**Natureza do Trabalho:** em grupo de 2 a 3 alunos

**Forma de Entrega:** em formato eletrônico. A pasta do projeto deve conter os arquivos gerados pela ferramenta NetBeans 8.2 A pasta que vai conter os arquivos do trabalho deve ser nomeada da seguinte forma:

**prjPOON1N2N3**

onde **N1N2N3** representa os nomes dos integrantes

Exemplo: **prjPOOCarlosChristinaVinicius**

**Enunciado:**

- Faça as correções descritas no pdf **POO\_Correcao\_Trab\_1\_AD.pdf** disponibilizado na área da disciplina no Teams.
- Faça a implementação das operações de cadastro de Curso, de Turma, de Instrutor e de Aluno de acordo com as especificações abaixo e seguindo as interações entre as classes definidas no diagrama de classes do 1º. Trabalho com as respectivas Gui definidas no 2º. Trabalho.

## Especificação das Operações

### • Menu

- Na classe GuiMenu faça a inclusão das instruções abaixo:

```
private ArrayList<Pessoa>  cadAlunoInstrutor = new ArrayList<Pessoa>();  
private ArrayList<Curso>   cadCurso  = new ArrayList<Curso>();  
private ArrayList<Turma>   cadTurma  = new ArrayList<Turma>();
```

### • Cadastro de Instrutor

Na operação Consultar, caso seja informado um CPF inválido exiba uma mensagem de advertência e reposicione o foco na entrada de dados para o CPF. Caso o CPF seja válido, o cpf do instrutor deve ser pesquisado no ArrayList de Instrutor/Aluno. Caso seja encontrado o instrutor no ArrayList, o objeto da classe Instrutor deve ser instanciado e o conteúdo dos seus atributos devem ser exibidos na GUI, o botão Consultar deve ser desabilitado e os botões Alterar e Excluir devem ser habilitados, a caixa de edição do CPF deve ser desabilitada e as demais caixas de edição devem ser habilitadas. O foco deve ser posicionado na caixa de edição nome. Caso não exista no ArrayList um instrutor com o cpf informado, o botão Consultar deve ser desabilitado e o botão Incluir deve ser habilitado. A caixa de edição do CPF deve ser desabilitada e as demais caixas de edição devem ser habilitadas. O foco deve ser posicionado na caixa de edição nome. Os dados das entradas formatados devem inseridos no objeto Instrutor sem a formatação.

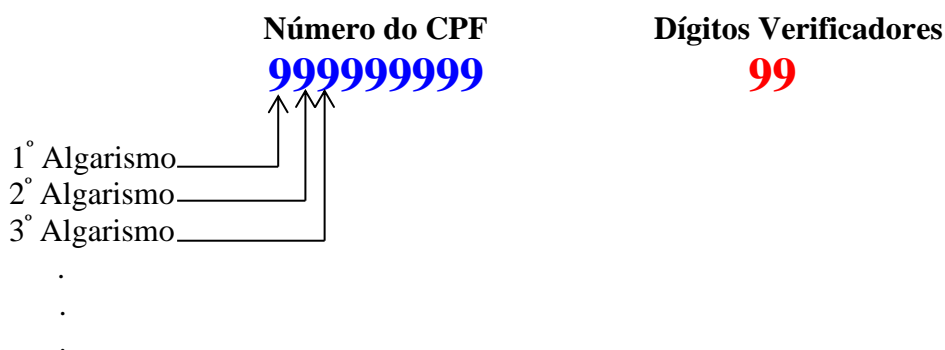
O CPF deve ser validado, para tal, na classe Pessoa defina um método de classe que tenha como entrada de dados o CPF e tem como retorno um valor booleano.

```
public static boolean validarCPF(<string>){..... return(<boolean>);}
```

O CPF deve ser validado conforme os critérios definidos abaixo:

### Forma do cálculo dos Dígitos Verificadores::

O CPF é um número composto por 9 algarismos. Os dígitos verificadores são calculados com base nos valores e na posição sequencial destes algarismos.



Inicialmente calcula-se o valor do primeiro dígito, que será utilizado posteriormente para o cálculo do segundo dígito.

#### ▪ Cálculo do 1º Dígito Verificador

1. Multiplicar cada um dos 9 dígitos iniciais pelo número correspondente a sua posição no CPF, e somar os resultados de cada multiplicação:

$$(\text{Algarismo1} * 1) + (\text{Algarismo 2} * 2) + (\text{Algarismo 3} * 3) + \dots + (\text{Algarismo 9} * 9)$$

2. Calcular o resto da divisão do valor obtido na soma por 11. Esse resto é o primeiro dígito verificador e, caso seja igual a 10, considerá-lo como sendo o número zero.

#### ▪ Cálculo do 2º Dígito Verificador

1. Multiplicar cada um dos 10 algarismos disponíveis (os 9 do CPF mais o primeiro dígito verificador já obtido) de acordo com a seguinte regra: O primeiro (o mais à esquerda) por 11, o segundo por 10, o terceiro por 9, sucessivamente até o último (o mais à direita, que é o primeiro dígito verificador) que será multiplicado por 2, e somar os resultados de cada multiplicação:

$$(\text{Algarismo1} * 11) + (\text{Algarismo2} * 10) + (\text{Algarismo3} * 9) + \dots + (1^\circ \text{ Dígito Verificador} * 2)$$

2. Multiplicar o valor obtido com a soma por 10 e, em seguida, calcular o resto da divisão desse resultado por 11. Esse resto é o segundo dígito verificador e, caso seja igual a 10, considerá-lo como sendo o número zero.
- Na operação Incluir deve ser instanciado um objeto da classe Vendedor que deve ser inserido na tabela de vendedores.
  - Na operação Alterar as alterações devem ser feitas a partir do objeto vendedor instanciado que deve ser atualizado na tabela de vendedores.
  - Na operação Excluir o objeto vendedor instanciado deve ser excluído da base de dados.
  - Os critérios para desabilitar e habilitar os recursos de interface conforme a execução das operações de inclusão, alteração e exclusão devem ser feitos conforme o projeto exemploDao.

#### • Cadastro de Aluno

Na operação Consultar caso seja informado um CPF inválido exiba uma mensagem de advertência e reposicione o foco na entrada de dados para o CPF. Caso o CPF seja válido, o cpf do aluno deve ser pesquisado no ArryList de Instrutor/Aluno. Caso seja encontrado o aluno no ArrayList, o objeto da classe Aluno deve ser instanciado e o conteúdo dos seus atributos devem ser exibidos na GUI, o botão Consultar deve ser desabilitado e os botões Alterar e Excluir devem ser habilitados, a caixa de edição do CPF deve ser desabilitada e as demais caixas de edição devem ser habilitadas. O foco deve ser posicionado na caixa de edição nome. Caso não exista no ArrayList um aluno com o cpf informado, o botão Consultar deve ser desabilitado e o botão Incluir deve ser habilitado. A caixa de edição do CPF deve ser desabilitada e as demais caixas de edição devem ser habilitadas. O foco deve ser posicionado na caixa de edição nome. Os dados das entradas formatadas devem ser inseridos no objeto aluno sem a formatação. O CPF deve ser validado.

- **Cadastro de Curso**

Na operação Consultar a sigla do curso deve ser pesquisada no ArrayList de Curso. Caso seja encontrado o curso no ArrayList, o objeto da classe Curso deve ser instanciado e o conteúdo dos seus atributos devem ser exibidos na GUI, o botão Consultar deve ser desabilitado e os botões Alterar e Excluir devem ser habilitados, a caixa de edição de Sigla deve ser desabilitada e as demais caixas de edição devem ser habilitadas. O foco deve ser posicionado na caixa de edição nome. Caso não exista ArrayList um curso com a sigla informada, o botão Consultar deve ser desabilitado e o botão Incluir deve ser habilitado. A caixa de edição da sigla deve ser desabilitada e as demais caixas de edição devem ser habilitadas. O foco deve ser posicionado na caixa de edição nome. Os dados das entradas formatadas devem ser inseridos no objeto Curso sem a formatação.

- **Cadastro de Turma**

No combobox de curso deve ser inserido a sigla de cada objeto curso inserido no ArrayList de Curso. Para tal, insira na classe Curso o método de classe **listarSiglas** (**public static ArrayList<String> listarSiglas(ArrayList<Curso>)**). Este método tem como retorno um ArrayList de String contendo as siglas de todos os objetos cursos inseridos no ArrayList de Curso. No método construtor da GuiTurma preencha o combobox, utilizando o método **addItem**, com a sigla de cada curso contida no ArrayList de String. Habilita a caixa de edição da Sigla do Turma. Na operação Consultar a sigla da turma deve ser pesquisada no ArrayList de Turma. Caso seja encontrado a turma no ArrayList de Turma, o objeto da classe Turma deve ser instanciado bem como o objeto da classe Curso que está associado com o objeto da classe Turma que foi pesquisado tendo os seus atributos exibidos na GUI, o botão Consultar deve ser desabilitado e os botões Alterar e Excluir devem ser habilitados, a caixa de edição de Sigla deve ser desabilitada e as demais caixas de edição devem ser habilitadas. O foco deve ser posicionado na caixa de edição nome. Caso não exista no ArrayList de Turma uma turma com a sigla informada, o botão Consultar deve ser desabilitado e o botão Incluir deve ser habilitado. A caixa de edição da sigla de turma deve ser desabilitada e as demais caixas de edição devem ser habilitadas. O foco deve ser posicionado na caixa de edição nome. Os dados das entradas formatadas devem inseridos no objeto Turma sem a formatação.