

CONHECIMENTOS ESPECÍFICOS

Acerca de *web services*, e-ping, XML e XSLT, julgue os itens a seguir.

- 61 De acordo com a e-ping, todos os sistemas necessitam ter capacidade de se comunicar diretamente em XML, pois essa linguagem permite armazenar dados utilizando elementos, *tags* e atributos.
- 62 A arquitetura orientada a serviço constitui um modelo arquitetônico que visa aumentar a agilidade e melhorar a relação custo/benefício de uma organização com referência à implantação de sistemas interoperáveis. Esse modelo tem como princípio a disponibilização de unidades lógicas de solução, em que a orientação a serviços tem sido aplicada de forma significativa.
- 63 *Web service* é uma solução lógica que oferece a possibilidade de recuperar informações remotamente por meio de *cloud computing*. Esse serviço utiliza a definição WSDL em um ou mais XML Schemas, que são armazenados na UDDI.
- 64 Um *web service* pode ser desenvolvido, também, com o uso de REST, que utiliza o protocolo HTTP para comunicação entre emissor e destinatário, e o SOAP, para encapsular as mensagens trafegadas.
- 65 Referenciada na e-ping, a XSLT é uma linguagem que transforma documentos XML em outros documentos XML, o que permite o intercâmbio de informações e a interoperabilidade entre sistemas.

A respeito de gerenciamento de conteúdo e E-MAG, julgue os itens subsequentes.

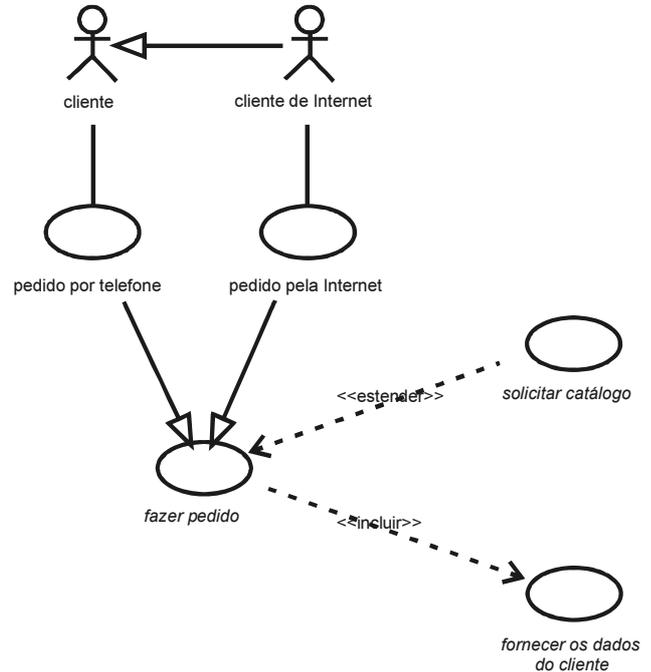
- 66 As etapas do processo de acessibilidade a um sítio, preconizadas pela E-MAG, incluem a verificação da necessidade de acessibilidade, a validação e promoção de acessibilidade e a garantia contínua da acessibilidade.
- 67 Sistemas de gerenciamento de conteúdo, quando desenvolvidos com ferramentas livres, não mostram conteúdos diferentes, independentemente de o usuário estar ou não autenticado, pois isso feriria o princípio da arquitetura *web* e das liberdades preconizadas para esse tipo de *software*.
- 68 Em relação ao uso, a E-MAG identifica quatro situações principais relacionadas ao acesso a computador por parte de usuários portadores de necessidades especiais: acesso ao computador sem teclado, sem *mouse*, sem monitor e sem áudio.

Julgue os itens seguintes, relativos a processos de *software* e a sistemas orientados a objetos (OO).

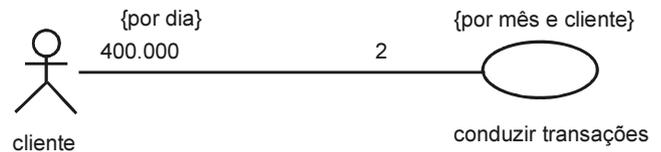
- 69 O SCRUM é um método ágil em que todos os itens do *sprint backlog* advêm do *product backlog*.
- 70 O desenvolvimento de um código na Extreme Programming está relacionado à fase de planejamento, pois, nessa metodologia, não há fase de desenvolvimento, haja vista que a codificação é realizada em pares.
- 71 Segundo o conceito de orientação a objetos, uma classe é um conjunto de objetos em comum, e um objeto é uma instância de uma classe.
- 72 Em uma análise de sistemas OO, devem ser verificados eventos que interajam com os objetos, tendo-se como referência a recomendação de o projeto de *software* ser construído com ênfase na reusabilidade de classes de objetos existentes e, quando necessário, em novas classes. Nesse sentido, a reusabilidade é uma das vantagens da orientação a objetos, por permitir que o arquiteto de *software* trabalhe, se necessário, em padrão conceitual único durante todo o processo de criação.
- 73 O diagrama de caso de uso de negócio é um diagrama do RUP utilizado para mapear e descrever atores e funções envolvidos na modelagem de negócio.

No que se refere a UML, julgue os próximos itens.

- 74 O diagrama de caso de uso ilustrado abaixo está incorreto, pois não há herança entre *use case*, apesar de haver entre os atores.



- 75 É possível indicar multiplicidade em diagramas de caso de uso. O diagrama ilustrado abaixo indica que o caso de uso conduzir transações é utilizado 400.000 vezes por dia pelos clientes. Cada cliente inicia o caso de uso duas vezes ao mês.



- 76 Em um diagrama de implantação com componentes, é possível indicar o protocolo na ligação entre dois nós, bem como mostrar relações de dependência entre componentes.
- 77 Considerando que C2 e C3 sejam classes que herdem de C1, é possível que um método M1 em C1 também possa estar presente na classe C2. Assim, essa operação pode ser implementada por meio de polimorfismo com a sobrescrita do método e, nesse caso, os parâmetros e, conseqüentemente, a assinatura de M1 em C2 serão diferentes dos de M1 da classe C1.
- 78 Uma agregação composta presente entre duas classes indica uma associação do tipo todo parte, em que uma classe é a parte e a outra, o todo.
- 79 O diagrama de sequência descreve as mudanças de estado ou condição de uma instância de uma classe na troca de mensagens entre os objetos.

Acerca de métricas e arquiteturas de sistemas, julgue os itens que se seguem.

- 80 *Web service* é um exemplo de arquitetura orientada a serviços que, de acordo com suas características, está mais bem adequada à arquitetura cliente/servidor do que à monolítica.
- 81 A arquitetura de *software* abrange, entre outros aspectos, questões estruturais como estrutura de controle, protocolos de comunicação, sincronização e acesso a dados, composição de elementos de *design*, além de escalonamento e desempenho do sistema.
- 82 A identificação dos ALIs somente é possível após o desenvolvimento do modelo relacional do sistema, no qual são descritos as entidades, os atributos e os relacionamentos.
- 83 A arquitetura cliente/servidor enseja o desenvolvimento de um sistema com, no máximo, duas camadas, quais sejam, cliente e servidor.
- 84 A APF tem fórmulas específicas para projetos novos e para manutenção de legados, sendo compatível até mesmo para contagem de sistemas transacionais de grande porte.

Com referência a linguagens de programação, algoritmos e estruturas de dados, objetos e Java, julgue os itens de 85 a 89.

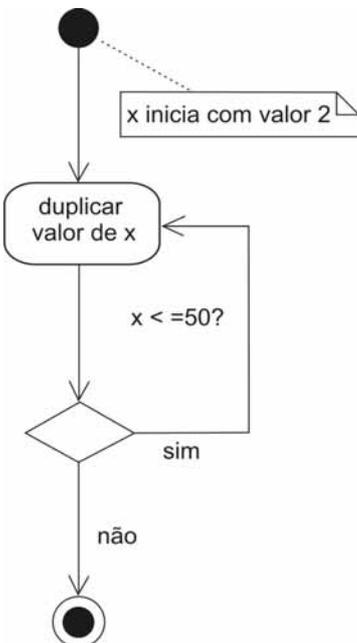
- 85 `int`, `byte`, `double` e `char` são alguns tipos primitivos de variáveis suportadas pelo Java. As linhas abaixo declaram, corretamente, `var1`, `var2` e `var3` como sendo do tipo `int`.

```
int var1;
int var2;
int var3;
```

 Outra forma também correta para a mesma declaração seria a que se segue `int var1, var2, var3;`

- 86 O fluxo de controle correspondente ao diagrama de atividades mostrado na figura abaixo pode ser descrito corretamente em Java da forma seguinte.

```
int x = 2;
while (x <= 50)
    x*=2;
```



- 87 Considerando que as linhas abaixo tenham sido extraídas de um programa Java, é correto afirmar que as linhas 10 e 20 declaram as variáveis `Jdbc` e `Jurl`; a linha 30 carrega a classe do `MySQL` e, na linha 40, o método `getConnection` recebe da classe `DriverManager` a URL com usuário e senha para conexão ao banco de dados bd.

```
10 static final string xpto = "com.mysql.jdbc.driver";
20 static final string xyz = "jdbc:mysql://localhost/bd";
30 Class.forName (xyz);
40 Conn = DriverManager.getConnection (xpto,
"usuario", "senha")
```
- 88 `AWT` é um conjunto de componentes GUI, integrante da arquitetura `JEE6`, constante no pacote `java.awt`, desenvolvidas para substituir as GUIs do `SWING`.
- 89 `Applets Java` são interfaces gráficas com o usuário em que é possível se fazer a inserção de componentes GUI que podem ser incorporados a páginas `HTML`.

Julgue os itens que se seguem, acerca de qualidade de *software*, linguagens de programação e padrões de projetos.

- 90 O padrão *Singleton* garante que uma classe tenha somente uma instância, fornecendo, assim, um ponto global de acesso a essa instância.
- 91 O padrão *Facade* assemelha-se ao *Mediator* no aspecto em que este abstrai um subsistema de objetos a fim de fornecer uma interface encapsulada na forma como um conjunto de objetos interage entre si.
- 92 O padrão *Prototype* pode ser usado no desenvolvimento de programas escritos com a linguagem `PHP 5.0`, atuando como padrão estrutural que permite construir tanto classes quanto objetos.
- 93 O `CMMI` (*Capability Maturity Model Integration*), desenvolvido pelo `SEI` (`Software Engineering Institute`), visa mensurar a maturidade dos processos de desenvolvimento da organização, auxiliando na garantia e melhoria da qualidade de seus produtos e serviços na área de tecnologia da informação.
- 94 Nos programas desenvolvidos em `Delphi` sob o paradigma da orientação a objetos, é permitido, por meio de ligação dinâmica, implementar polimorfismo e, assim, desacoplar objetos entre si.
- 95 Os padrões de projeto são úteis tanto na fase de planejamento da arquitetura quanto na de desenvolvimento e codificação.
- 96 O padrão *Abstract Factory* é corretamente aplicável, quando necessário, para fornecer uma biblioteca de classes e não revelar suas interfaces.

tabela cliente

fname	lname	age
Paulo	Silva	53
Dara	Carvalho	15
Fátima	Silva	57

Tendo como referência a tabela acima, julgue os itens seguintes, referente a SQL.

- 97** O comando SQL mostrado abaixo, construído a partir dos dados da tabela cliente e executado no Oracle 10g, retornará apenas um registro.

```
SELECT fname FROM cliente WHERE SAL ^= 53
```

- 98** No Oracle 10g, é possível combinar, em uma mesma instrução SELECT do SQL, os argumentos HAVING e WHERE.

- 99** O comando SQL mostrado abaixo, construído a partir dos dados da tabela cliente e executado no PostgreSQL 8.2, retornará apenas um registro, concatenando os atributos fname e lname.

```
SELECT fname+lname FROM cliente WHERE age >56
```

- 100** O comando SQL mostrado abaixo, se executado no SQL SERVER 2008 R2, permite que se adicione uma coluna, col2, à tabela cliente.

```
ADD COLUMN col2 int NOT NULL ON TABLE cliente
```

- 101** Os comandos SQL mostrados abaixo, executados no SQL Server 2008 R2, permitem criar duas tabelas temporárias: #temp1 e ##temp2, respectivamente, dos tipos local e global.

```
CREATE TABLE #temp1 (col1 INT PRIMARY KEY)
```

```
GO
```

```
CREATE TABLE ##temp2 (col1 INT PRIMARY KEY)
```

- 102** O operador ROLLUP, quando combinado ao comando SQL SELECT com GROUP BY, no SQL Server 2008 R2, permite obter totais parciais e geral nos agrupamentos de colunas do tipo text.

- 103** Em um comando SQL, a cláusula DISTINCT faz que sejam retornados apenas atributos exclusivos na consulta.

A respeito de algoritmos, estruturas de dados e objetos, julgue os itens que se seguem.

- 104** A função mostrada abaixo é um exemplo de algoritmo recursivo.

```
Function fat (x:integer) : integer;
begin
    if x=0 then cfat := 1
    else
        fat := x*cfat (n-1);
    end;
```

- 105** O algoritmo mostrado a seguir resume corretamente a inserção com um leque de, no máximo, sete elementos.

```
Type leque = record
    topo: integer;
    memo: array [1..7] of char;
end;

var L:leque;
p.topo: = 0
procedure push (var L:leque; x:char);
begin
    if not L.topo>7 then
        begin
            L.topo :=L.topo+1;
            L.memo [L.topo]:= x;
        end;
    end;
```

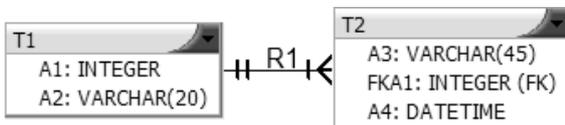
- 106** A estrutura de dados aplicada às linguagens de programação visa, do ponto de vista teórico, identificar e desenvolver modelos matemáticos que resolvam problemas e, do prático, criar representações concretas dos objetos e desenvolver rotinas capazes de atuar sobre essas representações, de acordo com o modelo considerado.

- 107** Pilha é uma estrutura de dados do tipo lista linear, em que as operações TOP, PUSH e POP são realizadas no topo da pilha e, por isso, são denominadas LIFO.

- 108** Árvore de dados é uma estrutura formada por uma coleção finita de dados, em que um deles é denominado raiz e os demais, folhas. Por serem estruturas disjuntas, não é possível que a árvore seja nula.

Acerca de banco de dados, OLAP, normalização e MVC, julgue os itens subsequentes.

- 109 Devido à necessidade de atualização sincronizada e imediata dos dados com o ambiente transacional, o planejamento de uma arquitetura OLAP deve ser combinado com a arquitetura MVC, privilegiando o controlador em detrimento do modelo.
- 110 A abordagem MVC pode ser utilizada como arquitetura de sistemas desenvolvidos em linguagem Delphi. Essa abordagem é composta de três objetos: modelo — o objeto da aplicação —; a visão — a apresentação na tela —; e o controlador — define a maneira como a interface do usuário reage às suas entradas.
- 111 Considere que as classes do diagrama de classes X tenham sido desenhadas a partir de um diagrama relacional Y. Nesse caso, no processo de criação das classes, as entidades de Y que se encontrarem na terceira forma normal deverão ser normalizadas para a forma de Boyce Codd, uma vez que os diagramas de classes não suportam a terceira forma normal.
- 112 Para garantir que haja visão abstrata do banco de dados, o SGBD possui três níveis de abstração. No nível de abstração de usuário, são definidos quais os dados estão armazenados e quais são os seus relacionamentos.
- 113 Em um projeto de banco de dados, o modelo conceitual e o modelo lógico definem as entidades e seus relacionamentos. O primeiro modelo independe do tipo de SGBD e o segundo depende do tipo de SGBD a ser utilizado.
- 114 Os atributos, propriedades que descrevem uma entidade, são considerados uma função que mapeia essas entidades em domínio.
- 115 Para que uma relação X esteja na segunda forma normal, é imperativo que não possua grupos de repetição. Dessa forma, ela deve conter apenas valores atômicos e todos os atributos não chaves devem ser dependentes da totalidade da chave primária.
- 116 Em uma arquitetura OLAP em que haja enorme volume de processamento de dados, é tecnicamente inviável inserir dispositivos móveis, mesmo para clientes e usuários, uma vez que essa arquitetura é voltada para processamento massivo de dados com milhares de registros.



Tendo como referência o modelo de dados acima ilustrado, em que A1 é chave primária da tabela T1 e FKA1 é chave estrangeira entre T1 (A1) e T2 (FKA1), julgue os itens que se seguem, acerca de abstração e modelo de dados.

- 117 Se R1 for um relacionamento do tipo identificado, FKA1 será, obrigatoriamente, parte da chave primária T2.
- 118 Se A4 for um atributo derivado, T2 estará na primeira forma normal.
- 119 O modelo apresentado está mais bem relacionado a um modelo lógico que a um modelo físico.

Julgue os itens seguintes, relativos a tecnologias associadas a Java.

- 120 JSF é uma tecnologia que incorpora características MVC para arquitetura *web*, separando visualização (*view*) e regras de negócio (modelo). Desse modo, a JSF estabelece padrão para a construção de interfaces de usuário do lado do servidor.
- 121 JNDI (*Java Naming and Directory Interface*) é um *framework* de desenvolvimento Java que unifica e padroniza classes e interfaces para acesso a diretórios baseados em LDAP.
- 122 O Jboss Seam é um *framework* compatível com AJAX, EJB3, JSF, Portlets Java e gerenciamento de processos de negócios.
- 123 Pacotes RMI pertencentes ao `java.rmi` permitem visualizar dados entre redes, uma vez que possibilitam que objetos Java se comuniquem, mesmo quando executados em JVM distintas, pela chamada de métodos remotos.
- 124 A JSP possui quatro componentes: chaves diretivas, ações, elementos de *script* e bibliotecas de *tags*. As ações são mensagens para o contêiner de JSP e os elementos de *script* permitem aos navegadores inserir códigos Java que interajam com os componentes JSP.
- 125 A JSP permite que se trabalhe com objetos implícitos e assim se obtenham capacidades de *servlet* em seu contexto. Um exemplo de objeto implícito é o *application* do `javax.servlet.ServletContext`, que representa o contêiner em que o JSP é executado.
- 126 A JSP é uma extensão da tecnologia dos *servlets* que permite simplificar o processo de criação de páginas, separando a apresentação do conteúdo.
- 127 A tecnologia EJB (*Enterprise JavaBeans*), arquitetura de componentes do lado do servidor, permite o desenvolvimento rápido e simplificado de aplicações transacionais, seguras e portáteis, baseadas na tecnologia Java. Seu objetivo é facilitar o trabalho do desenvolvedor para que ele não tenha de se preocupar com aspectos de infraestrutura.

A respeito de portais corporativos, bancos de dados textuais e tipos de dados, julgue os itens a seguir.

- 128 Diferentemente dos bancos de dados relacionais, os textuais não possuem índices para recuperação da informação, haja vista não terem campos específicos. Em vez de índices, esses bancos de dados possuem *stopwords*, que são palavras-chaves utilizadas como base nas consultas de dados.
- 129 *Portlets* são componentes de portais corporativos que oferecem conteúdo, colaboração e também informações transacionais.
- 130 Procedimentos e funções podem receber parâmetros, sendo mais comum a passagem de parâmetros às funções que permitem recuperar o retorno (resultado) após a execução de sua chamada.