

Nome: Guilherme Cavenaghi RA: 109317

- O que é IDE? Qual pode ser utilizada em Java? O que é necessário para rodarmos programas em Java na nossa máquina?

R. Uma IDE (Integrated Development Environment) é uma ferramenta de desenvolvimento que facilita a escrita, compilação e execução de código. Em Java, você pode usar o Eclipse, NetBeans ou IntelliJ IDEA. Para rodar programas Java na sua máquina, você precisa ter o JDK (Java Development Kit) instalado.

- Ambas as linguagens permitem implementações utilizando o paradigma orientado a objetos?

R. Ambas as linguagens suportam programação orientada a objetos.

- Os conceitos de abstração, classe, objeto, encapsulamento, herança e polimorfismo são os mesmos para ambas as linguagens?

R. C++ e Java compartilham os conceitos de abstração, classe, objeto, encapsulamento, herança e polimorfismo.

- Quais as diferenças entre as linguagens no contexto de herança?

R. A principal diferença é que em C++, você pode ter uma herança múltipla, o que não é permitido em Java. Java usa herança simples, mas permite a implementação de múltiplas interfaces.

- Mostre um exemplo de uma classe simples em Java.

```
public class Exemplo {  
    private int numero;  
  
    public Exemplo(int numero) {  
        this.numero = numero;  
    }  
  
    public int getNumero() {  
        return numero;  
    }  
}
```

- Mostre como é instanciado um objeto dessa classe em Java.

R. Exemplo `exemploObj = new Exemplo(42);`

- Quais as principais diferenças entre as linguagens no contexto de memória (alocação dinâmica e desalocação)?

R. Em C++, você pode alocar memória dinamicamente com `new` e liberá-la com `delete`.

Em Java, a alocação e desalocação de memória são tratadas automaticamente pelo coletor de lixo (garbage collector).

- O que são Interfaces em Java? Como uma classe com essa finalidade é implementada em C++ ?

R. Em Java, uma interface é uma coleção de métodos abstratos que uma classe pode implementar.

Em C++, você pode alcançar algo semelhante usando classes abstratas com métodos virtuais puros.

- Como é definida uma classe Abstrata em Java? Essa ideia de classe existe em C++?

R. Em Java, uma classe abstrata é definida usando a palavra-chave `abstract` antes da declaração da classe.

```
public abstract class MinhaClasseAbstrata {  
    // Atributos e métodos com implementação  
    public void metodoComImplementacao() {  
        // Implementação  
    }  
  
    // Método abstrato (sem implementação) que as subclasses devem fornecer  
    public abstract void metodoAbstrato();  
}
```

Em relação a C++, C++ também possui conceitos de classes abstratas, embora a sintaxe seja ligeiramente diferente. Em C++, uma classe abstrata é uma classe que contém pelo menos um método virtual puro, declarado com `virtual e = 0`.

```
class MinhaClasseAbstrata {  
public:  
    // Método virtual puro  
    virtual void metodoAbstrato() = 0;  
  
    // Outros métodos com implementação  
    void metodoComImplementacao() {  
        // Implementação  
    }  
};
```