

PROGRAMAÇÃO ORIENTADA A OBJETOS

LISTA DE EXERCÍCIOS 4 – PILARES DA POO SEM GABARITO

WALTER TRAVASSOS SARINHO

@WALTEROPROFESSOR

WALTER.TRAVASSOS@UNIFE.EDU.BR

Lista de Exercícios 4

Luke



Herança

Exercício 1

A utilização dos métodos getters e setters, à maneira dos declarados nas linhas 4, 7, 10 e 13 do código apresentado, é uma estratégia para aplicar um importante conceito de orientação a objetos chamado

- a) Generalização
- b) Herança
- c) Encapsulamento
- d) Sobrecarga
- e) Sobrescrita

```
1      public class Pessoa {  
2          private String nome;  
3          private int idade;  
4          public void setNome (String nome) {  
5              this.nome = nome;  
6          }  
7          public void getNome () {  
8              return this.nome;  
9          }  
10         public void setIdade (int idade) {  
11             this.idade = idade;  
12         }  
13         public void getIdade () {  
14             return this.idade;  
15         }  
16     }
```

Exercício 2

A herança é um dos princípios da programação orientada a objetos implementada na linguagem Java. Qual alternativa faz parte da especificação, em Java, de que uma classe B é subclasse de A?

- a) class A extends B
- b) class A superclass of B
- c) class B extends A
- d) class B specializes A
- e) class B subclasse of A

Exercício 3

A linguagem Java segue, em grande parte de sua concepção, o paradigma da Orientação a Objetos. Dessa forma, quanto melhor for o seu conhecimento sobre esse paradigma, melhor será o reuso do seu código e a resolução de alguns problemas triviais a partir de certos conceitos. Qual alternativa representa corretamente esses conceitos?

- a) Herança, Polimorfismo e Encapsulamento.
- b) Herança e Polimorfismo, apenas.
- c) Polimorfismo e Encapsulamento, apenas.
- d) Herança e Encapsulamento, apenas.
- e) Abstração e Encapsulamento, apenas.

Polimorfismo

Exercício 4

- Certo ou Errado?
- O polimorfismo de objeto pode criar objetos que herdam comportamento de uma classe e pode possibilitar que o desenvolvedor programe um comportamento diferente para os métodos de classes existentes, mas, para que seja da forma sobrescrita, o polimorfismo precisa possuir a mesma assinatura.

Exercício 5

Na programação orientada a objetos, as classes podem conter, dentre outros elementos, métodos e atributos. Os métodos...

- a) devem receber apenas parâmetros do mesmo tipo.
- b) não podem ser sobrecarregados em uma mesma classe.
- c) precisam possuir corpo em interfaces e classes abstratas.
- d) podem ser sobrescritos em aplicações que possuem relação de herança.
- e) definidos como `private` só podem ser acessados de classes do mesmo pacote.

Exercício 6

Assinale a opção correta quanto à abordagem conceitual de abstração sob o paradigma de programação orientada a objetos.

- a) As abstrações, idealmente, caracterizam-se por não serem grandes demais em comparação aos módulos, pois senão elas se tornam multifuncionais e de difícil compreensão. Como consequência, a abstração deve ser implementada apenas no nível de estruturas de dados necessários para se atingir o objetivo pretendido.
- b) Abstração consiste em uma linguagem puramente lógica. A motivação para isso veio em parte da vontade de se reconciliar o uso da lógica como uma linguagem declarativa de representação do conhecimento com a representação procedimental do conhecimento.
- c) Abstração é uma linguagem declarativa que permite acesso à base de dados mediante a utilização da teoria dos conjuntos e da álgebra relacional como fundamento de seu funcionamento.
- d) Abstração é um conceito segundo o qual o sistema ou software é dividido em partes distintas. Compõe o ferramental necessário para um programa mais legível com uma melhor manutenção e melhor desempenho por meio da programação orientada a objetos.
- e) Abstração é a habilidade de se concentrar nos aspectos essenciais de um contexto qualquer, ignorando características menos importantes ou acidentais. Em modelagem orientada a objetos, uma classe é uma abstração de entidades existentes no domínio do sistema de software.

Exercício 7

São conceitos chaves do paradigma Orientado a Objetos:

- a) Classes, objetos, regras e funções.
- b) Casamento de padrões, herança, classes e objetos.
- c) Classes, objetos, herança e polimorfismo por inclusão.
- d) Polimorfismo por inclusão, casamento de padrões, transparência referencial e herança.
- e) Sobrecarga, inferência lógica, backtracking e herança.

Exercício 8

O que será exibido no console?

```
1 public class Principal {
2
3     public static void main(String[] args) {
4
5         Pai p = new Pai();
6         Filha f = new Filha();
7         p.fazC();
8         f.fazC();
9         f.fazD();
10        p.fazB();
11    }
12 }
```

```
1 public class Filha extends Pai{
2
3     public void fazA() {
4         fazB();
5     }
6     public void fazD() {
7         fazA();
8     }
9     public void fazB() {
10        super.fazB();
11    }
12 }
```

```
1 public class Pai {
2
3     public void fazA() {
4         System.out.println("A");
5     }
6     public void fazB() {
7         System.out.println("B");
8     }
9     public void fazC() {
10        System.out.println("B");
11    }
12 }
```


Exercício 9

No contexto de programação orientada a objetos, considere as afirmativas abaixo.

- I. Objetos são instâncias de classes.
- II. Herança é uma relação entre objetos.
- III. Mensagens são formas de executar métodos.
- IV. Classes são apenas agrupamentos de métodos.
- V. Ocorre herança múltipla quando mais de um método é herdado.
- VI. Herança é uma relação entre classes.

Está correto o que se afirma APENAS em

- a) I, III e IV. b) I, III e VI. c) III, IV e VI. d) II, III e V. e) II, IV e V.

Exercício 10

Na orientação a objetos, em uma relação de herança entre classes, a subclasse herda da superclasse

- a) apenas as variáveis públicas de instância.
- b) apenas os métodos e variáveis de instância públicos.
- c) todas as variáveis de instância e apenas os métodos estáticos.
- d) todas as variáveis e métodos, exceto os públicos e os que foram sobrescritos.
- e) todas as variáveis de instância e os métodos, entretanto, podem explicitamente sobrescrever alguns destes componentes.

Exercício 11

Sobre herança na orientação a objetos com Java é correto afirmar:

- a) Um erro de compilação ocorre se um construtor de subclasse chamar um de seus construtores de superclasse com argumentos que não correspondem exatamente ao número e tipos de parâmetros especificados em uma das declarações do construtor da superclasse.
- b) Com a herança, as variáveis de instância comuns e os métodos de todas as classes na hierarquia são declarados em uma subclasse. Quando as alterações são requeridas para esses recursos comuns, os desenvolvedores só precisam fazer as alterações na subclasse - as superclasses herdam as alterações.
- c) É aconselhável utilizar o modificador de acesso public quando uma superclasse precisar fornecer um método somente para suas subclasses e outras classes do mesmo pacote, mas não para outros clientes.
- d) Não é adequado sobrescrever na subclasse métodos da superclasse para fornecer versões que são mais apropriadas para os objetos da subclasse.
- e) Quando um programa cria um objeto da superclasse, o construtor da superclasse imediatamente chama o construtor da subclasse (explicitamente, via super ou implicitamente).

Exercício 12

Sobre orientação a objetos é correto afirmar:

- a) Na hierarquia de classes, se superclasse é uma generalização de subclasses, pode-se inferir que a subclasse é uma especialização de superclasse.
- b) Numa árvore genealógica de classes, a classe mais baixa herda os atributos e métodos somente da superclasse no nível imediatamente acima.
- c) As variáveis de uma classe só podem ser alteradas por métodos definidos nos seus objetos.
- d) O polimorfismo se caracteriza quando, para mensagens distintas, objetos diferentes responderem ou agirem de forma idêntica.
- e) Os objetos de uma classe são idênticos no que se refere à sua interface e ao seu estado.

Exercício 13

Na programação orientada a objetos, temos conceitos como Herança e Polimorfismo. Sobre esses conceitos analise as assertivas e assinale a alternativa que aponta a(s) correta(s).

- I. Para evitar código redundante, o paradigma de orientação a objetos oferece uma estrutura hierárquica e modular para reutilização de código através de uma técnica conhecida como herança.
- II. Herança permite projetar classes genéricas que podem ser especializadas em classes mais particulares, onde as classes especializadas reutilizam o código das mais genéricas.
- III. Literalmente, polimorfismo significa muitas formas. No contexto e projeto orientado a objetos, entretanto, refere-se à habilidade de uma variável de objeto de assumir formas diferentes.
- IV. Polimorfismo permite que os atributos de uma classe não tenham acesso diretamente.

- a) Apenas I.
- b) Apenas I e III.
- c) Apenas I, II e III.
- d) Apenas II, III e IV.
- e) I, II, III e IV.

PROGRAMAÇÃO ORIENTADA A OBJETOS

LISTA DE EXERCÍCIOS 4 – PILARES DA POO SEM GABARITO

WALTER TRAVASSOS SARINHO

@WALTEROPROFESSOR

WALTER.TRAVASSOS@UNIFE.EDU.BR