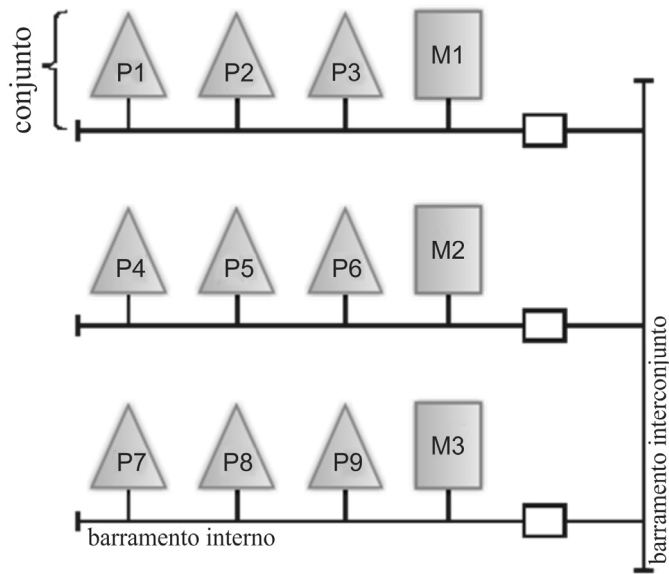


CONHECIMENTOS ESPECÍFICOS



Com relação à figura acima, que ilustra um sistema com múltiplos processadores, na qual os símbolos triangulares, identificados de P1 a P9, representam processadores; os símbolos retangulares, identificados de M1 a M3, representam memórias; ao passo que as linhas que interligam esses conjuntos são barramentos, julgue o item a seguir.

51 A figura em tela representa a arquitetura de sistemas NUMA (*non-uniform memory access*), na qual o tempo de acesso à memória principal depende da localização física do processador, e todos os conjuntos compartilham o mesmo sistema operacional e referenciam o mesmo espaço de endereçamento.

A respeito do gerenciamento do processador e da memória em sistemas operacionais, julgue os itens subsequentes.

52 Caso utilize a técnica de memória virtual, um sistema operacional pode carregar e executar um programa de 32 Mb em um computador com 8 Mb de memória principal.

53 Tempo de *turnaround* é o tempo total que um processo permanece na fila, durante o seu processamento, enquanto aguarda para ser executado.

Acerca dos conceitos de rede de computadores, julgue os itens que se seguem.

54 Modulação e multiplexação são técnicas amplamente utilizadas em canais de comunicação com e sem fio. A multiplexação é concebida para transportar sinais em múltiplos fios.

55 A Ethernet comutada por intermédio de enlaces ponto a ponto utiliza o *switch* como equipamento que repassa os pacotes entre os computadores conectados e utiliza o endereço existente em cada pacote para determinar o computador destino.

A respeito dos protocolos de rede de computadores, julgue os itens subsequentes.

56 TTL (*time to live*) é um campo do protocolo IP (*internet protocol*) que determina o tempo que o pacote de dados pode permanecer na rede antes que o equipamento de rede o destrua. Esse procedimento evita o congestionamento da rede pelos pacotes de dados perdidos.

57 O CSMA/CD (*carrier sense multiple access with collision detection*) garante que uma estação da rede não comece a transmitir dados se detectar que o canal está ocupado, sendo os algoritmos da camada de enlace responsáveis por estabelecer o novo momento de transmissão dos dados.

À luz da norma ISO/IEC 38500, julgue os próximos itens, acerca de governança de TI.

58 O comportamento humano é um dos princípios de boa governança corporativa de TI no sentido de se orientar processos de tomada de decisão.

59 Os mecanismos e componentes integrados da governança de TI servem como base para o processo de tomada de decisão da empresa e subsidiam o planejamento estratégico.

No que se refere ao COBIT versão 5, julgue os itens seguintes.

60 O COBIT 5 provê um modelo de maturidade genérico que demonstra como o gerenciamento do controle interno e a necessidade do estabelecimento de melhores controles tipicamente se desenvolvem de um nível *ad hoc* para um nível otimizado.

61 É recomendável que o COBIT seja utilizado em todos os níveis organizacionais, pois ele se concentra em como atingir o conjunto de atividades previstas para a TI em vez de se preocupar com o que deve ser atingido.

62 As tabelas RACI (*responsible, accountable, consult e inform*) associam as atividades do processo aos papéis individuais na organização e podem ter relação com o habilitador estruturas organizacionais.

Com relação a ITIL versão 3, julgue os itens a seguir.

63 O gerenciamento da continuidade dos serviços de TI, pertencente ao desenho de serviço, tem como objetivo manter e melhorar a qualidade dos serviços de TI por intermédio de um ciclo contínuo de atividades que envolvem planejamento, coordenação, elaboração e estabelecimento de acordo de metas de desempenho e responsabilidades mútuas.

64 A garantia da qualidade de um serviço novo ou alterado é uma política explícita para o processo de transição.

65 O portfólio de aplicativo, utilizado no desenho de serviço da ITIL, pode ser implementado como parte do portfólio de serviço ou do sistema de gerenciamento de configuração.

No que se refere a gerenciamento de projetos PMBOK 5, julgue os próximos itens.

- 66 Caso o gerente de um projeto necessite recalcular a duração de sua execução, decidindo-se por analisar a sequência de atividades do cronograma com menor flexibilidade, a melhor técnica para esse fim será o método do diagrama de precedência.
- 67 O caminho crítico de um projeto é calculado a partir das datas de início e término mais cedo e das datas de início e término mais tarde, para todas as atividades, sem que sejam consideradas quaisquer limitações de recursos. Assim, a sequência de atividades obtida representa o caminho mais longo e determina a menor duração possível.
- 68 A técnica heurística pode ser utilizada para estimar partes do cronograma de um projeto.

Julgue os seguintes itens, relativos a gerenciamento de projetos e à gestão ágil de projetos com Scrum.

- 69 No contexto da metodologia Scrum, ocorre um *sprint* quando o *scrum master* acelera o ciclo de desenvolvimento de um produto ou serviço.
- 70 Os objetivos de uma *daily scrum* incluem a disseminação do conhecimento sobre o que foi executado no dia anterior, a identificação de impedimentos e a priorização do trabalho a ser realizado.
- 71 O índice de desempenho para término é uma métrica de desempenho de custos obtida pela razão entre o custo para se terminar o trabalho restante e o orçamento restante do projeto.

No que concerne a DevOps, julgue os itens que se seguem.

- 72 O profissional especialista em DevOps deve atuar e conhecer as áreas de desenvolvimento (engenharia de *software*), operações e controle de qualidade, além de conhecer, também, de forma ampla, os processos de desenvolvimento ágil.
- 73 DevOps é um conceito pelo qual se busca entregar sistemas melhores, com menor custo, em menor tempo e com menor risco.

Com referência aos conceitos básicos de engenharia de *software*, julgue os itens seguintes.

- 74 Os procedimentos de engenharia de *software* constituem o elo entre os métodos e as ferramentas, o que possibilita o desenvolvimento racional e oportuno de *software*.
- 75 A engenharia de *software*, cujo objetivo é o de apoiar o desenvolvimento de programas de computador, prioriza a programação individual e especializada, excluindo-se a documentação e as configurações do programa, que são tratados pela engenharia de sistemas e pela ciência da computação.
- 76 Os métodos da engenharia de *software* proporcionam os detalhes de como construir o *software* e introduzem, muitas vezes, uma notação gráfica ou orientada à linguagem especial.
- 77 As etapas da engenharia de *software* são, frequentemente, citadas como paradigmas, cuja escolha é realizada em função da natureza do projeto e da aplicação, dos métodos e das ferramentas a serem utilizados.

Acerca de processos e práticas ágeis de desenvolvimento e de análise de requisitos funcionais e não funcionais, julgue os itens a seguir.

- 78 Serão classificados como organizacionais os requisitos operacionais e de desenvolvimento que forem identificados na análise de requisitos não funcionais.
- 79 O foco nos métodos ágeis está na entrega incremental do *software*, ou seja, na entrega de valor, com fundamento no princípio de processos prescritivos.
- 80 Na *Extreme Programming*, a programação em pares cria ilhas de especialistas na equipe por meio da análise simultânea de duas pessoas no desenvolvimento do *software*.
- 81 Na análise de requisitos não funcionais, o tempo de treinamento, o tempo de reinício após falha, o número de sistemas-alvo e a disponibilidade são medidas para, respectivamente, facilidade de uso, robustez, portabilidade e confiabilidade.

No que concerne a análise, projeto e modelagem orientada a objetos, julgue os itens que se seguem.

- 82 Embora não afetem diretamente o que o *software* irá entregar, os requisitos de espaço são classificados, excepcionalmente, como requisitos funcionais em projetos orientados a objetos, especialmente devido à reusabilidade.
- 83 O polimorfismo de objeto pode criar objetos que herdam comportamento de uma classe e pode possibilitar que o desenvolvedor programe um comportamento diferente para os métodos de classes existentes, mas, para que seja da forma sobrescrita, o polimorfismo precisa possuir a mesma assinatura.

A respeito da modelagem de dados e da qualidade de *software*, julgue os itens subsecutivos.

- 84 O relacionamento no modelo entidade-relacionamento é uma associação intuitiva entre entidades, cujo número de entidades envolvidas é conhecido como hierarquia.
- 85 A qualidade do produto de *software* garante a aplicação correta do processo de desenvolvimento, concentrando esforços na busca pela qualidade do modo de criar e de manter o produto final.
- 86 Entidade-relacionamento é uma modelagem semântica cujo modelo resultante é estendido, e as entidades, nesse modelo, são definidas como um ente que pode ser distintamente identificado.

Acerca de arquitetura de *software* e *Domain-Driven Design*, julgue os seguintes itens.

- 87 No padrão em camadas modelo-visão-controle (MVC), o controle é responsável por mudanças de estado da visão.
- 88 Na arquitetura cliente-servidor, organizada como um conjunto de serviços e servidores associados e de clientes que acessam esses serviços, os clientes precisam conhecer seus servidores, mas a reciprocidade não é exigida, o que impossibilita sua implementação em ambientes de Internet.
- 89 *Domain-Driven Design* pode ser aplicada ao processo de concepção arquitetural de um sistema de *software*, sendo que *domain*, em um *software*, designa o campo de ação, conhecimento e influência.

Em relação a arquiteturas de integração, julgue os itens subsequentes.

- 90 Os protocolos de *web services* cobrem aspectos da arquitetura orientada a serviços, como o SOAP, para troca de informações, o WS-BPEL, para linguagem de programação, e o UDDI, para descoberta de serviços, mas excluem aspectos voltados para *workflow* e para transações em sistemas distribuídos.
- 91 A arquitetura orientada a serviços (SOA) é uma forma de desenvolvimento monolítica em que os componentes de sistemas são serviços autônomos baseados em XML.
- 92 A WSDL (*Web Services Description Language*) permite descrever *web services* em XML, tornando-os disponíveis para usuários e programas *web*.

Julgue os próximos itens, relativos a *design patterns*, ECM (*Enterprise Content Management*) e gerenciamento de processos de negócio (BPM).

- 93 ECM define estratégia, métodos e ferramentas empregados para captar, gerenciar e armazenar conteúdo e documentos relacionados aos processos organizacionais.
- 94 O uso de BPM melhora sempre o desempenho nos processos existentes, uma vez que é necessário um conjunto de *softwares* para implantá-lo.
- 95 O padrão de projeto *mediator* visa padronizar a gramática e a interpretação de uma linguagem, ao passo que o padrão *iterator* verifica como os objetos padronizados interagem entre si.
- 96 Uma das disciplinas do *emergent design* refere-se ao uso de padrões de projeto para criar arquiteturas de aplicativos que sejam resilientes e flexíveis.

Acerca de lógica de programação, programação estruturada e programação orientada a objetos, julgue os itens a seguir.

- 97 Classes predefinidas e definidas pelo desenvolvedor são tratadas de formas diferentes pelos compiladores, ainda que a computação seja realizada por meio de passagem de mensagens.
- 98 Tipos de dados estruturados são definidos por meio de operadores de tipos, também conhecidos como construtores, utilizados para formar expressões de tipos.
- 99 Classes podem ter dois tipos de métodos e dois tipos de variáveis. Assim, dois objetos da mesma classe são distinguidos pelo estado de suas variáveis de instância.

No que se refere a conceitos básicos e aplicações de criptografia, protocolos criptográficos e algoritmos, julgue os itens seguintes.

- 100 O comprimento do fluxo de *bits*, na entrada de uma S-box, é igual ao comprimento do fluxo de *bits* resultante, na saída da S-box.
- 101 RSA nomeia um criptossistema usado para a criptografia e a decifração de chaves simétricas, e Diffie-Hellman consiste em um protocolo de estabelecimento e troca de chaves simétricas.
- 102 Se o texto cifrado do texto claro HELLO for ABNZF, então a cifra será monoalfabética.

A respeito de métricas de qualidade de código, código limpo e refatoração, julgue os itens subsequentes.

- 103 O uso de comentários é uma das técnicas de código limpo que, em conjunto com a refatoração de códigos, permite aumentar a produtividade de desenvolvimento de códigos.
- 104 Na classificação das métricas de *software* como quantitativas ou qualitativas, a métrica de pontos de função é tida como qualitativa por ser derivada de uma relação empírica embasada em medidas calculáveis da complexidade do *software*.
- 105 O custo para projetar, codificar e testar o *software* pode ser estimado por meio do uso de ponto de função em dados históricos de sistemas similares.
- 106 O foco das métricas baseadas em herança é a forma como os métodos são sobrecarregados na hierarquia de classe.

Em relação a desenvolvimento orientado a testes (TDD) e a testes automatizados, julgue os próximos itens.

- 107 A execução de testes de unidade automatizados garante a inalterabilidade do comportamento do código.
- 108 Um dos passos executados no ciclo de atividades do processo TDD é a criação de novos testes para as falhas encontradas no código original, sem alteração deste.

Julgue os itens subsequentes, relativos a Delphi.

- 109 Na versão 7 do Delphi, os *drivers* nativos para o Firebird não estão disponíveis, mas é possível acessá-los por meio de componentes *dbExpress*, em cuja paleta a propriedade *DriverName* define o *driver* do banco de dados utilizado.
- 110 O *TdataSource* é um componente não visual da VCL (*visual component library*) utilizado em formulários para conectar um controle e um conjunto de dados usado pelo controle.
- 111 A diretiva `$V` controla a verificação de tipos em sequências curtas, que são passados como parâmetros variáveis.

Julgue os itens que se seguem, que tratam de Java.

- 112 A injeção de dependência e contextos (CDI) é parte integrante do Java EE 6 e fornece uma arquitetura que permite aos componentes do Java EE existirem dentro do ciclo de vida de uma aplicação com escopos bem definidos.
- 113 Embora não sejam objetos, os valores dos tipos primitivos escalares são suportados por Java.
- 114 Interface é um substituto para herança múltipla, uma vez que existe reuso de código na herança múltipla.
- 115 A vinculação estática permite a sobrescrita, se o método for estático ou privado.

Acerca de *frameworks* para camada de apresentação e persistência, julgue os itens seguintes.

- 116 Na injeção de dependência via *setter*, se a dependência não estiver disponível, a classe será criada de forma consistente, dado que as dependências necessárias para o funcionamento do objeto poderão ser passadas ao instanciá-lo.
- 117 O *framework* Cayene pode persistir objetos Java por meio da interface `org.apache.cayenne.Persistent`, a qual provê acesso às propriedades `objectId`, `persistenceState` e `objectContext`.

No que se refere à programação *web* e a sistemas de controle de versão, julgue os itens subsequentes.

- 118** O atributo `ngBind` informa ao AngularJS para atualizar o conteúdo do texto, quando o valor da expressão for alterado.
- 119** As *borders*, desenhadas ao redor dos elementos depois de estes serem dispostos, diferentemente dos *outlines*, participam do leiaute.
- 120** O *Subversion*, que oferece diversos algoritmos de diferenciação de três vias para manipulação de dados até uma granularidade de caracteres individuais, não permite complementar o processamento de fusão de conteúdo com utilitários de diferenciação externos.

Espaço livre
