UNIVERSIDADE FEDERAL DE UBERLÂNDIA

Faculdade de Computação

Av. João Naves de Ávila, nº 2121, Bloco 1A - Bairro Santa Mônica, Uberlândia-MG, CEP 38400-902 Telefone: (34) 3239-4144 - http://www.portal.facom.ufu.br/ facom@ufu.br

No dia 01 de novembro de 2023, fora aplicada prova teórica relacionada ao conteúdo programático do componente curricular GAC058 - Programação Orientada a Objetos 1, presencialmente, na sala 1A420 do Campus de Monte Carmelo em atendimento ao edital **EDITAL FACOM Nº 20/2023** que possui como objeto: **Exame de suficiência com vistas ao atendimento da solicitação apresentada pelo discente Claudionei Pereira da Cunha Filho - Matrícula 31811ECA003 que, nos termos previstos nas Normas da Graduação da Universidade Federal de Uberlândia, requereu abreviação do tempo de conclusão do Curso de Graduação em Engenharia de Agrimensura e Cartográfica da Universidade Federal de Uberlândia – UFU**. A prova iniciou às 13h, com término às 14h05. Registra-se que da correção da aludida prova teórica resultou em 45 pontos dos 100 possíveis. Portanto, conforme se preconiza o § Único, do Art. 200, das Normas Gerais de Graduação da Universidade Federal de Uberlândia, o candidato deve ser considerado **REPROVADO**.

Abaixo segue o espelho da correção da avaliação supracitada.

Correção Prova - Claudionei Pereira da Cunha Filho - Programação Orientada a Objetos 1			
	Gabarito	**Resposta**	**Quantidade de Acertos**
Questão 1	C	A	**0**
Questão 2	A	A	**1**
Questão 3	C	E	**0**
Questão 4	C	C	**1**
Questão 5	D	A	**0**
Questão 6	D	C	**0**
Questão 7	A	A	**1**
Questão 8	E	E	**1**
Questão 9	A	C	**0**
Questão 10	B	B	**1**
Questão 11	C	D	**0**
Questão 12	A	D	**0**
Questão 13	E	A	**0**
Questão 14	D	D	**1**
Questão 15	B	B	**1**

Questão 16	C	D	**0**
Questão 17	D	A	**0**
Questão 18	A	C	**0**
Questão 19	B	B	**1**
Questão 20	B	B	**1**
Nota Final:	**45**		
Status:	**Reprovado**		

Banca Examinadora do Componente Curricular GAC058 - Programação Orientada a Objetos 1

Profª Drª Elaine Ribeiro de Faria Paiva (presidente)

Prof. Dr. Diego Nunes Molinos

Prof. Dr. Fabiano Azevedo Dorça

Prof. Dr. Marcelo Zanchetta do Nascimento (suplente)

UNIVERSIDADE FEDERAL DE UBERLÂNDIA
Disciplina: Programação Orientada a Objetos 1
Curso: Sistemas de Informação
Professora: Elaine Ribeiro de Faria Data: 01.11.2023
1ª avaliação

Aluno: [illegible]

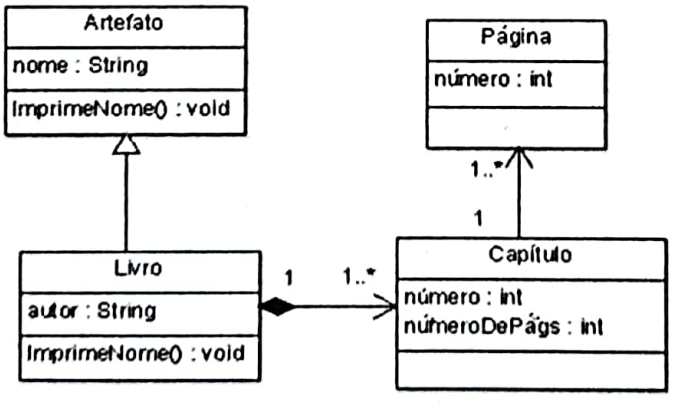
Matrícula: 3/811ECA003 Valor: 100 pts Nota: _____

INSTRUÇÕES PARA A REALIZAÇÃO DESTA PROVA

I- Esta atividade é individual e deve ser resolvida sem consulta.
II- A leitura e interpretação das questões fazem parte da avaliação.
III- O professor não fará nenhum tipo de atendimento durante a prova.
IV- Qualquer tentativa de consulta ao material resultará no recolhimento da prova e atribuição de Nota Zero (0) ao aluno.
V- É proibido o uso de telefone celular durante avaliação (Desligue-o!). Caso o aluno atenda o celular durante a prova, automaticamente sua nota será zero.
VI- As questões podem ser resolvidas à caneta.
VII- Só será permitida a saída do aluno da sala após 30 minutos do início da avaliação.
VIII- Se a letra do aluno estiver ilegível, será atribuída nota zero à questão.
IX- Não é permitido o uso de calculadora ou celular durante a prova.
X- As questões podem ser resolvidas fora de ordem. Numere suas respostas.

Questões

1- (Poscomp 2017) De acordo com o diagrama de classes UML a seguir, assinale a alternativa que se relaciona diretamente com o conceito de polimorfismo da programação orientada a objetos. (5 pontos)



A. A relação entre as classes "Livro" e "Capítulo".
B. Os atributos "número: int" e "númeroDePágs: int" da classe "Capítulo".
C. O método "ImprimeNome" das classes "Artefato" e "Livro".
D. O atributo "autor: String" da classe "Livro".
E. A relação entre as classes "Capítulo" e "Página".

2- (Poscomp 2015) Considere o seguinte código desenvolvido em Java. (5 pontos)

```
public class Animal {
     int numeroPatas;
     public  void fale (){};
}
public class Cao extends Animal {
    public void fale() {
        System.out.println ("au au");
     }
}
public class Gato extends Animal {
    public void fale() {
        System.out.println ("miau");
     }
}
public class GatoPersa extends Gato {
    public void fale() {
        System.out.println ("miauuuu");
     }
}
public class Tigre extends Gato {
public void fale() {
    super.fale();
    System.out.println ("rrrrrr");
 }
}
public class Principal {
    public static void main(String[] args) {
    Gato gato = new GatoPersa();
    gato.fale();
    Cao cao = new Cao();
    cao.fale();
    Tigre tigre = new Tigre();
    tigre.fale();
    }
}
```

Ao executar o código, a saída impressa no console é:

A. miauuuu
au au
miau
rrrrrr

B. miauuuu
au au
rrrrrr

C. miau
au au
miau
miau

D. miau
au au
rrrrrr

E. miau
au au
miau
rrrrrr

3- (Poscomp 2014) Considere as classes Java, que pertencem ao mesmo pacote, a seguir. (5 pontos)

```
abstract public class C1 {
  abstract public Object cria();
  public void mostra(){
    System.out.print("Poscomp 2014");
  }
}
```

```
public class C2 extends C1 {
  static int i = 0;
  Integer j;
  public Object cria() {
      i++;
      j = new Integer(i);
      return j;
  }
  public void mostra() {
      System.out.print("j=" + j);
  }
}
```

```
public class C3 extends C1 {
  double d=3.14;
  Float f;
  public Object cria() {
    d = d + 1.0;
    f = new Float(d);
    return f;
  }
  public void mostra() {
    System.out.print("f="+f);
  }
}
```

```
public class PosComp2014 {
  public static void main(String[] z) {
    C1 a,b,c;
    Object o1,o2,o3;
    a = new C2();
    b = new C2();
    c = new C3();
    o1 = a.cria();
    o1 = a.cria();
    o2 = b.cria();
    o3 = c.cria();
    o3 = c.cria();
    a.mostra();
    System.out.print("  ");
    b.mostra();
    System.out.print("  ");
    c.mostra();
    System.out.print("   " + o1);
    System.out.print("   " + o2);
    System.out.print("   " + o3);
  }
}
```

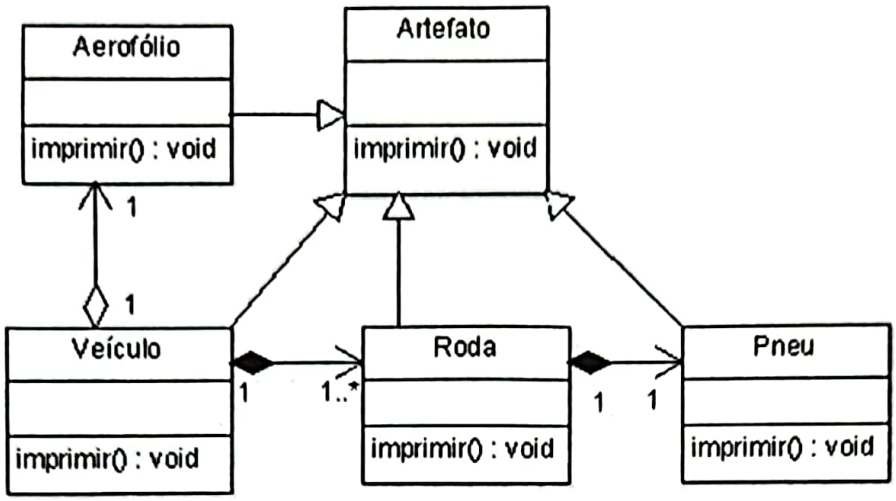
Assinale a alternativa que apresenta, corretamente, os valores impressos pela execução desse programa.

A. O programa está sintaticamente incorreto, ~~não sendo possível executá-lo.~~
B. j=2 j=1 f=5.14 2 1 5.14
C. j=2 j=3 f=5.14 2 3 5.14
D. Poscomp2014 Poscomp2014 Poscomp2014 2 1 5.14
E. Poscomp2014 Poscomp2014 Poscomp2014 2 3 5.14

4- Qual dos seguintes conceitos descreve a capacidade de um objeto de se comportar de várias maneiras diferentes, dependendo do contexto em que é usado? (5 pontos)

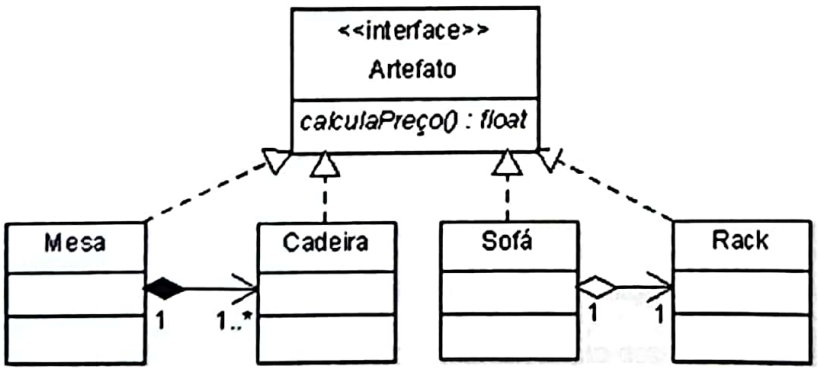
A. Encapsulamento.
B. Herança.
C. Polimorfismo.
D. Interfaces.
E. Abstração

5- (Poscomp 2018) De acordo com o diagrama de classes UML a seguir, assinale a alternativa correta. (5 pontos)



A. O relacionamento entre "Veículo" e "Roda" se relaciona diretamente com polimorfismo.
B. O relacionamento entre "Roda" e "Pneu" se relaciona diretamente com polimorfismo.
C. O relacionamento entre "Veículo" e "Pneu" se relaciona diretamente com polimorfismo.
D. O relacionamento entre "Artefato" e "Veículo" se relaciona diretamente com polimorfismo.
E. O relacionamento entre "Veículo" e "Aerofólio" se relaciona diretamente com polimorfismo.

6- (Poscomp 2022) De acordo com o diagrama de classes UML a seguir assinale a alternativa correta. (5 pontos)



A. A classe "Sofá" tem uma relação de polimorfismo com a classe "Rack".
B. A classe "Sofá" tem uma relação de composição com a classe "Rack".
C. A classe "Cadeira" é herdada da classe "Mesa".
D. Todas as classes devem implementar o método "float calculaPreço()" obrigatoriamente.
E. Quando uma instância da classe "Mesa" é apagada, a(s) instância(s) que existir(em) de classe "Cadeira" permanece(m).

7- (Poscomp 2012) O encapsulamento dos dados tem como objetivo ocultar os detalhes da implementação de um determinado módulo. Em linguagens orientadas a objeto, o ocultamento de informação é tornado explícito requerendo-se que todos os métodos e atributos em uma classe tenham um nível particular de visibilidade com relação às suas subclasses e às classes clientes. Em relação aos atributos de visibilidade, assinale a alternativa correta. (5 pontos)

A. Um atributo ou método público é visível a qualquer classe cliente e subclasse da classe a que ele pertence.
B. Um atributo ou método protegido é visível somente à classe a que ele pertence, mas não às suas subclasses ou aos seus clientes.
C. Um atributo ou método privado é visível somente às subclasses da classe a que ele pertence.
D. Um método protegido não pode acessar os atributos privados declarados na classe a que ele pertence, sendo necessária a chamada de outro método privado da classe.
E. Um método público pode acessar somente atributos públicos declarados na classe a que ele pertence.

8- (Poscomp 2008) Analise as seguintes afirmativas. (5 pontos)

I. Ocultar dados dentro das classes e torná-los disponíveis apenas por meio de métodos é uma técnica muito usada em programas orientados a objetos e é chamada de sobrescrita de atributos.
II. Uma subclasse pode implementar novamente métodos que foram herdados de uma superclasse. Chamamos isso de sobrecarga de métodos.
III. Em Java não existe Herança múltipla como em C++. A única maneira se se obter algo parecido é via interfaces.

A análise permite concluir que

A. apenas a afirmativa I está incorreta.
B. apenas a afirmativa II está incorreta.
C. apenas a afirmativa III está incorreta.
D. apenas as afirmativas I e III estão incorretas.
E. apenas as afirmativas I e II estão incorretas

9- (Poscomp 2007) Analise as seguintes afirmativas. (5 pontos)

I. Encapsulamento é a capacidade de uma operação atuar de modos diversos em classes diferentes.
II. Polimorfismo é o compartilhamento de atributos e métodos entre classes com base em um relacionamento hierárquico.
III. Herança consiste no processo de ocultação dos detalhes internos de implementação de um objeto.
IV. Sobreposição é a redefinição das funções de um método herdado. Os métodos apresentam assinaturas iguais.
V. Em JAVA, todos os métodos numa classe abstrata devem ser declarados como abstratos.

A partir da análise, pode-se concluir que

A. Apenas a afirmativa IV está correta.
B. Apenas as afirmativas III e IV estão corretas.
C. Apenas as afirmativas I, IV e V estão corretas.
D. Apenas as afirmativas I, III e V estão corretas.
E. Todas as afirmativas são falsas.

10- Considere o seguinte código em Java que recebe um valor (variável x) pela linha de comando. Qual das alternativas representa o que será impresso na tela, caso o programa seja executado com a entrada 1. (5 pontos)

```
import java.io.EOFException;
import java.io.IOException;

class Test {
  public static void main(String[] args) {
    int x = Integer.parseInt(args[0]);
    try {
      primeiro(x);
      System.out.println("Após primeiro");
    } catch (IllegalArgumentException e) {
      System.out.println("trata primeiro");
    }
    System.out.println("saiu primeiro");
  }
  static void primeiro(int x)
      throws IllegalArgumentException {
    try {
      segundo(x);
      System.out.println("depois segundo");
    } catch (IOException e) {
      System.out.println("trata segundo");
    }
    System.out.println("saiu segundo");
  }
  static void segundo(int x)
      throws IllegalArgumentException,
             IOException {
    try {
      switch (x) {
        case 0:
          throw new IllegalArgumentException();
        case 1:
          throw new IOException();
        case 2:
          throw new EOFException();
      }
      System.out.println("depois switch");
    } catch (EOFException e) {
      System.out.println("trata terceiro");
    }
    System.out.println("saiu terceiro");
  }
}
```

A. trata primeiro
saiu primeiro
trata segundo
saiu segundo
Após primeiro
saiu primeiro

C. trata terceiro
saiu terceiro
saiu segundo
saiu primeiro

D. trata segundo
saiu segundo
Após primeiro
trata primeiro
saiu primeiro

E. Nenhuma da anteriores

11- (Concurso IF-MT 2023- Adaptada) - Considere as afirmações relacionadas ao código fonte a seguir: (5 pontos)

Conta.java

```
1 public class Conta {
2     public double saldo;
3
4     public Conta(double saldoInicial) {
5         saldo = saldoInicial;
6     }
7     public double getSaldo() {
8         return this.saldo;
9     }
10
11    public void setSaldo(double valor) {
12        this.saldo = valor;
13    }
14
15    public void deposito(double deposito) {
16        this.saldo = saldo + deposito;
17    }
18 }
```

ContaEspecial.java

```
1  public final class ContaEspecial extends Conta{
2      private double limite;
3
4      public ContaEspecial(double saldoInicial) {
5          super(saldoInicial);
6          this.limite=0;
7      }
8
9      public void setLimite(double valor){
10         this.limite = valor;
11     }
12
13     public double getLimite(){
14         return this.limite;
15     }
16     @Override
17     public double getSaldo(){
18         return this.limite+this.saldo;
19     }
20 }
```

AbreConta.java

```
1 public class AbreConta {
2     public static void main(String[] args) {
3         ContaEspecial conta = new ContaEspecial(1000);
4         System.out.println("Saldo atual: " + conta.getSaldo());
5     }
6 }
```

I. Classe *ContaEspecial* pode ser herdada por outras classes.
II. Existe uma violação de encapsulamento.
III. O método getSaldo é herdado da classe mãe, mas foi sobrescrito na classe filha.

Está **CORRETO** o que se afirma em:

A. Nenhuma das afirmações é verdadeira.
B. I e II, apenas.
C. **II e III, apenas**
D. I e III, apenas.
E. I,II e III.

12- (Concurso FUNDATEC - PROCERGS - Analista - Área: Ciência de Dados - 2023) - O(A) _____ de um método é o recurso por meio do qual uma classe derivada reescreve o método da classe-base a fim de atender a alguma particularidade. (5 pontos)

A. Sobreposição
B. Coesão
C. Acoplamentos
D. Sobrecarga
E. Herança Múltipla

13- (Concurso UFSC - 2023) - Considere as seguintes definições relacionadas à programação orientada a objetos, com lacunas a preencher, e assinale a alternativa que preencha corretamente as três definições, considerando sua ordem. (5 pontos)

1. _____ é a capacidade de objetos de classes distintas responderem a uma mesma chamada de método de maneiras diferentes. Isso permite que as subclasses redefinam o comportamento de métodos herdados da classe base.

2. _____ é a capacidade de um objeto ter vários métodos com o mesmo identificador, mas com assinaturas de métodos diferentes. Isso permite que os objetos respondam a chamadas de métodos distintos, mas com identificadores idênticos, com base na quantidade e no tipo de argumentos fornecidos.

3. _____ é a capacidade de uma subclasse substituir o comportamento de um método herdado da classe base. Isso permite que uma classe modifique o comportamento de um método para atender às suas próprias necessidades, mantendo a mesma assinatura de método.

A. Sobrecarga – Polimorfismo – Herança
B. Sobrescrita – Polimorfismo – Encapsulamento
C. Polimorfismo – Sobrecarga – Herança
D. Herança – Encapsulamento – Sobrescrita
E. Polimorfismo – Sobrecarga – Sobrescrita

14- (Concurso - FAU UNICENTRO - UNIOESTE - Analista de Informática - Área Desenvolvimento de Sistemas - 2023) O que é um objeto em programação orientada a objetos? (5 pontos)

A. Um princípio que promove a reutilização de código por meio da herança.
B. Uma técnica que permite ocultar os detalhes internos de uma classe e expor apenas a interface necessária.
C. Um recurso que permite que um objeto seja representado por várias classes.
D. Uma instância de uma classe que possui atributos e comportamentos específicos.
E. Um conjunto de instruções que realiza uma tarefa específica em um programa.

15- (Concurso - IBFC - MGS - Arquiteto Java - 2023) Abstração em programação orientada a objetos, é um dos conceitos mais importantes, assinale a alternativa correta sobre sua definição. (5 pontos)

A. Extrair conceitos de programação
B. Esconder detalhes de algo, ou seja, detalhes desnecessários
C. Auxiliar na criação de componentes visuais
D. Compilar as classes do programa
E. Criar classes que estendem outras classes

16- (Sanaza - Campinas) Considere que um Analista de TI sabe que uma classe Pessoa Física e uma classe Pessoa Jurídica possuem o atributo nome como uma informação em comum e que o CPF é um atributo específico para a Pessoa Física e o CNPJ é um atributo específico para Pessoa Jurídica. Então o Analista criou uma outra classe com o atributo nome e seu objetivo é que haja herança deste e, eventualmente, outros métodos e atributos, para as classes filhas, Pessoa Física e Pessoa Jurídica, que já existiam. (5 pontos)

Essa classe criada não é instanciada, apenas fornece um modelo para geração de outras classes; e é denominada.

A. Subclasse.
B. Classe construtora.
C. Classe abstrata.
D. Classe sobrescrita.
E. Pacote.

17- (IF-PA 2019) Quanto aos conceitos do paradigma da orientação a objetos, é CORRETO afirmar: (5 pontos)

A. por meio do conceito de Polimorfismo, é possível a definição de vários métodos ou funções com o mesmo nome, porém com diferentes assinaturas. Essa característica do conceito de Polimorfismo é denominada de Delegação.
B. por meio do conceito de Herança, uma subclasse é capaz de reutilizar os métodos e atributos de uma superclasse, desde que esses métodos e atributos estejam encapsulados, ou seja, suas visibilidades estejam como "private".
C. uma classe definida como Abstrata, é uma classe que define os seus atributos e métodos para que sejam herdados por uma outra classe que irá implementar os seus métodos. Em uma classe Abstrata não é possível a implementação dos seus métodos, somente os seus cabeçalhos.

D. em uma Interface definimos comportamentos (métodos) sem os implementar. Por meio da Interface podemos definir o que um objeto obrigatoriamente deve fazer e não como ele faz.
E. uma classe que implementa uma classe Abstrata, deverá obrigatoriamente redefinir os métodos e atributos que herdou. A classe que implementa a classe Abstrata não pode definir seus próprios atributos.

18- (UFRJ - 2018) Com relação aos conceitos de orientação objeto, existe uma característica que faz com que detalhes internos do funcionamento dos métodos de uma classe permaneçam ocultos para os objetos e que por conta dessa técnica, o conhecimento a respeito da implementação interna da classe é desnecessário do ponto de vista do objeto, uma vez que isso passa a ser responsabilidade dos métodos internos da classe. A característica apresentada se refere a: (5 pontos)

A. encapsulamento.
B. polimorfismo.
C. abstração.
D. herança.
E. namespaces.

19- (UNESPAR 2023) Com relação aos conceitos de Programação Orientada à Objetos é correto afirmar que: (5 pontos)

A. Para uma classe ser considerada abstrata, todos os seus métodos devem ser abstratos. Em Java, para se definir uma classe abstrata deve-se utilizar a palavra chave abstract no início de sua declaração.
B. Polimorfismo é o princípio pelo qual duas ou mais classes derivadas de uma mesma superclasse podem invocar métodos que têm a mesma identificação (assinatura) mas comportamentos distintos, especializados para cada classe derivada, usando para tanto uma referência a um objeto do tipo da superclasse.
C. O encapsulamento é alcançado por meio da definição da visibilidade pública aos atributos e métodos.
D. Interface pode ser considerada como a forma com que um objeto se apresenta para outros, no que diz respeito aos seus atributos e métodos. Em Java, uma mesma classe não pode implementar mais de uma interface.
E. A herança é um mecanismo que permite que uma classe herde todo o comportamento e os atributos de outra classe. Em Java, pode-se implementar tanto a herança única quanto a herança múltipla

20- (FUNDATEC 2023) Em linguagens orientadas a objetos, como Java, trabalha-se com classes e modificadores de visibilidade que determinam como deve ocorrer o acesso a determinadas partes da classe. Os modificadores de visibilidade são denotados em UML por um nome e um símbolo. Assinale a alternativa correta sobre modificadores de visibilidade. (5 pontos)

A. Público (–), Privado (+) e Protegido (*).
B. Público (+), Privado (–) e Protegido (#).
C. Público (#), Privado (*) e Protegido (+).
D. Público (*), Privado (#) e Protegido (–).
E. Público (+), Privado (–) e Protegido (+).