



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 – 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)

Em relação à orientação a objetos, defina e exemplifique:

(a) Classe

Resposta: 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().

(b) Objeto

Resposta: instancição 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.

(c) Herança

Resposta: 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.

(d) Polimorfismo

Resposta: 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.

Questão 2 (3 pontos)

Em relação a componentes,

(a) Quais são as suas principais características?

Resposta: As principais características de componentes são: serem unidades de composição auto-contidas (encapsulamento), terem interfaces bem definidas (contrato), terem grau de maturidade e documentação (qualidade), serem especificados em diferentes níveis (abstração) e obedecerem a restrições (arquitetura/plataforma).

(b) Quais as desvantagens de utilizar interfaces muito específicas ou muito genéricas?

Resposta: Interfaces muito específicas fazem com que poucas oportunidades de reutilização do componente possam ser identificadas. Por outro lado, com o uso de interfaces muito genéricas, apesar de os componentes se conectarem com facilidade, o protocolo de comunicação pode ser sobrecarregado.

(c) Quais as técnicas existentes para customização?

Resposta: A customização de componentes pode ser via variabilidade ou adaptação. A customização via variabilidade consiste em utilizar mecanismos predefinidos para customização. Esses mecanismos são: geração, parametrização e interface de configuração. Caso não seja possível customizar via variabilidade, alguma técnica de adaptação pode ser adotada. Exemplos de técnicas de adaptação são: copiar e colar, herança e embrulho.

Questão 3 (2 pontos)

Com relação às Arquiteturas Específicas de Domínio (DSSA):

(a) Defina DSSA.

Resposta: “Uma DSSA é uma coleção de componentes de software especializados para um determinado tipo de tarefa (domínio), generalizados para que seja possível seu uso efetivo através do domínio, e compostos em uma estrutura padronizada (topologia) efetiva para a construção de aplicações” (HAYES, 1994) ou “Uma DSSA compreende

um modelo de domínio, requisitos de referência, uma arquitetura de referência, a infraestrutura ou ambiente de suporte e um processo de desenvolvimento.” (TAYLOR, 1995)

(b) Quais são os benefícios de uma DSSA?

Resposta: Alguns dos benefícios de uma DSSA são: (i) a reutilização de requisitos, de projeto, de testes, de pessoal e de documentação; (ii) a redução de defeitos; (iii) a previsão e repetição do processo de desenvolvimento; e (iv) a redução de esforço e risco de desenvolvimento.

Questão 4 (3 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) Quais são as dificuldades da Engenharia de Requisitos?

Resposta: Algumas das dificuldades são: a comunicação com o usuário, mudanças constantes, previsão de situações futuras, entendimento completo do problema e diversas formas de representação, dentre outras.

(c) 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 iterativo incremental, diferentemente do desenvolvimento tradicional linear-sequencial.