

CONHECIMENTOS ESPECÍFICOS

Uma metodologia de desenvolvimento de *software* pode ser classificada como uma metodologia ágil quando efetua o desenvolvimento do *software* de forma incremental (libera pequenas versões, em iterações de curta duração) e é colaborativa (cliente e desenvolvedores trabalham juntos, em constante comunicação), direta (o método em si é simples de aprender e modificar) e adaptativa (capaz de responder eficientemente às mudanças).

Considerando a definição acima, de Abrahamsson, julgue os itens a seguir, a respeito das metodologias ágeis de desenvolvimento de *software*.

- 51 No *Scrum*, o *taskboard* é um grande painel onde podem ser colocadas informações importantes para o acompanhamento do *sprint*. Por meio do *taskboard*, as informações acerca das atividades não iniciadas (*sprint backlog*), das que estão em andamento e das concluídas ficam sempre visíveis e disponíveis para todos os interessados no projeto.
- 52 No desenvolvimento de *software*, a geração automática de código, proporcionada por várias ferramentas de desenvolvimento, é um fator que aumenta a produtividade, melhora a padronização e diminui os defeitos no produto.
- 53 É conveniente que o contrato, entre cliente e fornecedor, para o desenvolvimento de um sistema computacional, contenha a lista de requisitos para o *software*. Contudo, os métodos ágeis de desenvolvimento preconizam que o referido contrato estabeleça o preço, a ser pago pelo cliente, com base no tempo necessário para o desenvolvimento do sistema e não com base no conjunto de requisitos.
- 54 O que os métodos ágeis buscam é como evitar as mudanças desde o início do projeto e não a melhor maneira de tratar essas mudanças.
- 55 Considerando o conceito de metodologia ágil em apreço, é correto afirmar que as seguintes metodologias são ágeis: XP (*Extreme Programming*), *Scrum*, *Crystal*, FDD (*Feature Driven Development*), DSDM (*Dynamic Systems Development Method*) e *Open Source Software Development*.
- 56 O XP segue um conjunto de valores, princípios e regras básicas que visam alcançar eficiência e efetividade no processo de desenvolvimento de *software*. Os valores são cinco: comunicação, simplicidade, *feedback*, coragem e respeito.

O RUP (*Rational Unified Process*) é um processo de desenvolvimento de *software*. Nesse processo, são descritos os papéis e as atividades que cada membro da equipe de projeto deve desempenhar ao longo do ciclo de desenvolvimento do *software* e os produtos que devem ser gerados como resultado dessas atividades, os chamados artefatos. Com relação a esse assunto, julgue os itens seguintes.

- 57 O RUP tem duas dimensões: o eixo horizontal e o eixo vertical. A primeira dimensão representa o aspecto estático do processo quando ele é aprovado e é expressa em termos de fases, iterações e marcos. A segunda dimensão representa o aspecto dinâmico do processo, como ele é descrito em termos de componentes, disciplinas, atividades, fluxos de trabalho, artefatos e papéis do processo.
- 58 Um projeto que usa o desenvolvimento iterativo tem um ciclo de vida que consiste em várias iterações. Uma iteração incorpora um conjunto quase sequencial de atividades em modelagem de negócios, requisitos, análise e *design*, implementação, teste e implantação, em várias proporções, dependendo do local em que ela está localizada no ciclo de desenvolvimento.
- 59 Na denominada fase de elaboração, tipicamente, o foco é inserido na maneira como se gerenciam requisitos e como se gerencia o projeto.

Gerenciamento de processos de negócio (BPM) é uma abordagem disciplinada para identificar, desenhar, executar, documentar, medir, monitorar, controlar e melhorar processos de negócio, automatizados ou não, para alcançar os resultados pretendidos consistentes e alinhados com as metas estratégicas de uma organização. Acerca dessas informações, julgue os itens que se seguem.

- 60 No contexto do BPM, um processo é um conjunto definido de atividades ou comportamentos executados por humanos ou máquinas para alcançar uma ou mais metas. Os processos possuem atributos e características que descrevem propriedades, comportamento, propósito, ou outros elementos de processo.
- 61 Um diagrama de processo retrata uma notação conveniente do fluxo de trabalho básico de um processo. Modelagem agrega maior detalhe sobre o processo e os relacionamentos com outros elementos. Mapeamento representa o desempenho do que está sendo modelado.
- 62 Modelos de processo são úteis como uma documentação, um meio para comunicação e alinhamento, desenho e requisitos, ou um meio para analisar aspectos de processo, treinamento e explicação.

A engenharia de requisitos (ER) é uma subárea da engenharia de *software* que estuda o processo de produção e gerência dos requisitos que o *software* deverá atender. Esse processo tem início junto aos clientes e perpassa todas as fases do processo de desenvolvimento do *software*. O objetivo da ER é fornecer métodos, técnicas e ferramentas que forneçam suporte adequado às tarefas de produção e gerência dos requisitos do sistema. Levando em conta esses aspectos, julgue os próximos itens.

- 63 No processo de construção e(ou) manutenção de um produto de *software*, o termo requisito pode ser definido da seguinte forma: uma condição, característica ou capacidade, determinada no universo das necessidades do negócio do usuário, que deve ser atendida por um *software* na forma de aspectos funcionais e não funcionais.
- 64 Uma das principais técnicas de verificação é a prototipação. Um protótipo é um produto parcialmente desenvolvido, que possibilita aos clientes e desenvolvedores examinarem certos aspectos do sistema proposto e decidir se eles são ou não apropriados ou adequados para o produto acabado.
- 65 O principal artefato elaborado no processo de produção de requisitos do sistema, segundo a ER, é o documento de requisitos. Por sua vez, o documento de requisitos é uma declaração formal dos requisitos para os *stakeholders*, que podem ser clientes, usuários finais ou a equipe de desenvolvimento do *software*.
- 66 Gerência de requisitos pode ser conceituada como enfoque sistemático para a elicitação, organização e documentação dos requisitos do sistema e um processo que estabelece e mantém o acordo entre usuários e a equipe de projeto à medida que os requisitos são modificados.

Julgue os itens seguintes, relativos a testes de *software* e gerência de projeto.

- 67 No contexto dos requisitos e restrições de um projeto, o propósito da gerência de projetos é identificar, estabelecer, coordenar e monitorar as atividades, tarefas e recursos de que um projeto necessita para produzir um produto.
- 68 No contexto da gerência do projeto, integração inclui características como unificação, consolidação, articulação e ações de integração que são cruciais para concluir o projeto, atender satisfatoriamente os requisitos dos interessados e clientes e gerenciar as expectativas.
- 69 Nas atividades de desenvolvimento, a validação refere-se ao processo de examinar o resultado de uma atividade para determinar sua conformidade com os requisitos estabelecidos para a mesma atividade, enquanto a verificação se refere ao processo de examinar um produto para determinar sua conformidade com as necessidades do usuário.
- 70 Revisão por pares e testes são considerados métodos de validação. As informações obtidas durante as revisões são extremamente úteis para os testes, pois permitem a identificação dos módulos críticos e pensos a erros.

Tendo em vista que, na UML (*Unified Modeling Language*), a utilização de diversos diagramas possibilita descobrir falhas não detectadas em diagramas gerados anteriormente, diminuindo a possibilidade da ocorrência de erros durante a fase de desenvolvimento do *software*, julgue os itens a seguir.

- 71 O diagrama de sequência baseia-se no diagrama de caso de uso e no diagrama de classes. O diagrama de sequência é considerado uma excelente forma de validar o diagrama de classes.
- 72 O diagrama de comunicação concentra-se em como os objetos estão vinculados e quais mensagens trocam entre si durante o processo. Esse diagrama está associado ao diagrama de classe.
- 73 Estereótipos são uma maneira de destacar ou diferenciar um componente ou relacionamentos iguais, atribuindo-lhes características especiais ou modificando-as de alguma forma.
- 74 O diagrama de estrutura composta é similar ao denominado diagrama de classes, porém este último apresenta uma visão estática da estrutura de classes, enquanto o primeiro tenta expressar arquiteturas de tempo de execução.

Considerando a organização e arquitetura de computadores, julgue os itens que se seguem.

- 75 São componentes de alto nível de um computador: contador de programa, registrador de instruções, registrador de endereçamento à memória, registrador temporário de dados, registrador de endereçamento de entrada/saída (E/S) e registrador de armazenamento temporário de dados de E/S.
- 76 A aritmética computacional geralmente opera com dois tipos de números: inteiros e de ponto flutuante. Números de ponto flutuante são expressos na forma de um número multiplicado por uma constante elevada a uma potência inteira e seu uso é restrito à representação de números de valores muito elevados.
- 77 O termo arquitetura de um computador refere-se aos atributos de um sistema que são visíveis para o programador ou, em outras palavras, aos atributos que têm impacto direto sobre a execução lógica de um programa. O termo organização de um computador refere-se às unidades operacionais e suas interconexões que implementam as especificações de sua arquitetura.
- 78 São funções básicas de um computador: processamento de dados, armazenamento de dados, transferência de dados e controle. São componentes estruturais de um computador: unidade central de processamento, memória principal, dispositivos de entrada e saída e sistemas de interconexão.
- 79 Uma função importante de um sistema operacional é o escalonamento de processos ou tarefas. O escalonamento a curto prazo faz parte da função de troca de processos (*swapping*) entre a memória principal e a memória secundária do computador.

Julgue os itens a seguir, referentes a sistemas de entrada, saída e armazenamento em computadores.

- 80 As funções mais importantes de um módulo de E/S podem ser divididas nas seguintes categorias: controle e temporização, comunicação com o processador, comunicação com dispositivos, área de armazenamento temporário de dados e detecção de erros.
- 81 As principais técnicas de entrada/saída (E/S) utilizadas em computadores são: E/S programada, E/S dirigida por interrupção e acesso direto à memória. A última é efetuada sob controle direto e contínuo do programa que requisitou a operação de E/S.

Com relação aos fundamentos de redes de computadores, julgue o item abaixo.

- 82 Três importantes componentes do correio eletrônico são os agentes usuários, os servidores de *email* e o protocolo SMTP (*Simple Mail Transfer Protocol*).

A respeito dos ambientes GNU/Linux e Windows, julgue os itens subsequentes.

- 83 É possível utilizar um programa para Windows no ambiente GNU/Linux. Isso pode ser feito instalando-se o programa em uma API Windows para GNU/Linux ou por meio da virtualização — isto é, usando-se um programa que emule um *hardware*.
- 84 Para acessar um sistema GNU/Linux é necessária uma conta, composta por um *login* de acesso e sua respectiva senha. Há basicamente três modalidades de usuários. Uma delas é o usuário de sistema, que somente pode realizar tarefas autorizadas pelo administrador (*root*).
- 85 Ao se configurar um computador novo, antes de instalar o sistema operacional, é necessário selecionar o sistema de arquivos para o *hard disk* (HD) (processo conhecido como formatação). No Windows, quando se formata um HD, é possível escolher entre dois tipos de sistema de arquivos: o FAT (*File Allocation Table*) e o NTFS (*New Technology File System*). O sistema operacional GNU/Linux é compatível com esses dois sistemas de arquivo.

Acerca de pseudocódigo, descrição narrativa e fluxograma, que são usados para a descrição de algoritmos, julgue os próximos itens.

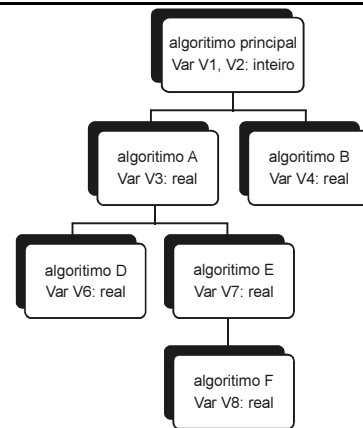
- 86 Para especificar os passos de um algoritmo, o pseudocódigo utiliza uma linguagem natural com característica pouco formal, o que pode causar ambiguidade e propiciar interpretações errôneas.
- 87 Uma forma universal de representação de um algoritmo, o fluxograma, ou diagrama de blocos, utiliza figuras geométricas a fim de ilustrar os passos a serem seguidos para a resolução de um problema.
- 88 A descrição narrativa é uma forma de algoritmo que utiliza linguagem estruturada e, embora sua estrutura e simbologia apresentem pequenas diferenças em relação à língua portuguesa, também é chamada de português estruturado.

Julgue os itens que se seguem, a respeito de programação orientada a objetos.

- 89 A herança representa uma generalização, dessa forma, por meio de herança é feito o compartilhamento de atributos e operações entre classes, com base em um relacionamento hierárquico.
- 90 Quando vários objetos, identificados com atributos e operações semelhantes, são agrupados em uma classe, ocorre o processo de especialização. Por outro lado, quando um objeto constituir-se em uma classe de objetos, da qual podem ser derivados outros objetos, ocorre a generalização.
- 91 Na programação orientada a objetos, uma classe é uma coleção de objetos que podem ser descritos por um conjunto básico de atributos e possuem operações semelhantes.

Julgue os itens subsequentes, referentes aos módulos que constituem a técnica de modularização, utilizada para desenvolver algoritmos.

- 92 Em Java, os módulos, sejam eles procedimentos ou funções, são representados pelas classes, nas quais a instrução `return` indica que o controle do fluxo de dados deverá retornar ao algoritmo principal.
- 93 Um procedimento ou sub-rotina é um conjunto de instruções que realiza determinada tarefa. Um algoritmo de procedimento, como qualquer outro algoritmo, deve ser identificado e pode possuir variáveis, operações e até funções.
- 94 De forma semelhante aos procedimentos, as funções podem ser utilizadas em expressões, como se fossem variáveis, ou seja, elas retornam valores que são associados ao seu nome; portanto, é necessário declarar que tipo de dados será retornado.



Sandra Puga. *Lógica de programação e estrutura de dados*. São Paulo: Pearson Prentice Hall, 2009 (com adaptações).

Considerando a figura acima, que ilustra um diagrama representando a hierarquia para a resolução de um problema em módulos, julgue os itens a seguir.

- 95 A variável V3, declarada no algoritmo A, pode ser utilizada pelos algoritmos A, B, D, E e F.
- 96 As variáveis V6 e V8 podem ser utilizadas pelos respectivos algoritmos em que foram declaradas e pelo algoritmo A.
- 97 A variável V7, declarada no algoritmo E, pode ser utilizada pelos algoritmos E, F e D.
- 98 As variáveis V1 e V2, que foram declaradas no algoritmo principal, podem ser utilizadas por todos os algoritmos representados no diagrama apresentado.

A respeito de estruturas de controle de fluxo em algoritmos, julgue os próximos itens.

- 99 A estrutura de seleção de múltipla escolha permite a execução de determinadas instruções, de acordo com a opção escolhida. Na linguagem Java, a palavra reservada `break` é utilizada para garantir que apenas a instrução selecionada seja executada.
- 100 Nas estruturas de repetição, que são utilizadas quando se deseja repetir certo trecho de instruções, o número de repetições deve ser conhecido ou determinado previamente e precisa ser finito.
- 101 Na linguagem Java, a expressão `contador ← contador + 1` é representada, usando-se o operador de incremento `++`, por `contador ++`.
- 102 A estrutura de repetição `para` utiliza variáveis de controle que definem exatamente o número máximo de vezes que a sequência de instruções será executada.

Acerca das estruturas de dados estáticas e homogêneas (vetores e matrizes) e das estruturas de dados dinâmicas (listas, pilhas, filas), utilizadas para armazenar conjuntos de valores, julgue os itens a seguir.

- 103 As pilhas, também conhecidas como listas LIFO ou PEPS, são listas lineares em que todas as operações de inserção e remoção de elementos são feitas por um único extremo da lista, denominado topo.
- 104 Vetores são utilizados quando estruturas indexadas necessitam de mais que um índice para identificar um de seus elementos.
- 105 Uma lista é uma coleção de elementos do mesmo tipo dispostos linearmente, que podem ou não seguir determinada organização. As listas podem ser dos seguintes tipos: de encadeamento simples, duplamente encadeadas e ordenadas.

Julgue os itens subsequentes, relativos a comandos SQL (*structured query language*).

- 106 O comando a seguir permite criar a tabela `ALUNO` inserindo as restrições de integridade. Os dados da tabela e as suas restrições podem ser alteradas utilizando-se o comando `ALTER TABLE`.

```
CREATE TABLE ALUNO
(
  Matricula smallint not null,
  Nome char(20),
  Salario money,
  Cidade char(20),
  CEP smallint
  Primary key (matricula)
);
```

- 107 O comando `drop table aluno` pode ser utilizado para eliminar a tabela `aluno`, seus dados e suas referências com outras tabelas.
- 108 O comando `Select * from aluno` permite exibir, na tela do computador, todas as colunas da tabela `aluno`.
- 109 O comando a seguir permite exibir, na tela do computador, o nome, a cidade e o CEP dos alunos constantes da tabela `aluno` que moram na cidade de Brasília e cujo endereço tenha o CEP maior que 70000.


```
Select nome from aluno where cidade = "Brasília" and CEP > 70000
```
- 110 O comando a seguir permite exibir, na tela do computador, os dados de todos os alunos constantes da tabela `aluno` que recebam salário maior que 2000.


```
Select count (*) from aluno where salário > 2000
```

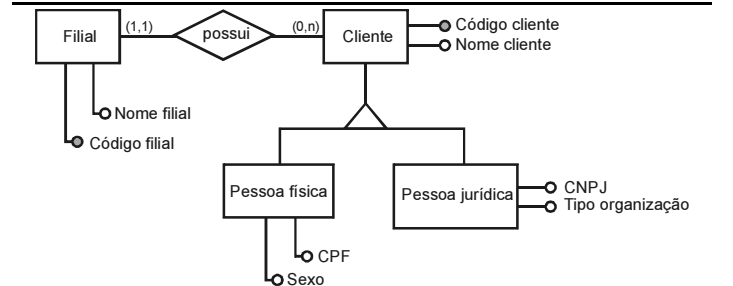
A respeito dos modelos conceitual e lógico, normalmente considerados no projeto de banco de dados, julgue os seguintes itens.

- 111 A seguir, apresenta-se uma representação de forma textual no modelo conceitual.


```
Aluno(matricula, nome, salário, cidade, CEP)
```
- 112 No modelo conceitual, o banco de dados é descrito de forma independente da implementação em um SGBD. Esse modelo cita os dados que podem ser armazenados no banco de dados, mas não especifica como esses dados estão armazenados no nível de SGBD.
- 113 No modelo lógico, que depende do tipo de SGBD em uso, o banco de dados é descrito no nível de abstração visto pelo usuário do SGBD.

Considerando que um SGBD é um pacote de *software* para a implementação e manutenção de bancos de dados computacionais, julgue os itens a seguir.

- 114 Considerando-se bases de dados muito grandes, o MySQL é mais rápido que o PostgreSQL; entretanto, o PostgreSQL oferece uma série de recursos extras que o tornam especializado em operações complexas.
- 115 O SQL Server oferece os seguintes tipos de integridade de dados: integridade de domínio, integridade de entidade e integridade referencial.
- 116 PostgreSQL e MySQL são exemplos de SGBD que executam em ambiente Linux e Windows.



Considerando o diagrama ilustrado na figura acima, julgue os itens que se seguem.

- 117 De acordo com o diagrama ilustrado, uma entidade `Filial` deve ter pelo menos um `Cliente` vinculado a ela.
- 118 De acordo com o diagrama apresentado, a entidade `Cliente` poderá estar vinculada a uma ou nenhuma `Filial`, ou seja, poderão existir clientes sem estarem vinculados a nenhuma `Filial`.
- 119 No diagrama apresentado, o triângulo representa uma generalização/especialização e indica que a entidade `Cliente` é dividida em dois subconjuntos, as entidades `Pessoa física` e `Pessoa jurídica`. No conceito de generalização/especialização, a ideia de dependência de propriedades significa que cada ocorrência da entidade especializada possui, além de suas propriedades, as propriedades da entidade genérica correspondente.
- 120 No modelo representado no diagrama acima, a implementação de hierarquias de generalização na abordagem relacional pode ser feita usando-se uma tabela para cada entidade ou uma única tabela para toda a hierarquia de generalização/especialização.