
Avaliação 02

1. (1.0) Responda as seguintes questões quanto a modelagem de sistemas:

a. O que é Modelagem de Sistemas? Por que é importante?

R.: A Modelagem de Sistema consiste em criar um modelo/representação que pode ser através de figuras gráficas ou texto corrido do que como será estruturado o Sistema que se está trabalhando. É importante, porque faz a documentação para usuários e desenvolvedores que não participaram do período da criação, ajuda a identificar problemas e necessidades dos projetos, a estipular custos de tempo e recursos, entre outras coisas.

b. O que é UML?

R.: A sigla significa *Unified Modeling Language* (UML), ou seja, Linguagem de Modelagem Unificada. É uma linguagem gráfica para modelar/descrever o projeto - que pode ser um site, um programa, um aplicativo, entre outros - a partir de usos de diagramas que abordam diferentes perspectivas de um mesmo projeto. Os diagramas são divididos em duas grandes categorias: Estrutural e Comportamental.

c. Cite dois tipos de diagramas de Estrutura, Comportamento e Interação da UML.

R.: Diagrama de Estrutura: entre essa categoria, temos o Diagrama de Classe, esse diagrama tem como objetivo definir a estrutura de classe utilizados pelo sistema, e Diagrama de Objetos, que é praticamente um Diagrama de Classes, mas fornece uma visão dos valores armazenados em determinado momento da execução.

Diagramas de Comportamento: temos o Diagrama de Caso de Uso, que é voltado para a apresentação das características de um sistema, e o Diagrama de Atividade, que contempla diversas tarefas desempenhadas na execução de uma atividade.

Diagrama de Interação: temos o Diagrama de Sequência, que demonstra a interações entre objetos na execução de uma operação, e Diagrama de Tempo, é um tipo específico de Diagrama de Sequência, descrevendo mudanças de estado e interações entre objetos dentro do intervalo de tempo.

2. (1.0) Declare (e instancie) vetores em Java preparados para armazenar:

a. Número de dias em cada um dos meses do ano.

R.:

b. Temperatura mínima e máxima de cada dia de uma semana;

R.:

c. As três notas de cada aluno de uma turma com 40 matriculados.

R.:

3. (2.0) Defina os seguintes conceitos:

a. Sobrecarga

R.: Sobrecarga é também conhecida como Polimorfismo Estático. Nele você cria métodos com mesmo nome, sobrecarregando o “nome” (colocando vários métodos com o mesmo nome), mas com parâmetros diferentes, ou seja, pode de tipos, ordem e quantidades diferentes.

b. Polimorfismo

R.: O Polimorfismo (poli: múltiplas; morfismos: formas) são formas diferentes de fazer a mesma coisa. Com esse princípio temos que a partir de uma classe as classes que derivar delas são capazes de invocar os métodos que, embora apresentem a mesma assinatura, comportam-se de maneira diferente para cada uma das classes derivadas.

c. Herança

R.: A Herança é a forma de receber comportamentos (métodos) e características (atributos) da classe que “dá” origem. É uma forma de reaproveitar o código, além de torna-lo mais organizado. A classes que são origem de outra é chamada de classe pai (ou super-classe) e já as que foram originadas é chamada de classe filha (sub-classe). A relação entre esses dois tipos é que a sub-classe É UMA super-classe, mas mais especializada. Enquanto a super-classe é mais genérica.

d. Classe abstrata

R.: A abstração consiste em simplificação. Seria selecionar o conceito do mundo real e simplificar o conceito dentro do software. A partir desse conceito, é mais fácil conceituar as Classes abstratas que não permitem fazer qualquer tipo de instância. São classes feitas para serem modelos para classes que serão originadas. As classes originadas deverão subescrever os métodos para ser possível sua implementação e são chamadas de classes concretas.

4. (2.0) Faça um programa que leia e armazene as alturas de grupo de 20 pessoas, calcule a média e indique quantas pessoas estão acima dessa média.

R.:

5. (4.0) Implemente o seguinte conjunto de classes usando herança. Codifique um programa que leia os dados de 5 funcionários, armazene em vetor e exiba seus dados usando polimorfismo. Observe que a classe funcionário é abstrata e o pagamento do funcionário é calculado nas subclasses concretas: Programador (possui salário mensal) e Instrutor (recebe de acordo com as horas trabalhadas).

