



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 AP1 – 1º semestre de 2011

Nome –

Assinatura –

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 questões e 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 pontos)

Responda, com suas palavras, às seguintes questões:

- (a) Quais são os tipos de domínio possíveis para as classes em um sistema OO? Detalhe as suas principais características.

Resposta: Os tipos de domínio possíveis para classes em são: base, arquitetura, negócio e aplicação. O domínio de base contém as classes fundamentais do sistema e já vem nas bibliotecas das linguagens de programação. O domínio de arquitetura contém classes para abstrair elementos de hardware ou software necessários ao desenvolvimento de sistemas. Normalmente, essas classes também são incluídas nas linguagens de programação. O domínio de negócio contém classes específicas para uma área de conhecimento. É raro encontrar bibliotecas disponíveis com esses tipos de classes. Por fim, o domínio de aplicação contém classes híbridas, com dependências para diferentes domínios, que servem para viabilizar a construção de sistemas específicos sem fazer com que as classes de outros domínios aumentem os seus acoplamentos.

- (b) Qual a diferença entre interface com comportamento ilegal e interface com comportamento incompleto?

Resposta: Esses dois problemas citados são opostos no sentido do que oferecem ao usuário da classe em questão. A interface com comportamento ilegal fornece algum método que possibilita uma troca de estados irregular na classe. Por outro lado, a

interface com comportamento incompleto deixa de fornecer um método necessário, não permitindo que uma troca de estados válida ocorra.

Questão 2 (3 pontos)

Para cada uma das afirmativas a seguir, indique se ela é correta ou incorreta e justifique a resposta com as suas palavras.

- (a) “Em um sistema com interface com o usuário, as classes de modelo devem depender das classes de interface”.

Resposta: Incorreta. Para facilitar a reutilização das classes de modelo e eventual substituição da interface, elas não devem conter dependência para nenhuma classe de interface.

- (b) “Minimize o número de mensagens enviadas entre uma classe e as suas classes colaboradoras”.

Resposta: Correta. Um número excessivo de mensagens entre classes pode indicar que o encapsulamento das classes está sendo quebrado, e que métodos com mais responsabilidades são necessários.

- (c) “Para aumentar a encapsulamento do sistema, sempre faça uso de atributos protegidos”.

Resposta: Incorreta. Para que o encapsulamento seja aumentado, é necessário fazer uso de atributos privados, e fornecer métodos de acesso a esses atributos.

Questão 3 (2 pontos)

Com relação ao desenvolvimento de software em geral:

- (a) Qual a importância da modelagem para o desenvolvimento de software?

Resposta: Modelos são úteis para o entendimento de problemas, a difusão do conhecimento entre os desenvolvedores e a possibilidade de testar hipóteses antes de realizá-las. Eles são criados devido à limitação do ser humano em lidar com a complexidade.

- (b) Como o desenvolvimento orientado a objetos se difere do desenvolvimento tradicional?

Resposta: O desenvolvimento orientado a objetos se caracteriza por ser centrado em dados e processos, diferentemente do desenvolvimento tradicional que é centrado em processos ou dados de forma excludente. É também caracterizado por um

desenvolvimento interativo incremental, diferentemente do desenvolvimento tradicional linear-sequencial.

Questão 4 (3 pontos)

Sobre Casos de Uso, responda com suas palavras:

(a) Qual é a diferença entre Caso de Uso Essencial e Concreto?

Resposta: Um caso de uso essencial apresenta a intenção do usuário e as responsabilidades do sistema. Um caso de uso concreto detalha as decisões sobre interface ou tecnologia utilizada para a realização de ações.

(b) O que são atores? Cite três exemplos.

Resposta: Atores representam qualquer elemento externo que possa interagir direta ou indiretamente com o sistema. Exemplos de atores são: outros sistemas, seres humanos e dispositivos de hardware.

(c) Que tipos de relações podem existir entre Casos de Uso? Cite duas delas.

Resposta: Relações de inclusão, em que casos de uso podem compartilhar um mesmo comportamento. Relações de Generalização/Extensão que indicam comportamentos opcional/alternativo.