



Fundação CECIERJ - Vice Presidência de Educação Superior a Distância

**Curso de Tecnologia em Sistemas de Computação**

**Disciplina: Engenharia de Software**

**AP2 1º semestre de 2007.**

**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.
- 

- 1) Explique o que você entende pelos termos defeito, erro e falha. Explique a diferença entre eles, e cite pelo menos uma técnica de VV&T que pode ser usada para identificar cada um deles. Dê um exemplo de como defeitos podem ser classificados (valor: 1,5 pontos; máximo: 10 linhas).

Defeito: Deficiência mecânica ou algorítmica que se ativada pode levar a uma falha. Defeitos podem ser classificados como de omissão, inconsistência, fato incorreto, ambigüidade e informação estranha. Técnicas que podem ser utilizadas para identificar defeitos: Inspeção Ad-hoc, checklist ou Técnica de Leitura.

Erro: Item de informação ou estado de execução inconsistente.

Falha: Evento notável onde o sistema viola suas especificações. Técnica que pode ser utilizada para identificar erros: Testes de software

2) Liste as fases do processo de inspeção de software, apresente o objetivo de cada uma e que papéis são necessários para sua execução. (valor: 1,5 pontos; máximo: 15 linhas).

**Fase:** Planejamento

**Responsável:** Moderador.

**Tarefas:** Definir contexto da inspeção (descrição da inspeção, como a preparação individual deverá ocorrer, documento a ser inspecionado, autor do documento, entre outros); Selecionar inspetores (recomenda-se utilizar entre 3 e 5 inspetores em uma inspeção); Distribuir material.

**Fase:** Preparação Individual

**Responsável:** Inspetor.

**Tarefas:** Estudar os artefatos; Fazer anotações sobre os artefatos.

**Fase:** Reunião de Inspeção

**Envolvidos:** Moderador, Inspetores e Autor.

**Tarefas:** Leitura do documento, com a equipe discutindo possíveis defeitos (Duração recomendada 2hrs); Produzir uma lista de defeitos; Em casos de discordância a decisão sobre registrar um defeito ou não (falso positivo) é do moderador.

**Fase:** Retrabalho

**Responsável:** Autor.

**Tarefas:** Corrigir os defeitos encontrados.

**Fase:** Continuação

**Responsável:** Moderador.

**Tarefas:** Analisar correções do autor e inspeção como um todo; Re-avaliar qualidade do artefato inspecionado; Decidir sobre a necessidade de uma nova inspeção.

3) Explique a diferença entre teste funcional e teste estrutural, apresentando pelo menos um exemplo de critério que pode ser utilizado para projetar casos de teste. (valor: 1,5 pontos; máximo: 10 linhas).

**Teste Funcional:** baseia-se na especificação do software para derivar os requisitos de teste. Aborda o software de um ponto de vista macroscópico.

**Critérios:** Particionamento em Classes de Equivalência; Análise do Valor Limite; Grafo de Causa-Efeito.

**Teste Estrutural:** É baseada no conhecimento da estrutura interna da implementação. Teste dos detalhes procedimentais. A maioria dos critérios dessa técnica utiliza uma representação de programa conhecida como grafo de programa ou grafo de fluxo de controle.

**Critérios:** Todos os caminhos, todos os usos, todas as definições, todos os usos computacionais, ...

4) Explique a relação existente entre os processos da área de conhecimento de tempo e da área de conhecimento de escopo (valor: 2,0 pontos; máximo: 10 linhas).

Os processos da área de conhecimento de escopo têm como principal objetivo identificar o trabalho que deve efetivamente ser realizado no decorrer do projeto. Os processos da área de conhecimento de tempo têm como principal objetivo gerar o cronograma do projeto, a partir de uma divisão do trabalho que deve ser realizado em unidades menores e dependentes entre si. Assim, os processos da área de conhecimento de escopo geram a informação que é consumida pelos processos da área de conhecimento de tempo, antecipando a estes nas fases de gerenciamento de projeto.

5) O que é a inversão de comando na codificação de uma rotina? Explique como o uso desta inversão de comando afeta a independência da rotina. (valor: 1,5 pontos; máximo: 10 linhas).

Inversão de comando ocorre quando uma rotina recebe um parâmetro que define como a rotina deve se comportar. O caso mais comum da inversão de comando ocorre quando uma rotina recebe um parâmetro booleano, executando códigos diferentes de acordo com o valor deste parâmetro. Também é comum que uma rotina deste tipo receba um tipo enumerado como parâmetro, cumprindo este mesmo papel.

A inversão de comando torna complexo o projeto da rotina, pois seu comportamento (ou seja, o código executado por ela) depende de seu cliente (ou seja, o trecho de código que chama a rotina).

6) Considerando o Diagrama de Classes abaixo, encontre os valores de NOC (número de filhos), DIT (profundidade de herança) e CBO (acoplamento entre objetos) para as classes identificadas na tabela. Destas quatro classes a que poderia ser mais propensa à falha quando comparada com as outras e por quê? (valor: 2,0 pontos):

CLASSE	NOC	DIT	CBO	Propensa à Falha
Equipamento	3	0	1	
Administrador	0	2	0	quanto maior DIT, maior a probabilidade de identificação de falhas
Periférico	0	1	2	
ComponenteDeHardware	0	0	1	

