

CONCEITOS BÁSICOS DE SISTEMAS COMPUTACIONAIS

QUESTÃO 31

A evolução dos sistemas operacionais permitiu a otimização do uso do processador e a melhor utilização dos recursos computacionais pelos usuários. Considerando as técnicas utilizadas pelos sistemas operacionais modernos, julgue os itens abaixo.

- ① A multiprogramação permite a execução de programas em seqüência, pelo processador, a partir de uma fila de *jobs* denominada *spooling* de entrada.
- ② O multiprocessamento permite a execução de múltiplas tarefas simultâneas em vários processadores.
- ③ A paginação é uma forma de comunicação entre processos que utilizam memória comum compartilhada e segmentada em blocos de tamanho fixo.
- ④ A troca de mensagens permite o sincronismo e a comunicação entre processos.
- ⑤ A memória virtual é uma forma de gerência de acesso e expansão da memória física secundária.

QUESTÃO 32

Segundo Peter Chen, o projeto de banco de dados é um processo complicado, uma vez que o projetista tem de considerar não apenas como modelar o mundo real, mas também as limitações do sistema de banco de dados e a eficiência de recuperação e atualização dos dados. Considerando os problemas de projeto lógico de banco de dados, julgue os itens seguintes.

- ① Os relacionamentos muitos-para-muitos entre dois tipos de entidades, tais como relacionamentos entre funcionários e projetos, não podem ser representados diretamente na modelagem E-R.
- ② A independência dos dados diz respeito à habilidade de modificar o esquema conceitual ou físico, sem a necessidade de reescrever os programas aplicativos.
- ③ As linguagens de manipulação de dados permitem o acesso e a manipulação dos dados tal como organizados pelo modelo de dados.
- ④ O resultado da compilação de comandos em linguagem de manipulação de dados é um conjunto de tabelas que são armazenadas em um arquivo especial, chamado **dicionário de dados**.
- ⑤ O controle de concorrência implementado pelos sistemas gerenciadores de banco de dados preservam os requisitos de segurança de acesso aos dados.

QUESTÃO 33

Cada camada do modelo OSI é constituída por um conjunto de subsistemas, cada um residente em um sistema aberto diferente, responsáveis por funções específicas. Cada subsistema, por sua vez, é constituído por uma ou mais entidades, que são elementos ativos responsáveis pela execução de um conjunto de funções de uma dada camada. Considerando essas informações, julgue os itens a seguir.

- ① As entidades (N) prestam serviços às entidades (N+1), utilizando, na execução das funções (N), os serviços prestados pelas próprias entidades (N).
- ② Entidades (N) de sistemas abertos distintos comunicam-se entre si por intermédio de protocolos da camada (N).
- ③ A comunicação entre entidades de camadas adjacente de sistemas abertos distintos ocorre por meio de pontos de acesso de serviço SAP.
- ④ As primitivas de serviço descrevem eventos que ocorrem entre duas camadas do mesmo sistema aberto.
- ⑤ Um serviço (N) com conexão corresponde a uma associação entre entidades (N) pertencentes a sistemas abertos distintos.

QUESTÃO 34

Acerca da arquitetura TCP/IP, julgue os itens que se seguem.

- ① A arquitetura não determina as sub-redes de comunicação, permitindo que diversas tecnologias possam ser empregadas para interconectar um sistema computacional a uma rede *internet*.
- ② O IP não exige conexão, não estabelecendo circuitos virtuais para transporte dos pacotes de dados.
- ③ Os protocolos de transporte TCP são usados na troca de dados fim-a-fim entre dois sistemas de uma rede *internet*.
- ④ No TCP, o controle de fluxo é implementado pela técnica de janela deslizante, com alocação de créditos expressos em número de pacotes.
- ⑤ A segmentação garante que um *datagrama* possa atravessar sub-redes com diferentes limites de tamanho de bloco de transmissão, até o destino.

QUESTÃO 35

Um sistema cliente-servidor distribuído é qualquer sistema que divide as tarefas computacionais (implementadas por meio de módulos, tais como rotinas, funções, objetos e processos) entre dois ou mais sistemas computacionais distintos. O cliente efetua solicitações de algum serviço e o servidor responde, realizando-o. Considerando a definição de sistema cliente-servidor, julgue os seguintes itens.

- ① Os resultados de uma mesma solicitação de serviço *idempotente* com semântica ao-menos-um modificam-se após a ocorrência de atrasos e falhas na comunicação cliente-servidor.
 - ② Na solução cliente-servidor para a WWW, o diálogo entre *browser* e servidor é implementado por meio de interfaces de programação como RPC, BSD *sockets* e *WinSockets*.
 - ③ No modelo cliente-servidor em três níveis (*three-tiered*), um sistema é constituído dos módulos Armazenamento e Recuperação de Informações, Lógico e Interface de Usuário.
- Uma das vantagens na implementação de sistemas cliente-servidor por meio de RPC é a resolução automática das diferenças de representação de dados entre plataformas distintas.
- ⑤ O uso de *threads* pelo servidor, no tratamento concorrente de múltiplas solicitações de clientes, reduz os custos associados à criação de novos processos e mudanças de contexto.

QUESTÃO 36

A linguagem SQL estabeleceu-se como um padrão para acesso a banco de dados relacional. Como exemplo, considere o seguinte fragmento.

linha	
1	SELECT fnome, produto
2	FROM Fornec
3	WHERE cidade = 'brasil'ia'
4	AND produto IN
5	(SELECT produto
6	FROM Fornec
7	WHERE cidade = 'recife');

A partir dos conceitos da referida linguagem e do fragmento acima, julgue os itens a seguir.

- ① O comando SELECT não obriga o uso da cláusula WHERE.
- ② Se o asterisco fosse usado no comando SELECT da linha 1, estaria indicando a seleção do primeiro registro de uma tabela.
- ③ Na linha 2, Fornec representa uma tabela.
- ④ Um possível resultado da pesquisa incluirá uma lista de fornecedores em Brasília dos mesmos produtos de qualquer um dos fornecedores no Recife.
- ⑤ Um possível resultado da subpesquisa (linhas de 5 a 7) incluirá uma lista de fornecedores no Recife.

TÉCNICAS DE PROGRAMAÇÃO

QUESTÃO 37

Os conceitos de pilhas e filas são valiosos como ferramentas para a concreta solução de problemas em computação. Esses conceitos são aplicações de outra estrutura, a lista, cujas variantes são utilizadas na implementação de pilhas e filas. Considerando tais conceitos, julgue os itens abaixo.

- ① Pilhas implementadas por vetores são mais eficientes no uso da memória disponível para o programa que aquelas implementadas com listas encadeadas.
- ② A eliminação de um elemento de uma fila de prioridade implementada por lista desordenada é menos eficiente que a inserção.
- ③ Listas duplamente encadeadas permitem a busca de elementos em ambos os sentidos.
- ④ Uma fila FIFO implementada como uma lista circular duplamente encadeada exige, pelo menos, dois ponteiros, para o início e o fim da lista.
- ⑤ Após a inserção dos elementos A, B, C e D, nesta ordem, em uma lista do tipo LIFO (*last-in, first-out*), o elemento obtido na segunda retirada será o B.

QUESTÃO 38

Um apontador é uma variável que contém o endereço de outra variável. Apontadores são muito usados, em parte porque são, às vezes, a única forma de se expressar um processo de computação e, em parte, porque, normalmente, implicam um código mais compacto e eficiente que o obtido de outras formas. Considerando o uso de apontadores nas linguagens procedurais atuais, julgue os itens que se seguem.

- ① O uso de apontadores como argumentos de funções que realizem chamada por valor não permite alteração de seus conteúdos de memória na função.
- ② Com apontadores, é possível criar funções que retornem estruturas de dados complexas.
- ③ Para recuperar o valor da variável apontada, basta que o apontador seja atribuído a uma variável de mesmo tipo.
- ④ Arranjos de apontadores podem ser usados em funções que recebam como argumentos um número variável de cadeias de caracteres.
- ⑤ Em geral, para que um arranjo seja percorrido por um ponteiro, para cada elemento do arranjo deve-se adicionar ao ponteiro o número de *bytes* que o elemento ocupa em memória.

QUESTÃO 39

Considere uma linguagem com as características abaixo, em que a precedência dos operadores é maior para aqueles do topo da tabela e menor para aqueles da base.

tipo de operador	operadores	associatividade
expressão primária	()	esquerda para direita
adição, subtração	+ -	esquerda para direita
multiplicação, divisão	* /	direita para esquerda
operadores relacionais	< <= > >=	esquerda para direita
igualdade, diferença	= !=	esquerda para direita
e lógico, ou lógico	&&	direita para esquerda

Considerando verdadeiro = 1, falso = 0 e as variáveis inteiras $a = 4$, $b = 2$, $c = 3$ e $d = 6$, julgue os itens seguintes.

- | | |
|--|---------------------|
| ① $a - b * d / a - c * d / 6 - c$ | resulta 6. |
| ② $d + a / 2 > 3 d * b + 1 < 14 \&\& d / b - 1 < 6$ | resulta verdadeiro. |
| ③ $(1 0 \&\& 0) (1 1 \&\& 0) \&\& (0 0 \&\& 1)$ | resulta verdadeiro. |
| ④ $1 != 0 == 0 != 1 0 != 1 == 1 == 0 \&\& 0 == 1 != 0 == 1$ | resulta falso. |
| ⑤ $d > c \&\& 2 < d 10 > d \&\& c + b < 5 6 != d$ | resulta falso. |

QUESTÃO 40

A frequência de uma linha de programa corresponde à quantidade de vezes que ela é executada. Considere as declarações de dois segmentos de programas abaixo, nas quais o operador \leftarrow é de atribuição e são usadas estruturas tradicionais de controle **for** e **while**. Considere, ainda, que n seja um valor inteiro maior ou igual a 1.

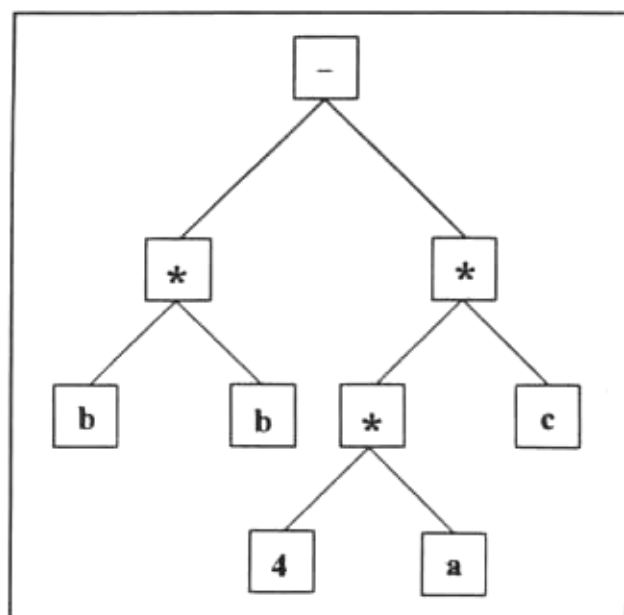
estrutura while	estrutura for
while condição instruções end	for var \leftarrow início to fim by incremento do instruções end
programa I	programa II
1 $i \leftarrow 2$ 2 while $i \leq n$ do 3 $x \leftarrow x + 1$ 4 $i \leftarrow i + 1$ 5 end	1 for $i \leftarrow 1$ to n by 1 do 2 for $j \leftarrow 1$ to n by 1 do 3 for $k \leftarrow 1$ to n by 1 do 4 $x \leftarrow x + 1$ 5 end 6 end 7 end

Com base no texto acima, julgue os itens que se seguem.

- ① A linha 4, no programa I, tem frequência n .
- ② A linha 1, no programa II, tem frequência $(n+1)$.
- ③ A linha 4, no programa II, tem frequência $3n$.
- ④ A linha 2, no programa II, tem frequência $(n^2 + n + 1)$.
- ⑤ Ao se substituir n por i , na linha 2 do programa II, a frequência será menor que $2n$.

QUESTÃO 41

Em linguagens de programação, as expressões podem ser representadas por meio de árvores binárias, com o operador no nó-raiz que deve ser aplicado aos resultados das avaliações das expressões representadas pelas subárvores esquerda e direita. Constantes e variáveis são representadas nos nós-folhas. A representação da estrutura dos operadores e operandos de uma expressão é chamada de árvore de sintaxe abstrata.



Com o auxílio do texto e da representação de árvore binária acima, julgue os itens seguintes.

- ① O nível de um nó, em uma árvore binária, é definido como sendo 0 para o nó-raiz e, para um outro nó qualquer, como sendo um nível a mais que o nível de seu pai.
- ② Uma árvore binária completa, de profundidade d , tem $2^{(d+1)}$ nós.
- ③ A notação infixa da expressão representada pela árvore binária é $b * b - 4 * a * c$
- ④ A notação posfixa da expressão representada pela árvore binária é $b * b * 4 * a * c -$
- ⑤ A notação prefixa da expressão representada pela árvore binária é $- * b b * * 4 a c$

QUESTÃO 42

No enfoque orientado a objetos, os átomos do processo de computação são os objetos. É correto relacionar como benefício do desenvolvimento orientado a objetos o(a)

- ① adaptabilidade aos modelos tradicionais de gestão.
- ② adequabilidade às medidas de desempenho de sistemas.
- ③ aumento da produtividade no desenvolvimento.
- ④ facilidade para a manutenção de sistemas.
- ⑤ facilidade para a gerência da complexidade.

QUESTÃO 43

Se tivéssemos aplicativos orientados ao objeto, existiria uma *ecologia* de aplicativos diferentes. Seria o fim da era dos aplicativos monolíticos.

Kapor, 1983

A ecologia de sistemas a que Kapor se refere no texto é suportada, na orientação a objetos, pelos mecanismos de

- ① encapsulamento.
- ② processos.
- ③ polimorfismo.
- ④ protocolos.
- ⑤ abstração.

QUESTÃO 44

Como a orientação a objetos está integrada aos componentes fundamentais de *software*, ela representa o que a programação estruturada representava nos anos 70: um importante paradigma de criação, manutenção e uso de *software*. Acerca das semelhanças entre esses dois paradigmas, julgue os itens que se seguem.

- ① Um método é como um procedimento, porque dois contêm instruções de processamento.
- ② As variáveis de classes e instâncias correspondem aos dados na programação tradicional.
- ③ A hereditariedade pode ser representada na programação convencional por meio da passagem de parâmetros entre procedimentos ou funções.
- ④ Um objeto é como um tipo de dados inicializado, sendo que, na orientação a objetos, a inicialização ocorre por intermédio de mensagens.
- ⑤ O envio de mensagens substitui as chamadas de função como método primário de controle nos sistemas orientados a objetos.

QUESTÃO 45

No estudo da lógica, é costume reservar o nome de *falácia* a aqueles argumentos e raciocínios que, embora incorretos, possam ser psicologicamente persuasivos. Julgue os itens que se seguem, considerando certos aqueles em que *inexistem* argumentos logicamente falaciosos.

- ① Ao julgar a validade de um argumento, pode-se incorrer em erro de raciocínio por falta de atenção ao tema a que esse argumento se refere, ou por ser iludido por alguma ambigüidade na linguagem usada para formulá-lo.
- ② Não creio que o cigarro faça mal à saúde. Fumo há vinte anos e conheço gente que morreu mais jovem que eu sem ter botado cigarro algum na boca.
- ③ Não creio em telepatia, pois ninguém, até hoje, provou cientificamente que tal fenômeno existe.
- ④ Esse programa de computador funciona corretamente, visto que, até hoje, nenhum usuário reclamou de seu comportamento.
- ⑤ A liberdade total do indivíduo deve ser garantida pelo Estado, pois interessa ao Estado que seus indivíduos gozem de toda a liberdade.

QUESTÃO 46

As técnicas de definição denotativas utilizam exemplos de objetos para designar classes; as conotativas utilizam o alcance de propriedades em uma classe para formar subclasses; as operacionais utilizam resultados de operações sobre objetos para decidir sua pertinência a classes. De acordo com esses conceitos, julgue os itens seguintes.

- ① Termos como **entidade e objeto** não são passíveis de definição conotativa.
 - ② A frase "Segundo Einstein, espaço é o que pode ser medido por réguas" contém uma definição denotativa de espaço.
 - ③ A frase "O triângulo é um polígono de três lados" contém uma definição conotativa.
 - ④ A frase "Arranha-céus de Brasília são os edifícios tão ou mais altos que o do Congresso Nacional" expressa uma definição denotativa.
- A frase "O homem é um bipede implume" define conotativamente a espécie homem.

QUESTÃO 47

Um silogismo categórico típico é um argumento contendo três proposições categóricas. Nele, uma conclusão é derivada de duas premissas, e três termos ocorrem exatamente duas vezes: o termo predicado da conclusão ocorre também em uma das premissas e o termo sujeito da conclusão ocorre na outra premissa. Julgue se os itens abaixo representam silogismos categóricos válidos.

- ① Nenhum universitário tem QI inferior a 70, mas todas as pessoas com QI inferior a 70 são estúpidas. Logo, nenhum universitário é estúpido.
- ② Alguns professores são homens de ação; daí, alguns soldados são professores, porque todos os soldados são homens de ação.
- ③ Nenhum criminoso é herói, pois todos os criminosos são pessoas más, e nenhum herói é pessoa má.
- ④ Alguns neuróticos não são parasitas, mas todo estelionatário é parasita. Segue-se que alguns neuróticos não são estelionatários.
- ⑤ Todas as ações penais são atos cruéis. Todo processo por homicídio é uma ação penal. Logo, todo processo por homicídio é um ato cruel.

QUESTÃO 48

Considerando que o símbolo \wedge representa a conjunção lógica, \vee , a disjunção lógica, \neg , a negação e o símbolo \rightarrow representa implicação material, julgue se as seqüências apresentadas nos itens abaixo constituem provas formais da validade do último argumento, a partir das hipóteses anunciadas.

- ① hipóteses: $(X \vee W) \rightarrow Z$; $X \wedge Y$.
prova: X ; $X \vee W$; Z ; $X \wedge Z$.
- ② hipóteses: $\neg(A \wedge C)$; $(A \wedge B) \rightarrow C$; $A \rightarrow B$.
prova: $A \rightarrow (A \wedge B)$; $A \rightarrow C$; $A \rightarrow (A \wedge C)$; A .
- ③ hipóteses: $D \rightarrow E$; $D \vee G$; $E \rightarrow F$; $G \rightarrow H$.
prova: $D \rightarrow F$; $(D \rightarrow F) \wedge (G \rightarrow H)$; $F \vee H$.
- ④ hipóteses: $M \rightarrow \neg N$; $\neg M \rightarrow (P \rightarrow \neg Q)$; $(\neg P \vee \neg Q) \rightarrow N$; $\neg Q$.
prova: $\neg Q \vee \neg P$; N ; $\neg M$; $P \rightarrow \neg Q$; $\neg P$.
- ⑤ hipóteses: $K \rightarrow I$; $(K \rightarrow J) \rightarrow (L \vee I)$; $(K \vee I) \rightarrow J$; $\neg L$.
prova: $K \rightarrow (K \wedge I)$; $K \rightarrow J$; $(L \vee I)$; $\neg I$.

ANÁLISE DE SISTEMAS

QUESTÃO 49

Os modelos são abstrações construídas para que um problema seja compreendido antes da implementação de uma solução. Com relação aos diferentes modelos, julgue os itens que se seguem.

- ① O modelo que expressa detalhes do problema representa melhor a realidade, sendo, em geral, preferido no processo de modelagem de sistemas.
- ② O modelo de objetos representa os aspectos estáticos e estruturais de dados de um sistema.
- ③ O modelo com base em eventos representa processos e fluxo de dados que são ativados a partir de operações.
- ④ O modelo dinâmico representa os aspectos temporais, comportamentais e de controle de um sistema.
- ⑤ O modelo funcional representa os aspectos relativos às transformações de valores, independentemente de como ou quando é feito.

Leia o texto abaixo e responda às questões 50 e 51.

Os sistemas atuais são diferentes do que eram há dez ou vinte anos. Em todos os aspectos, são maiores e mais complexos: são mais voláteis e sujeitos a alterações constantes. Além disso, os sistemas interativos e *on-line* dedicam muito mais atenção à interface com o usuário do que os antigos sistemas de processamento *batch* das décadas de 60 e 70. Um enfoque de análise, projeto e codificação com base em tais sistemas é uma forma natural de tratar sistemas voltados para usuários. Nesses sistemas, a complexidade funcional é menos importante e a modelagem de dados tornou-se prioritária.

Peter Coad

QUESTÃO 50

Os problemas principais da análise de sistemas atuais incluem o(a)

- ① estudo do domínio do problema, suas características e as responsabilidades do sistema.
- ② comunicação efetiva, visando extrair informações acerca do problema e dos requisitos do cliente.
- ③ identificação de um conjunto de requisitos que permaneça estável com o tempo.
- ④ identificação e a eliminação dos requisitos instáveis, visando garantir um ponto de equilíbrio no sistema.
- ⑤ utilização de planejamentos de domínio de problemas para a reutilização presente e futura.

QUESTÃO 51

Os princípios essenciais para a administração da complexidade dos sistemas modernos incluem

- ① a abstração como forma de ignorar aspectos não-relevantes, tornando possível uma concentração maior nos assuntos principais do problema.
- ② a herança como forma de diminuição do trabalho de desenvolvimento de um novo sistema, agrupando os aspectos relacionados, minimizando o fluxo entre as diferentes partes e separando os requisitos específicos.
- ③ o encapsulamento como um mecanismo de expressão das similaridades entre classes, simplificando a definição de classes iguais a outras já definidas, representando generalização e especialização, tornando explícitos os atributos e serviços comuns em uma hierarquia.
- ④ a associação, para agrupar certos aspectos que ocorrem em algum ponto no tempo ou sob circunstâncias similares.
- ⑤ os métodos de organização, visando identificar objetos e atributos, todo e partes, classes, membros e distinção entre eles.

QUESTÃO 52

Os modelos de dados orientados a objeto (O-O) incluem muitos dos conceitos do modelo entidade-relacionamento (E-R). Considerando as semelhanças e diferenças entre esses modelos, é correto afirmar que

- ① o modelo E-R baseia-se em uma percepção de um mundo que consiste de uma coleção de objetos básicos, chamados entidades, e em relacionamentos entre esses objetos.
- ② um objeto, no modelo O-O, contém valores armazenados em variáveis de instância dentro do objeto, sendo que eles mesmos são objetos.
- ③ uma entidade, no modelo E-R, é um objeto distinguível de outro objeto por um conjunto específico de atributos e operações sobre esses atributos.
- ④ o modelo O-O, diferentemente das entidades no modelo E-R, considera cada objeto como uma identidade própria e única, independentemente do valor que cada um contém.
- ⑤ os objetos que contêm os mesmos tipos de valores e os mesmos métodos são agrupados, no modelo O-O, em classes, que são semelhantes aos relacionamentos entre entidades do modelo E-R.

QUESTÃO 53

A análise de sistemas foi definida por Tom DeMarco como sendo o estudo de um problema, antes de qualquer ação. Considerando os diferentes métodos e processos de análise, é correto afirmar que

- ① o principal produto da análise de sistemas é a especificação de um documento que deve conter uma declaração completa, consistente e possível do que é necessário no sistema.
- ② a decomposição funcional é reconhecida pela identificação de passos e subpassos antecipados de processamento para um novo sistema, incluindo suas interfaces e apresentando, como resultado, os níveis de sistema, subsistema, função e subfunção.
- ③ a decomposição funcional, na análise orientada a objeto, é utilizada em um contexto específico: na definição de um serviço que determina um comportamento específico de um objeto.
- ④ o enfoque da análise estruturada utiliza diagramas de fluxo de dados que permitem representar relacionamentos lógicos além daqueles permitidos pelo enfoque funcional tradicional.
- ⑤ os diagramas de fluxo de dados são muito úteis para sistemas ou partes de sistemas que, principalmente, atualizam e recuperam dados.

QUESTÃO 54

As ferramentas CASE têm sido criadas para tornar o desenvolvimento de sistemas mais rápido, menos dispendioso e de melhor qualidade. Acerca dessas ferramentas, é correto afirmar que

- ① elas incluem como característica a modelagem da realidade da forma mais representativa possível e a tradução desses modelos em sistemas, de forma tão automática quanto possível.
- ② a ênfase maior da construção de sistemas com as ferramentas CASE-OO reside na modelagem e no projeto, não na programação.
- ③ elas incorporam meios gráficos para expressar planos, modelos e projetos.
- ④ as ferramentas CASE-OO estão atreladas às linguagens de programação orientadas a objetos.
- ⑤ as ferramentas CASE-I mais simples não incluem a geração automática de código.

QUESTÃO 55

O modelo de ciclo de vida clássico é o mais antigo paradigma utilizado em desenvolvimento de *software*. Com base nessa afirmativa e identificando problemas associados a esse enfoque, julgue os itens seguintes.

- ① A versão operacional de um sistema só fica pronta no final do projeto. Se um erro não for detectado antes do final do projeto, o resultado pode ser desastroso.
- ② O modelo de ciclo de vida dificulta a distribuição de trabalho em equipes e o acompanhamento do projeto.
- ③ Os projetos reais de desenvolvimento de *software* raramente seguem o fluxo seqüencial que o modelo propõe. O paradigma de ciclo de vida pressupõe um fluxo dessa natureza.
- ④ O ciclo de vida clássico requer que todos os requisitos sejam explicitamente especificados no início do projeto.
- ⑤ O paradigma de ciclo de vida proporciona um modelo bem definido, no qual são especificadas, de forma clara, as fases de análise – projeto, codificação, teste e manutenção –, o que facilita a estruturação do projeto de desenvolvimento de *software*.

QUESTÃO 56

O segmento da engenharia de *software* que trata do teste e da manutenção de *software* é uma das áreas que mais demandam recursos das organizações. A respeito desse assunto, julgue os itens que se seguem.

- ① A gerência de configuração é responsável por identificar, organizar e controlar as modificações a serem feitas em um *software* desenvolvido por uma equipe de desenvolvimento, com o objetivo de maximizar a produtividade e minimizar os erros.
- ② Os métodos de teste de *software* denominados *white box* focalizam os requisitos funcionais do *software*; esse tipo de teste procura analisar todas as condições relacionadas aos requisitos funcionais de um programa.
- ③ O método de teste de *software* denominado *black box* utiliza a estrutura dos programas para realizar o teste, levando em conta as decisões lógicas, os *loops* e seus limites e o teste das estruturas de dados para assegurar sua validade.
- ④ A reengenharia de *software* recupera a informação do projeto de *software* existente e a usa para alterar ou reconstruir o sistema existente, no esforço de melhorar a qualidade.
- ⑤ A engenharia reversa para *software* é o processo de analisar um programa para criar a sua representação em um nível maior de abstração que o código-fonte.

QUESTÃO 57

O planejamento de projeto de *software* inclui estimativas de prazos, custos e recursos de projeto. Acerca desse assunto, julgue os seguintes itens.

- ① As técnicas de estimativa de custo que utilizam linhas de código exigem uma decomposição funcional mais detalhada do projeto que as técnicas que utilizam pontos de função para atingir o mesmo objetivo.
- ② As técnicas de estimativa de tempo e custo de *software* são simples e precisas, dando ao gerente um instrumento extremamente confiável para a elaboração do projeto.
- ③ Os modelos de estimativa COCOMO e Putnam são modelos com base em ponto de função.
- ④ O objetivo do planejamento do projeto de *software* é prover uma estrutura ou um método que permita ao gerente fazer estimativa de recursos, custos e cronograma. As técnicas de estimação de custo proporcionam esse instrumento.
- ⑤ As técnicas de decomposição adotam um enfoque de dividir e conquistar para estimativa de projeto de *software*. Os modelos empíricos utilizam a experiência passada para estabelecer as funções que permitam estimar custos, prazos e recursos do projeto. Essas duas abordagens são excludentes.

QUESTÃO 58

A qualidade de *software* pode ser afetada por vários fatores, entre os quais se destacam: confiabilidade, eficiência, integridade, usabilidade, manutenibilidade, flexibilidade, testabilidade, adaptabilidade, portabilidade, reusabilidade e interoperabilidade. Com relação a esses fatores, julgue os itens abaixo.

- ① A confiabilidade representa o controle de acesso ao *software* ou a dados por pessoas não-autorizadas.
- ② A eficiência é o esforço requerido para aprender a operar, preparar entradas e interpretar saídas de um programa.
- ③ A reusabilidade define até que ponto um programa ou parte dele pode ser reusado em outras aplicações.
- ④ Interoperabilidade é o esforço requerido para testar um programa, a fim de garantir que ele execute as funções pretendidas.
- ⑤ A correção é o esforço requerido para localizar e consertar erros em um programa.

QUESTÃO 59

A gestão do processo de *software* permeia toda a atividade de desenvolvimento. Essa atividade pode ser dividida nos seguintes componentes principais: especificação do projeto de *software*, medidas e métricas, estimativas, análise de risco, cronogramas, orçamento, acompanhamento e controle. De acordo com essa classificação, julgue os itens a seguir.

- ① O planejamento de um projeto de *software* não exige que, *a priori*, sejam estabelecidos os objetivos e o escopo do problema, pois, geralmente, existe alto grau de incerteza nesse tipo de projeto.
- ② As métricas e medidas auxiliam a entender o processo técnico usado para desenvolver um produto de *software*. O processo é medido para poder ser aperfeiçoado e o produto, para melhorar sua qualidade.
- ③ A análise de risco constitui uma série de passos que permitem reduzir os riscos de um projeto de *software*. Esses passos são: identificação de risco, avaliação de risco, priorização, estratégias para gestão de risco, resolução de risco e monitoração de risco.
- ④ O processo de programação, incluindo cronograma, orçamento do projeto de *software*, deve ser simples e estruturado, e modelos fixos de cronogramas podem ser definidos e utilizados para diferentes projetos de *software*.
- ⑤ O acompanhamento e o controle de projetos de *software* são feitos de forma semelhante a outros projetos de engenharia; não é necessário monitorar mudanças de requisitos durante a execução das tarefas, pois as metas e os produtos intermediários são claramente definidos na totalidade dos projetos.

QUESTÃO 60

Atualmente, um dos mais importantes assuntos da gestão de projetos de *software* consiste na terceirização total ou parcial do desenvolvimento de sistemas. Após dez anos dessa prática, várias lições foram aprendidas, a partir de falhas ocorridas em projetos de terceirização, tanto na indústria quanto na acadêmica. Com relação a esse tema, julgue os itens seguintes.

- ① As informações públicas retratam uma visão extremamente pessimista da terceirização do desenvolvimento de sistemas.
- ② As empresas prestadoras de serviços são, por definição, mais eficientes que um departamento de informática de uma organização.
- ③ O departamento de informática de uma organização pode ser capaz de alcançar resultados melhores ou similares aos das empresas prestadoras de serviços de desenvolvimento de sistemas.
- ④ Se uma empresa decide terceirizar, o contrato é o único mecanismo que assegura que os objetivos do projeto sejam alcançados. Nesse tipo de terceirização, só existe parceria quando os lucros ou prejuízos são compartilhados entre os parceiros.
- ⑤ As atividades de tecnologia de informação, na maioria das empresas, podem ser consideradas não-estratégicas e, portanto, passíveis de terceirização integral.