

Herança em C++

Programação Orientada
a Objetos em C++

Introdução

- Ajuda na reutilização de códigos
- Extensibilidade do código
- Coerência com a Abstração.
- Define um padrão para as classes derivadas
- Define uma hierarquia “É-UM”

Exemplos

- Veículo:
 - Velocidade máxima
 - Tipo de combustível
 - Autonomia
- Carro “É-UM” Veículo
- Veículo **não** “É-UM” Carro
- Carro herda de Veículo

Exemplos

- Pessoa:
 - Nome
 - CPF
 - RG
- Estudante:
 - Nome
 - CPF
 - RG
 - Matrícula
- Estudante “É-UMA” Pessoa.
- Estudante “TEM-UMA” matricula
- **“É-UMA” é diferente de “TEM-UMA”**

Como saber se uma classe Herda ou possui Agregação?

- Faça a pergunta!
 - A “É-UM” B?
 - A “TEM-UM” B?

Exercícios

Classifique com “É-UM” (Herança) ou “TEM-UM” (Agregação)

- Carro _____ motor.
- Bicicleta _____ pedal.
- Televisão _____ eletrodoméstico.
- Moto _____ veículo.
- Candidato _____ número.
- Quadrado _____ forma geométrica.
- Computador _____ processador.

Sintaxe

```
class pessoa {
```

```
    ...
```

```
};
```

```
class estudante: public pessoa {
```

```
    ...
```

```
};
```

- estudante herda/extende pessoa
- estudante “É-UMA” pessoa
- pessoa é herdada de estudante

Exemplo

```
class Ave {  
public:  
    void voe() {  
        cout << "Ave voando" << endl;  
    }  
};
```

```
class Pinguim : public Ave{  
};
```

- Pinguim “É-UMA” Ave (correto)
- Pinguim voa???