

## Atividade complementar 5

### Objetivo:

Implemente um sistema que modele diferentes tipos de pessoas (como médicos e pacientes) e permita a execução de comportamentos polimórficos relacionados ao agendamento de consultas, além de incluir métodos sobrecarregados para personalização de agendamentos.

### Requisitos

#### 1. Herança:

- Crie uma classe abstrata chamada Pessoa que será a classe base para Medico e Paciente.
- A classe Pessoa deve possuir atributos comuns como nome e cpf, além de métodos abstratos como `exibirInformacoes()`.

#### 2. Classes Abstratas e Polimorfismo:

- Implemente a classe abstrata Pessoa. A classe Medico deve sobrescrever o método `exibirInformacoes()` para exibir informações específicas do médico (como especialidade e CRM). A classe Paciente deve exibir as informações de endereço.

#### 3. Interface:

- Crie uma interface Agendavel com o método `agendarConsulta()`.
- Ambas as classes (Medico e Paciente) devem implementar a interface, mas de formas distintas.

#### 4. Overloading:

- Na classe Consulta, implemente métodos sobrecarregados para agendar consultas com base em diferentes critérios, como data/hora ou data/hora com prioridade.

#### 5. Demonstrativo:

- Crie um programa principal que demonstre o uso das classes, exibindo a funcionalidade polimórfica e a sobrecarga de métodos.

O diagrama de classes do projeto é mostrado na figura seguinte.

