



Fundação CECIERJ - Vice Presidência de Educação Superior a Distância

**Curso de Tecnologia em Sistemas de Computação**  
**Disciplina de Arquitetura e Projeto de Sistemas II**  
**Gabarito da AP3 – 2º semestre de 2014**

**Observações:**

1. Prova sem consulta e sem uso de máquina de calcular.
2. Use caneta para preencher o seu nome e assinar nas folhas de respostas.
3. Você pode usar lápis para responder as questões.
4. Ao final da prova devolva as folhas de questões e as de respostas.
5. Todas as respostas devem ser transcritas nas folhas de respostas. As respostas nas folhas de questões não serão corrigidas.

Questão 1 (2,5 pontos)

Relacione cada elemento da coluna da esquerda com um e somente um elemento na coluna da direita.

- |                                      |   |
|--------------------------------------|---|
| (a) diagrama de classes              | (1) Detalha o comportamento de um objeto no decorrer da sua vida.   |
| (b) diagrama de casos de uso         | (2) Explicita a estrutura estática interna do sistema.  |
| (c) diagrama de transição de estados | (3) Explicita as possibilidades de interação entre os usuários e o sistema.                               |
| (d) diagrama de sequência            | (4) Explicita a eficiência do sistema em termos de utilização de recursos.                                |
| (e) descrição de casos de uso        | (5) Detalha a implantação do sistema em ambiente de uso.  |
|                                      | (6) Detalha a interação entre diferentes objetos do sistema para atender a uma funcionalidade específica. |
|                                      | (7) Detalha uma determinada possibilidade de interação entre o usuário e o sistema.                       |
|                                      | (8) Explicita como o estado dos objetos é persistido em banco de dados.                                   |

**Resposta: a → 2; b → 3; c → 1; d → 6; e → 7**

Questão 2 (2,5 pontos)

Relacione cada elemento da coluna da esquerda com um e somente um elemento da coluna da direita.

- |   |  |
|---|--|
| (a) Interface com estados ilegais           | (1) Falta de comportamento que possibilite transição de estado válida  |
| (b) Interface com comportamento perigoso    | (2) Não possibilita alcançar todos os estados válidos do espaço-estado |
| (c) Interface com comportamento irrelevante | (3) Exibe métodos privados como públicos                               |



- (c) Assumindo que o espaço-estado do atributo “quantia” da classe “Pagamento” seja de R\$ 0 a R\$ 1000,00, é permitido que a classe “PagamentoCheque” modifique esse espaço-estado para de R\$ 50,00 a R\$ 500,00? Justifique a sua resposta.

Resposta:

- (a) Grau de dependência direto para a classe Pagamento: 1 (classe Pedido).
- (b) Grau de dependência indireto para a classe ItemEstoque: 3 (classes Produto, Especificação e ItemPedido).
- (c) Sim. Tendo em vista que todo PagamentoCheque é um Pagamento, e que a variação de espaço-estado em casos de herança deve ser mais restritiva nas subclasses, a modificação proposta é válida. Ou seja, o intervalo entre 50 e 500 está dentro do intervalo entre 0 e 1000, como era de se esperar pois o PagamentoCheque é um Pagamento.

#### Questão 4 (2,0 pontos)

Em relação à orientação a objetos, defina e exemplifique:

- (a) Classe
- (b) Objeto
- (c) Herança
- (d) Polimorfismo

Resposta:

Classe: representação computacional de entidades ou processos do mundo real. São compostas de atributos (características – informações) e métodos (comportamentos – processos) e instanciam objetos. Exemplo: classe Gerente, com atributos nome e idade e métodos calculaSalario() e getIdade()

Objeto: instanciamento de uma classe. Possui um conjunto de serviços (interface) e sua implementação (estruturas de dados – atributos, e implementação de operações – métodos). Exemplo: objeto da classe Gerente com nome = “João” e idade = 25.

Herança: mecanismo que promove a reutilização de software por meio do reconhecimento da similaridade entre classes de objetos, formando uma hierarquia. Define uma relação do tipo “é um”, onde uma classe compartilha a estrutura e o comportamento definidos em uma ou mais classes. Exemplo: classe Gerente herda da classe Funcionario.

Polimorfismo: propriedade derivada do fato de que objetos de diferentes classes podem reagir a uma mesma mensagem de forma diferente. Dessa forma, cada classe implementa um método específico para uma operação, possibilitando a definição de protocolos comuns. Exemplo: método `getSalario` em `Gerente` leva em consideração a idade do gerente, enquanto em `funcionário` somente o tempo de trabalho é considerado.