



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 AP2 – 2º 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)

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

- |                             |   |
|-----------------------------|---|
| (a) Padrão Proxy            | (1) Define uma dependência de um para muitos com um mecanismo de notificação de eventos.                  |
| (b) Padrão Observer         | (2) Define o esqueleto de um algoritmo e delega alguns dos seus passos para as subclasses.                |
| (c) Padrão State            | (3) Fornece um substituto a um objeto.  |
| (d) Padrão Strategy         | (4) Provê uma interface única para um conjunto de interfaces de um subsistema, facilitando o seu uso.     |
| (e) Padrão Abstract Factory | (5) Converte a interface de uma classe em outra, para atender às expectativas do cliente.                 |
|                             | (6) Define uma família de algoritmos de forma encapsulada.  |
|                             | (7) Fornece uma interface para criação de objetos relacionados sem especificar as suas classes concretas. |
|                             | (8) Permite que um objeto modifique o seu comportamento em função do seu estado interno.                  |

**Resposta: a → 3; b → 1; c → 8; d → 6; e → 7**

Questão 2 (3 pontos)

Em relação a Componentes, responda com as suas palavras:

(a) Quais são as suas principais características?

Resposta: As principais características de componentes são: ser unidade de composição autocontida (encapsulamento), ter interfaces bem definidas (contrato), ter maturidade e documentação (qualidade), ser especificado em diferentes níveis (abstração) e obedecer a restrições de arquitetura/plataforma.

(b) Quais são os tipos de interface existentes e para que servem cada uma delas?

Resposta: As interfaces existentes são três tipos: providas, requeridas e de configuração. As interfaces providas explicitam as funcionalidades que o componente fornece para o mundo exterior. Já as interfaces requeridas explicitam as funcionalidades que o componente demanda do mundo exterior para viabilizar a sua execução. Finalmente, as interfaces de configuração permitem a customização do componente via variabilidade.

(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 à Engenharia de Domínio:

(a) Quais são seus objetivos?

Resposta: Objetivos da ED são: (i) originar meta-sistemas, ou seja, sistemas que são reutilizados na construção de aplicações específicas; (ii) descobrir e definir modelos de domínio e arquiteturas comuns às famílias de aplicações para suportar uma reutilização pré-planejada; e (iii) tornar explícito e formalizar as teorias específicas ao domínio que permitem aos projetistas e especialistas do domínio a raciocinar sobre problemas e sistemas no domínio da aplicação.

(b) Quais etapas estão envolvidas? Descreva brevemente cada uma delas.

Resposta: As etapas envolvidas são:

- **Análise de Domínio:** onde o conhecimento existente sobre o domínio é estudado e formalizado através de um modelo de domínio;

- Projeto de Domínio: em que arquiteturas de software são construídas para atender aos requisitos identificados no modelo de domínio; e
- Implementação do Domínio: em que artefatos reutilizáveis são implementados para compor as arquiteturas.

Questão 4 (3 pontos)

Com relação aos padrões de software, responda:

(a) Defina padrões de software.

Resposta: Um padrão descreve um problema recorrente em nosso meio e inclui uma solução genérica para o mesmo, de tal maneira que se possa usá-la várias vezes sem necessariamente fazê-lo de forma idêntica.

(b) Como pode ser sua utilização no desenvolvimento de software?

Resposta: Padrões podem complementar processos e métodos de construção de software com guias para a solução de problemas recorrentes. Eles permitem a reutilização de conhecimento sobre o desenvolvimento de software e facilitam a comunicação entre os desenvolvedores.

(c) Qual é sua relação com *frameworks*?

Resposta: Um *framework* pode conter vários padrões. Padrões costumam ser mais genéricos que frameworks. Um *framework* pode ser incorporado em código, mas apenas exemplos de padrões podem ser incorporados em código.