

Questão 1) Defina o que é manutenção de software e os seus principais tipos.

A manutenção de software é o processo geral de mudança em um sistema depois que ele é liberado para uso. O termo geralmente se aplica ao software customizado em que grupos de desenvolvimento separados estão envolvidos antes e depois da liberação. As alterações feitas no software podem ser simples mudanças para correção de erros de codificação, até mudanças mais extensas para correção de erros de projeto, ou melhorias significativas para corrigir erros de especificação ou acomodar novos requisitos. As mudanças são implementadas por meio da modificação de componentes do sistema existente e, quando necessário, por meio da adição de novos componentes. Tipos de manutenção de software:

1. Correção de defeitos. Erros de codificação são relativamente baratos para serem corrigidos; erros de projeto são mais caros, pois podem implicar reescrever vários componentes de programa. Erros de requisitos são os mais caros para se corrigir devido ao extenso reprojetado de sistema que pode ser necessário.
2. Adaptação ambiental. Esse tipo de manutenção é necessário quando algum aspecto do ambiente do sistema, como o hardware, a plataforma do sistema operacional ou outro software de apoio sofre uma mudança. O sistema de aplicação deve ser modificado para se adaptar a essas mudanças de ambiente.
3. Adição de funcionalidade. Esse tipo de manutenção é necessário quando os requisitos de sistema mudam em resposta às mudanças organizacionais ou de negócios. A escala de mudanças necessárias para o software é, frequentemente, muito maior do que para os outros tipos de manutenção.

Questão 2) Sobre teste de software considere as afirmações:

I. Uma estratégia de teste que é escolhida por grande parte das equipes de software adota uma visão incremental do teste, começando com o teste de unidades individuais de programa, avançando para testes projetados a fim de facilitar a integração das unidades e culmina com testes que exercitam o sistema construído.

II. O teste de unidade focaliza o esforço de verificação na menor unidade de projeto do software - o componente ou módulo de software. Usando a descrição de projeto no nível de componente como guia, caminhos de controle importantes são testados para descobrir erros dentro dos limites do módulo.

III. O teste de unidade é normalmente considerado um apêndice ao passo de codificação. O projeto de teste de unidade pode ser realizado antes que o código seja iniciado ou depois de o código-fonte ter sido gerado.

IV. O teste de integração é uma técnica sistemática para construir a arquitetura do software enquanto, ao mesmo tempo, conduz testes para descobrir erros associados às interfaces. O objetivo é, a partir de componentes testados no nível de unidade, construir uma estrutura de programa determinada pelo projeto.

Assinale a alternativa correta, referente aos itens I, II, III e IV.

- ☐ II, III e IV, apenas.
☒ I, II, III e IV.
☐ III e IV, apenas.
☐ I e III, apenas.
☐ I, II e IV, apenas.

Questão 3) Sobre teste de software, assinale a alternativa correta abaixo:

- ☐ o teste da caixa preta (teste funcional), baseia-se em analisar os arquivos de log do sistema procurando por mensagens de funcionamento inconsistente.
- ☐ um teste bem sucedido é um teste que não encontra nenhum erro no software.
- ☐ um caso de teste é uma categoria de possíveis resultados na execução de testes.
- ☐ o teste de operação é a fase onde é testada a ergonomia da interface de uso do software.
- ☒ o teste da caixa branca (teste estrutural), baseia-se em testar as estruturas do código fonte, como comandos condicionais e de repetição.

Questão 4) Uma das técnicas mais comuns utilizadas para o desenvolvimento/execução de testes de software é chamada de Caixa-Preta. Selecione os tipos de teste que são aplicáveis essa técnica:

- A - unitário.
 - B - integração.
 - C - sistema/funcional.
 - D - aceitação.
- Assinale a alternativa correta.

• Parte superior do formulário

- ☐ da relação apresentada somente o A, B e C
- ☐ da relação apresentada somente o A, C e D
- ☐ da relação apresentada somente o B, C e D
- ☒ da relação apresentada todos são aplicáveis essa técnica

Parte inferior do formulário

QUESTAO 5). Assinale com V (verdadeiro) ou F (falso) as afirmações abaixo sobre os vários tipos de testes existentes.

- ☒ Teste é a forma de avaliação de qualidade mais comum porque é a que fornece melhor custo-benefício, comparada com verificação formal e tolerância a falhas.
- ☐ As atividades relacionadas ao teste do software devem começar assim que o primeiro módulo do sistema estiver codificado.
- ☐ A diferença entre teste caixa-preta e teste caixa-branca está apenas na forma como os testes são gerados, pois os dois métodos detectam o mesmo tipo de falha.
- ☐ Teste de unidade só pode ser aplicado quando o sistema estiver completamente desenvolvido, pois uma unidade raramente funciona independente de outras.
- ☒ O teste funcional ou caixa-preta consiste em definir as classes de equivalência e escolher valores-limite em cada classe como dados de teste.
- ☒ O conceito de cobertura de arcos só se aplica ao teste estrutural, uma vez que é preciso conhecimento do código para gerar o grafo de fluxo de controle.

Questão 6) A respeito de engenharia de software, julgue os próximos itens.

A finalidade do processo de homologação de um software é a identificação de defeitos no funcionamento desse software, sob o ponto de vista da expectativa do usuário.

- ☐ Certo. ☒ Errado

Questão 7) Sobre Evolução do Software, assinale a alternativa INCORRETA.

- ☐ A flexibilidade dos softwares é uma das principais razões pelas quais softwares vêm se transformando em sistemas grandes e complexos.
- ☐ Uma vez que a decisão da construção de um hardware foi tomada, é caríssimo fazer alterações em seu projeto. No entanto, mudanças em um software podem ser feitas a qualquer momento durante ou após o desenvolvimento.
- ☐ Distinção entre desenvolvimento e manutenção de software é cada vez mais irrelevante.
- ☐ A mudança é inevitável em todos os grandes projetos de software.
- ☒ Mudanças no desenvolvimento de software não aumentam os custos, isso ocorre porque o trabalho seria

feito de qualquer forma. Muda-se apenas a maneira de como determinada função seria feita.

Questão 8) Considerando o desenvolvimento e a manutenção de sistemas e aplicações de software, marque V para as afirmativas verdadeiras e F para as falsas.

(F) No ciclo de vida tradicional de um software, são fases clássicas: análise, projeto, codificação, testes, evolução e transição.

(V) A construção de protótipos pode servir como um mecanismo para apoiar a validação de requisitos de usuários, isto é, o cliente pode visualizar a aplicação e oferecer feedback ao projetista de software.

(F) O desenvolvimento de aplicações no paradigma orientado a objeto requer o conhecimento de técnicas de delegação e distribuição de responsabilidades, que possibilitam o reuso de código, pois apresentam alto acoplamento e são aderentes aos padrões de projetos.

Questão 9). A finalidade do processo de homologação de um softwar é a identificação de defeitos no funcionamento desse software, sob o ponto de vista da expectativa do usuário.

() Certo. (X) Errado

Questão 10). Sobre Evolução do Software, assinale a alternativa INCORRETA.

() A flexibilidade dos softwares é uma das principais razões pelas quais softwares vêm se transformando em sistemas grandes e complexos.

() Uma vez que a decisão da construção de um hardware foi tomada, é caríssimo fazer alterações em seu projeto. No entanto, mudanças em um software podem ser feitas a qualquer momento durante ou após o desenvolvimento.

() Distinção entre desenvolvimento e manutenção de software é cada vez mais irrelevante.

() A mudança é inevitável em todos os grandes projetos de software.

(X) Mudanças no desenvolvimento de software não aumentam os custos, isso ocorre porque o trabalho seria feito de qualquer forma. Muda-se apenas a maneira de como determinada função seria feita.

Questão 11) Sobre o processo de mudança de software, assinale a opção correta.

() A manutenção adaptativa para adaptar o software a mudanças ambientais se refere a mudanças de diversos tipos, como uma nova plataforma de hardware. Mudanças de sistemas operacionais ou software em geral não estão incluídas nessa categoria.

() A manutenção corretiva para reparar defeitos de software diz respeito à mudança em um sistema para corrigir erros, que podem ser somente de codificação. Erros de projeto e de lógica não estão incluídos nessa categoria.

(X) A manutenção preventiva tem como objetivo detectar e corrigir falhas latentes no produto antes que elas virem falhas efetivas.

() A manutenção perfectiva tem como objetivo tornar o software perfeito, melhorar seu desempenho e sua manutenibilidade ou adicionar-lhe funcionalidade para satisfazer a novos requisitos.

() A manutenção de software é um processo geral de mudanças que ocorre antes ou depois que um sistema é entregue.