



Fundação CECIERJ - Vice Presidência de Educação Superior a Distância

Curso de Tecnologia em Sistemas de Computação

Disciplina: Engenharia de Software

AP2 1º semestre de 2010.

- 1) Porque o uso de bons nomes para variáveis e rotinas facilita o entendimento de um programa? (valor: 2,0 pontos; máximo: 10 linhas)

O código-fonte de um programa de computador é um artefato bastante complexo e difícil de entender. Assim, todos os esforços devem ser feitos para que estes artefatos sejam mais legíveis, de modo a reduzir as chances de um desenvolvedor introduzir um erro no sistema por desconhecimento sobre como realizar uma alteração no código. Comentários, boa organização, simplicidade, boa nomenclatura para os componentes do programa (como classes, métodos e variáveis) são ferramentas que auxiliam na clareza dos programas. Bom nome para rotina ajuda o desenvolvedor a entender seu objetivo, enquanto um bom nome para as variáveis e parâmetros facilita o entendimento da informação ali armazenada no contexto da rotina.

- 2) Quais são as três características fundamentais de um projeto que fazem com que ele seja diferente de uma operação continuada (dica: pense na definição de um projeto e no triângulo que define suas características)? (valor: 2,0 pontos; máximo: 10 linhas)

O projeto é temporário, tendo data de início e encerramento, gera um produto ou serviço único e é realizado de forma progressiva. Por temporário, entende-se que o projeto possui um início e um término definidos (não dura para sempre!). Por único, entende-se que existem aspectos no produto ou serviço sendo criado que o diferenciam de produtos ou serviços anteriores. Por progressivo, entende-se que as características de cada projeto são mais detalhadas na medida em que maior for o entendimento do produto ou do serviço.

- 3) Para as métricas a-e abaixo, indique seu nível de precisão (objetiva, subjetiva), escala (nominal, ordinal, intervalar, razão) e objeto de medição (processo, produto). Apresente 1 exemplo de uso para cada uma delas no contexto do software. (Valor: 2,0 pontos; máximo 10 linhas)

- a. Tempo de Desenvolvimento: objetiva, razão, processo
 - i. Ex.: usada para medir o tempo de uma determinada atividade ou projeto
- b. Nível de Experiência do Desenvolvedor: subjetiva, ordinal, processo
 - i. Ex.: usada para sugerir a experiência de um desenvolvedor que participa das atividades do processo
- c. Número de Defeitos: objetiva, razão, produto ou processo

- i. Ex. processo: usada para medir a quantidade de defeitos identificados numa determinada atividade do processo
 - ii. Ex. produto: usada para medir a quantidade de defeitos num determinado artefato do produto
- d. Perda de Coesão em Métodos: objetiva, razão, produto
 - i. Ex.: usada para medir a quantidade de referências externas existentes em um determinado método de uma classe
- e. Pontos por Função: subjetiva, intervalar, produto
 - i. Ex.: usada para sugerir o esforço/tamanho esperado para um determinado produto

- 4) No método de inspeção de software, para a fase de detecção de defeitos, a forma mais simples de realizar o procedimento de leitura é a partir da utilização de técnica *ad-hoc*. Entretanto, para aumentar a eficiência, inspetores podem usar checklists ou técnicas de leitura. Explique o que é um checklist e uma técnica de leitura e cite pelo menos 1 diferença entre os dois. (Valor: 2,0 pontos; máximo 10 linhas)

Um checklist consiste numa lista de questionamentos, usualmente respondidos com respostas do tipo sim/não, que direcionam a perspectiva a ser usada na inspeção. Inspetores seguem as perguntas do checklist, porém buscam por defeitos aplicando uma abordagem *ad-hoc*, ou seja, o checklist indica as características que devem ser observadas, mas não indicam como aplicá-las ou mesmo como identificar os defeitos. O resultado final da inspeção depende da qualidade do checklist e da experiência do inspetor.

Uma técnica de leitura, por sua vez, representa um conjunto concreto de instruções que permite indicar a um inspetor quais características se deseja inspecionar e como os defeitos podem ser localizados. Neste caso, o resultado da inspeção depende mais da qualidade da técnica do que propriamente da experiência do inspetor. Representam o estado da arte em técnicas de inspeção.

- 5) Alguns critérios podem ser utilizados para projetar casos de teste. Por exemplo, análise do valor limite é um deles, usado para teste funcional. Dê outro exemplo de critério, indicando se ele deve ser usado para teste funcional ou teste estrutural e apresente um exemplo explicando sua utilização (valor: 2,0 pontos; máximo: 10 linhas).

Teste funcional:

Particionamento por equivalência: usado para identificar as classes de equivalência válidas e inválidas para um determinado conjunto de dados de entrada. Por exemplo, se um determinado atributo Idade < 10 , as possíveis classes de equivalência inválidas seriam Idade ≥ 10 , Idade < 0 (considerando que idade 0 representa um indivíduo que não completou 1 ano de vida ainda) e válida $0 \leq \text{Idade} \leq 9$. Portanto, um valor de cada uma destas faixas deve ser utilizado para o teste. Com isso se consegue obter um conjunto mínimo com cobertura adequada para testar a existência deste atributo.

Outros possíveis critérios: grafo de causa-efeito (funcional), todos_caminhos (estrutural), todos_fluxos (estrutural), ... Ver material das aulas de testes para detalhamento.