

ENADE

Comentado

ANA PAULA TERRA BACELO
MARCELO HIDEKI YAMAGUTI
(Organizadores)

SISTEMAS DE INFORMAÇÃO

2014



ENADE

Comentado

**ANA PAULA TERRA BACELO
MARCELO HIDEKI YAMAGUTI**
(Organizadores)

SISTEMAS DE INFORMAÇÃO

2014





Pontifícia Universidade Católica
do Rio Grande do Sul

Chanceler

Dom Jaime Spengler

Reitor

Evilázio Teixeira

Vice-Reitor

Jaderson Costa da Costa

CONSELHO EDITORIAL

Presidente

Carla Denise Bonan

Diretor da EDIPUCRS

Gilberto Keller de Andrade

Editor-Chefe

Jorge Campos da Costa

Beatriz Correa P. Dornelles

Carlos Alexandre Sanchez Ferreira

Carlos Eduardo Lobo e Silva

Eleani Maria da Costa

Leandro Pereira Gonçalves

Luciano Aronne de Abreu

Newton Luiz Terra

Sérgio Luiz Lessa de Gusmão

ENADE

Comentado

ANA PAULA TERRA BACELO
MARCELO HIDEKI YAMAGUTI
(Organizadores)

SISTEMAS DE INFORMAÇÃO

2014



ediPUCRS

Porto Alegre, 2017

© EDIPUCRS, 2017

CAPA: RODRIGO BRAGA

REVISÃO DE TEXTO: FERNANDA LISBÔA

EDITORAÇÃO ELETRÔNICA: RODRIGO VALLS

EDIÇÃO REVISADA SEGUNDO O NOVO ACORDO ORTOGRÁFICO DA LÍNGUA PORTUGUESA.



EDIPUCRS – Editora Universitária da PUCRS

Av. Ipiranga, 6681 – Prédio 33
Caixa Postal 1429 – CEP 90619-900
Porto Alegre – RS – Brasil
Fone/fax: (51) 3320 3711
E-mail: edipucrs@pucrs.br
Site: www.pucrs.br/edipucrs

Dados Internacionais de Catalogação na Publicação (CIP)

E56 ENADE comentado [recurso eletrônico] : sistemas de informação 2014 /
organizadores Ana Paula Terra Bacelo, Marcelo Hideki Yamaguti.
– Dados eletrônicos. – Porto Alegre : EDIPUCRS, 2017.
Recurso on-line

Modo de acesso: <http://www.pucrs.br/edipucrs/>
ISBN 978-85-397-1004-1

1. Computação – Ensino – Avaliação. 2. Ensino superior – Brasil.
3. Informática. I. Bacelo, Ana Paula Terra. II. Yamaguti, Marcelo Hideki.

CDD 23. ed. 004

Ficha Catalográfica elaborada pelo Setor de Tratamento da Informação da BC-PUCRS.

TODOS OS DIREITOS RESERVADOS. Proibida a reprodução total ou parcial, por qualquer meio ou processo, especialmente por sistemas gráficos, microfílmicos, fotográficos, reprográficos, fonográficos, videográficos. Vedada a memorização e/ou a recuperação total ou parcial, bem como a inclusão de qualquer parte desta obra em qualquer sistema de processamento de dados. Essas proibições aplicam-se também às características gráficas da obra e à sua editoração. A violação dos direitos autorais é punível como crime (art. 184 e parágrafos, do Código Penal), com pena de prisão e multa, conjuntamente com busca e apreensão e indenizações diversas (arts. 101 a 110 da Lei 9.610, de 19.02.1998, Lei dos Direitos Autorais).

CONTEÚDO

APRESENTAÇÃO	7	QUESTÃO 21.....	49
NOTA DOS ORGANIZADORES	8	QUESTÃO 22.....	51
QUESTÃO DISCURSIVA 3	9	QUESTÃO 23.....	53
QUESTÃO DISCURSIVA 4	12	QUESTÃO 24.....	56
QUESTÃO DISCURSIVA 5	14	QUESTÃO 25.....	58
QUESTÃO 9.....	18	QUESTÃO 26.....	60
QUESTÃO 10.....	20	QUESTÃO 27.....	62
QUESTÃO 11.....	22	QUESTÃO 28.....	64
QUESTÃO 12.....	24	QUESTÃO 29.....	66
QUESTÃO 13.....	26	QUESTÃO 30.....	67
QUESTÃO 14.....	28	QUESTÃO 31.....	69
QUESTÃO 15.....	32	QUESTÃO 32.....	72
QUESTÃO 16.....	34	QUESTÃO 33.....	74
QUESTÃO 17.....	36	QUESTÃO 34.....	75
QUESTÃO 18.....	39	QUESTÃO 35.....	77
QUESTÃO 19.....	41		
QUESTÃO 20.....	47		

APRESENTAÇÃO

A consolidação do Sistema Nacional de Avaliação da Educação Superior (SINAES) – do qual o Exame Nacional de Desempenho de Estudantes (ENADE) é parte integrante – como mecanismo de avaliação da educação superior no Brasil revela o amadurecimento da sociedade em relação à compreensão da importância dos sistemas de avaliação externos de nossas instituições de ensino. Além dos objetivos mais óbvios de gerar indicadores que permitam as próprias instituições melhorarem sua qualidade nas dimensões de ensino, pesquisa, extensão e gestão, percebe-se a crescente importância desses indicadores tanto na orientação da expansão da oferta do ensino superior como na definição de políticas públicas voltadas ao financiamento do ensino e da pesquisa.

Em 2017 – ano de ENADE para a área de Exatas – a Faculdade de Informática da PUCRS completa 40 anos. Nesse momento simbólico, mais do que nunca, a Faculdade não poderia se furtar de estar mais uma vez engajada nesse processo avaliativo. O ENADE Comentado é a materialização desse engajamento. Resultado da colaboração de grande parte do corpo docente, resulta na visão que estes especialistas têm da área de Computação e abre aos discentes uma oportunidade objetiva de discussão.

O volume deste ano foi organizado pelos professores Ana Paula Bacelo e Marcelo Yamaguti e contou com a colaboração de todos os professores referenciados no texto. A todos os que participaram, deixamos nossos sinceros agradecimentos. Iniciativas como esta nos permitem entender como a Faculdade venceu seus primeiros 40 anos e, ao mesmo tempo, são a força que nos permitirá vencer os desafios que estão por vir.

Fernando Luis Dotti

Diretor da Faculdade de Informática (FACIN)

Bernardo Cosptein

Coordenador Acadêmico da FACIN

NOTA DOS ORGANIZADORES

Prezado leitor, a publicação do *ENADE Comentado 2014 – Sistemas de Informação* foi elaborada por professores da Faculdade de Informática da PUCRS com colaboração de professores da Escola de Negócios da PUCRS. O trabalho resultante visa auxiliar o estudo das questões formuladas no exame de 2014 e também disponibilizar material adicional para o desenvolvimento de atividades pedagógicas que orientem os estudantes no contínuo aprendizado e na revisão de seus conhecimentos na área.

Os organizadores agradecem aos professores que se dedicaram a comentar criticamente as questões e ao funcionário Evandro Gomes de Souza no apoio operacional, bem como à PROACAD e à EDIPUCRS pelo apoio incondicional que possibilitou a presente publicação.

Ana Paula Terra Bacelo
Marcelo Hideki Yamaguti
Organizadores

QUESTÃO DISCURSIVA 3

Uma aplicação mantém um grande conjunto de dados ordenados em uma lista linear em memória, implementada na forma de *array*. Os dados são estáveis, ou seja, são realizadas poucas operações de inserção e remoção, e a aplicação exige grande quantidade de pesquisas. Sabe-se que vários algoritmos permitem realizar pesquisas em um conjunto de dados, sendo alguns mais eficientes que outros. Nesse sentido, a pesquisa binária apresenta características bem definidas para se encontrar um dado procurado.

Considerando o contexto apresentado, desenvolva um algoritmo de pesquisa binária, de forma imperativa ou orientada a objetos, utilizando qualquer notação em português estruturado ou em uma linguagem de alto nível, como o Pascal, C ou Java. (Valor: 10,0 pontos)

RASCUNHO	
1	
2	
3	
4	
5	
6	
7	
8	
9	
10	
11	
12	
13	
14	
15	

Autores: Michael da Costa Móra e Bernardo Copstein.

COMENTÁRIO

Pesquisa Binária é um algoritmo de busca ou de pesquisa que encontra um determinado elemento em um *array* ordenado. O algoritmo compara o elemento sendo procurado com o elemento na posição do meio do *array*. Se eles forem diferentes, o algoritmo prossegue na metade onde o elemento pode se localizar: se o elemento sendo procurado for menor que o elemento do meio, a busca prossegue na metade inferior do *array*; caso contrário prossegue na metade superior do mesmo. Assim, o requisito para a utilização desse algoritmo é que os dados estejam ordenados.

A seguir, são apresentados dois exemplos de respostas escritos em linguagem de programação C, um utilizando a abordagem iterativa e outro utilizando a abordagem recursiva.

Solução iterativa:

```
// x: valor a ser pesquisado, n: tamanho do vetor, v: vetor de valores
// Se encontrar, retorna o índice do vetor, senão retorna -1
```

```
int buscaBinaria(int x, int n, int v[]) {
```

```
    int e, m, d;
```

```
    e = 0; d = n - 1;
```

```
    while (e <= d) {
```

```
        m = (e + d) / 2;
```

```
        if (v[m] == x)
```

```
            return m;
```

```
        if (v[m] < x)
```

```
            e = m + 1;
```

```
        else
```

```
            d = m - 1;
```

```
    }
```

```
    return -1;
```

```
}
```

Solução recursiva:

```
// x: valor a ser pesquisado, e: índice do início da pesquisa,
```

```
// d: índice do fim da pesquisa, v: vetor de valores
```

```
// Se encontrar, retorna o índice do vetor, senão retorna -1
```

```
int buscaBinaria(int x, int e, int d, int v[]) {  
  
    int m;  
  
    if (e <= d) {  
        m = (e + d) / 2;  
  
        if (v[m] == x)  
            return m;  
  
        if (v[m] < x)  
            return buscaBinaria(x, m+1, d, v);  
        else  
            return buscaBinaria(x, e, m-1, v);  
    }  
  
    else  
        return -1;  
}
```

REFERÊNCIAS

1. AHO, Alfred V. *Foundations of computer science*. New York: Computer Science Press, 1998.
2. CORMEN, Thomas H. *Introduction to algorithms*. 3. ed. Cambridge: The MIT Press, 2009.

QUESTÃO DISCURSIVA 4

Considere a situação em que o presidente de uma fábrica de embalagens tenha questionado os investimentos em Tecnologia da Informática (TI) realizados por seus antecessores, principalmente após seguidas paradas não programadas nos sistemas de informação da empresa. Em conversa com os executivos seniores, ele notou que era geral o descontentamento com os serviços prestados por TI. Expôs, então, o problema ao diretor de TI, que, demonstrando surpresa, afirmou: “Sei que temos passado por problemas ultimamente, mas esses detalhes não eram do meu conhecimento”. E continuou: “Temos trabalhado muito e prestado bons serviços para as áreas de negócio. Acredito que temos que melhorar nossa comunicação”.

O presidente da empresa concordou e solicitou-lhe a elaboração de um plano. Na semana seguinte, o diretor de TI apresentou um plano de ação em que propunha a adoção de várias práticas de ITIL (*Information Technology Infrastructure Library*).

Considerando a proposta desse profissional de TI, redija um texto esclarecendo, na perspectiva do ITIL, o que é Central de Serviços (*Service Level Agreement*) e como a adoção de ambos pode ajudar a área de TI da referida empresa a melhorar sua comunicação. (Valor: 10,0 pontos)

RASCUNHO	
1	
2	
3	
4	
5	
6	
7	
8	
9	
10	
11	
12	
13	
14	
15	

Autores: Dilnei Venturini e Leandro Bento Pompermaier.

COMENTÁRIO

Conforme o *Information Technology Infrastructure Library* (ITIL), a Central de Serviço é uma função dentro da TI que serve como ponto único de contato (SPOC – *Single Point of Contact*) entre os usuários e a área de Tecnologia da Informação (TI). Neste modelo, nas operações da TI, os profissionais de suporte aos usuários são separados daqueles que realizam atividade de desenvolvimento de *software* e resolução de problemas.

O principal objetivo da Central de Serviços é reestabelecer os serviços para o usuário o mais rápido possível. Esta organização oferece vantagens no atendimento aos usuários, como um suporte melhor e mais ágil, e também para a TI, pois o técnico especialista não é mais demandado diretamente pelas chamadas diretas do usuário.

Ainda conforme o ITIL, um *Service Level Agreement* (SLA) ou Acordo de Nível de Serviço (ANS) é um acordo interno entre a área de TI e seus usuários que descreve o serviço de TI, suas metas de nível de serviço, além dos papéis e responsabilidades das partes envolvidas. Um SLA inclui a definição de níveis mínimos de serviço que são esperados pelo usuário da área de TI; estes níveis são normalmente expressos através de indicadores que permitam a mensuração quantitativa da qualidade do serviço recebido pelo usuário.

Dentro desse contexto, a implementação desses conceitos promove uma maior transparência sobre a qualidade dos serviços prestados pela área de TI, pois, ao centralizar a comunicação e estabelecer níveis de serviço, é possível uniformizar expectativas e coletar indicadores sobre o desempenho da TI e de seus serviços.

REFERÊNCIAS

1. MAGALHÃES, Ivan Luiz; PINHEIRO, Walfrido Brito. *Gerenciamento de serviços de TI na prática: uma abordagem com base na ITIL: inclui ISO/IEC 20.000 e IT flex*. São Paulo: Novatec, 2008.
2. OFFICE OF GOVERNMENT COMMERCE. *ITIL*, v.3. 2007.

QUESTÃO DISCURSIVA 5

Uma empresa deseja lançar um sistema de comércio eletrônico para vender seus produtos. Essa empresa vende produtos de diversas categorias, como roupas, perfumes e eletrônicos, e aceita diversas formas de pagamento, como cartão de crédito e boleto bancário. No sistema de vendas implementado, cada produto deve ser cadastrado com sua descrição, preço de venda, quantidade em estoque e respectiva categoria. Cada cliente que deseja realizar compras tem de se cadastrar no sistema indicando seu nome, endereço e *e-mail*. Se o cliente for corporativo, deve cadastrar seu CNPJ e, se for individual, seu CPF. O cliente cadastrado pode realizar um pedido de compra dos produtos em estoque na quantidade que desejar. O cliente escolhe uma forma de pagamento disponível e recebe, por *e-mail*, o número do pedido e as informações do *status* do pedido. Após a confirmação do pagamento, a loja realiza a entrega dos itens solicitados no endereço do cliente e envia, por *e-mail*, a nota fiscal eletrônica. Tendo em vista que os preços dos produtos podem ser atualizados a qualquer momento, o sistema tem de ser capaz de reemitir uma nota fiscal de um pedido de compra de qualquer produto e respectivo preço na data da compra realizada pelo cliente.

Considerando esse cenário, proponha um Diagrama de Classes, segundo a UML (*Unified Modeling Language*), indicando nome de cada classe, respectivos atributos e relacionamentos entre as classes com as respectivas cardinalidades. Em sua proposta, identifique, pelo menos, um relacionamento de generalização e um relacionamento de composição, não sendo necessário indicar as operações de cada classe. (Valor: 10,0 pontos)

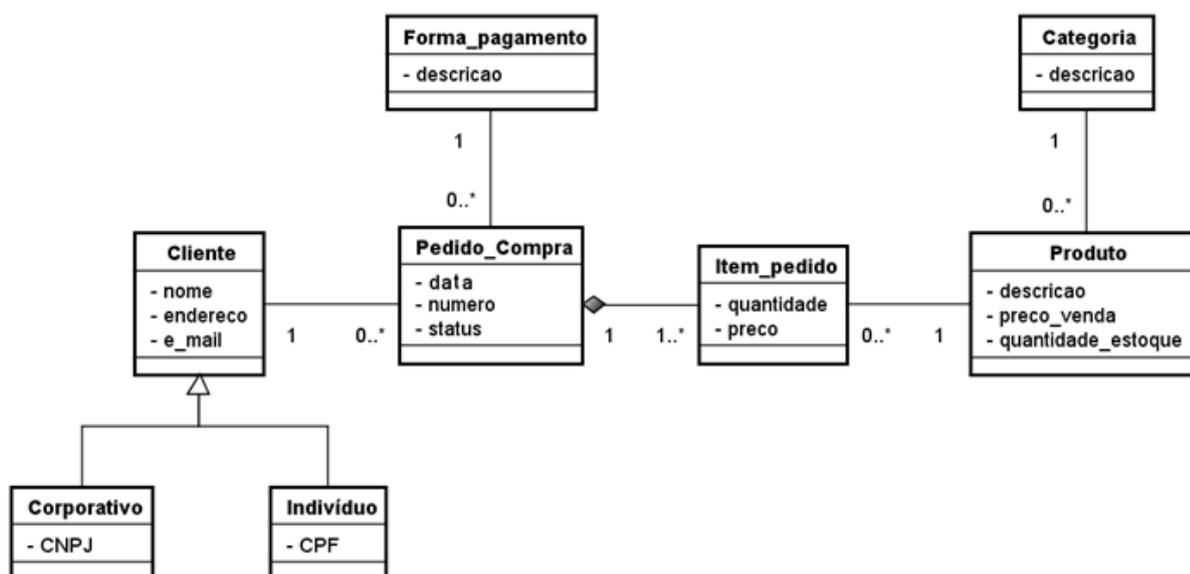
RASCUNHO	
1	
2	
3	
4	
5	
6	
7	
8	
9	
10	
11	
12	

13	
14	
15	

Autores: Marcelo Hideki Yamaguti e Daniel Antonio Callegari.

Padrão de resposta: O estudante deve apresentar o seguinte diagrama de classes padrão.

No mínimo, as classes, os atributos e as associações a seguir devem aparecer na solução proposta.



COMENTÁRIO

A questão envolve a modelagem, por meio de um diagrama de classes UML, de um cenário descrito textualmente. É solicitado que, na modelagem, seja apresentado o nome de cada classe e seus atributos, e os relacionamentos com suas cardinalidades. Também é solicitado que se identifique pelo menos um relacionamento de generalização e um de composição.

O contexto da descrição textual envolve um sistema de comércio eletrônico de uma empresa, sendo possível identificar algumas entidades candidatas, apresentadas a seguir.

- **Categoria**
- **Cliente**
- **Cliente corporativo**
- **Cliente individual**
- **Forma de pagamento**
- **Item solicitado**
- **Nota fiscal eletrônica**
- **Pedido de compra**
- **Produto**
- **Venda**

A entidade “Categoria” tem como instâncias “roupas”, “perfumes” e “eletrônicos”, enquanto a entidade “Forma de pagamento” tem como instâncias “cartão de crédito” e “boleto bancário”.

As entidades “Pedido de compra” e “Venda” podem ser consideradas sinônimos, pois do ponto de vista do cliente é uma compra, enquanto que do ponto de vista da empresa é uma venda.

Pode-se identificar uma generalização entre as entidades “Cliente”, “Cliente corporativo” e “Cliente individual”.

Assim, a lista de entidades candidatas poderia ser reduzida para a apresentada a seguir.

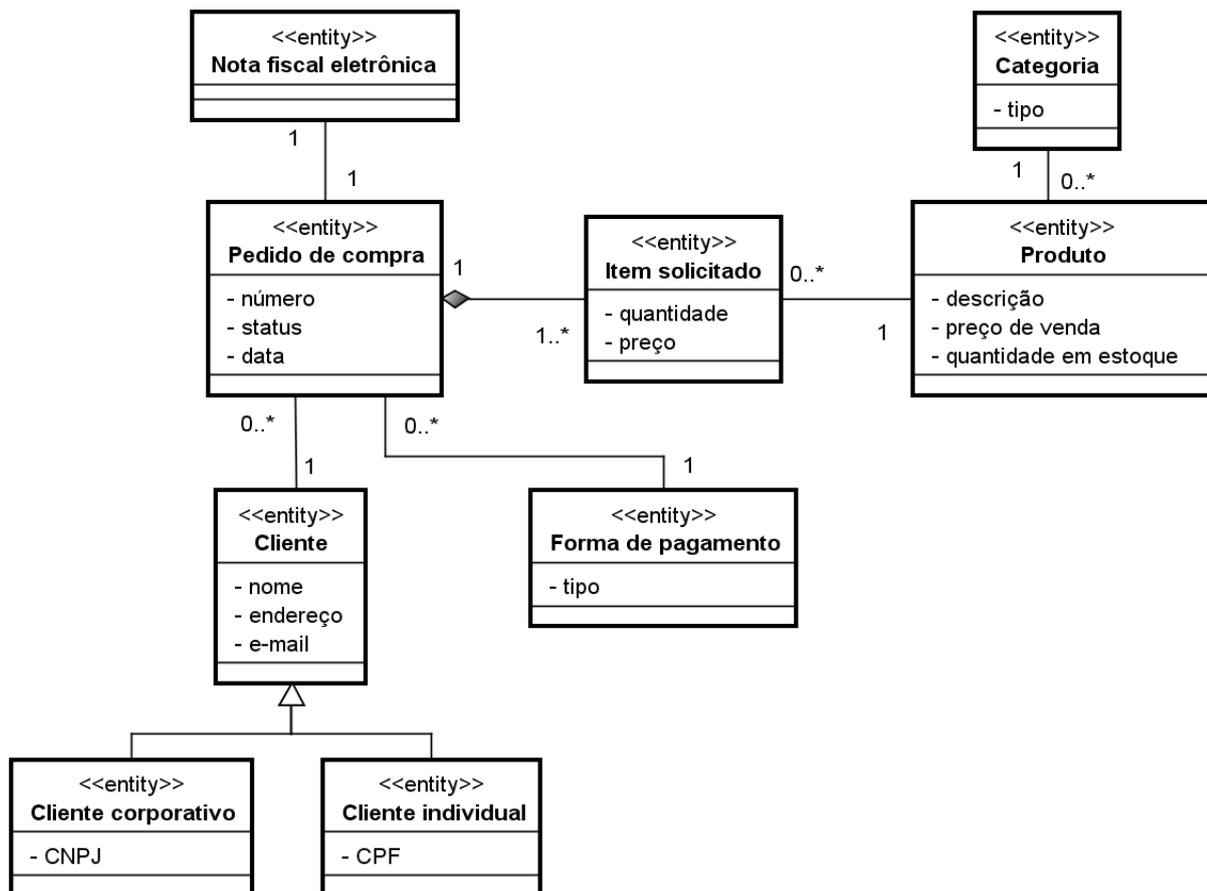
- **Categoria**
- **Cliente**
- **Cliente corporativo**
- **Cliente individual**
- **Forma de pagamento**
- **Item solicitado**
- **Nota fiscal eletrônica**
- **Pedido de compra**
- **Produto**

A partir dessas entidades, podem-se identificar alguns atributos correspondentes, listados a seguir.

- **Categoria**
- **Cliente (nome, endereço, e-mail)**
- **Cliente corporativo (CNPJ)**
- **Cliente individual (CPF)**
- **Forma de pagamento**
- **Item solicitado (quantidade, preço de venda)**
- **Nota fiscal eletrônica**
- **Pedido de compra (número, status, data)**
- **Produto (descrição, preço de venda, quantidade em estoque)**

Pode-se também identificar que um pedido de compra agrega vários itens solicitados e que cada item solicitado é de um pedido de compra. Assim, o relacionamento entre essas entidades poderia ser modelado como uma agregação de composição.

A partir dessas identificações, uma solução possível para o diagrama de classes correspondente é apresentada a seguir.



REFERÊNCIAS

1. BOOCH, G.; RUMBAUGH, J.; JACOBSON, I. *UML: guia do usuário*. 2. ed. Rio de Janeiro: Elsevier, 2012. 521 p.
2. GUEDES, G. T. A. *UML 2: uma abordagem prática*. 2. ed. São Paulo: Novatec, 2014. 192 p.

QUESTÃO 9

Considere uma situação em que um professor que queira saber se existem alunos cursando, ao mesmo tempo, as disciplinas A e B tenha implementado um programa que:

1. inicializa um *array* **a** de 30 posições que contém as matrículas dos alunos da disciplina A;
2. inicializa outro *array* **b** de 40 posições que contém as matrículas dos alunos da disciplina B;
3. imprime a matrícula dos alunos que estão cursando as disciplinas A e B ao mesmo tempo.

Considere, ainda, que *arrays* foram declarados e inicializados, não estão necessariamente ordenados, e seus índices variam entre 0 e $n - 1$, sendo n o tamanho do *array*.

1. For ($i = 0$ to 29) {
2. For ($j = 0$ to 39) {
- 3.
- 4.
- 5.
6. }
7. }

Com base nessas informações, conclui-se que o trecho a ser incluído nas linhas 3, 4 e 5 do código acima, para que o programa funcione corretamente, é

- A.** 3. if ($a[i] == b[j]$) {
4. print ($a[i]$);
5. }
- B.** 3 if ($a[j] == b[i]$) {
4. print ($a[j]$);
5. }
- C.** 3. if ($a[i] == b[j]$) {
4. print ($a[j]$);
5. }
- D.** 3. if ($a[i] == b[i]$) {
4. print ($a[i]$);
5. }

```
E. 3. if (a[j] == b[j]) {
    4.     print (a[j]);
    5. }
```

Gabarito: A.

Autores: Daniel Antonio Callegari e Silvia Maria Wanderley Moraes.

COMENTÁRIO

A questão solicita que sejam identificados, por meio dos seus números de matrícula, os alunos que estão cursando simultaneamente as disciplinas A e B. De acordo com o enunciado da questão, os números de matrícula dos alunos que estão cursando a disciplina A estão em um *array* **a** de 30 posições e daqueles que estão cursando a disciplina B em um *array* **b** de 40 posições. Ambos os *arrays* não estão necessariamente ordenados.

Para resolver esta questão, é necessário identificar os elementos de interseção entre os conjuntos de dados **a** e **b** ($a \cap b$), ou seja, identificar as matrículas em **a** que também estão em **b**: $\{x \mid (x \in a) \cap (x \in b)\}$.

Com esse propósito, a questão apresenta um trecho de código que contém estrutura de repetição aninhada necessária para implementar a operação de interseção entre os *arrays* **a** e **b**. O primeiro laço de repetição (linha 1) gera valores de 0 a 29 para a variável **i**, enquanto o segundo laço de repetição (linha 2) produz valores de 0 a 39 para a variável **j**. Como os laços estão aninhados, para cada valor de **i** (0, 1, 2, etc.), **j** assumirá 40 valores diferentes (0, 1, 2, ..., 39), resultando em 120 iterações (30x40).

Para responder à questão, deve-se escolher, dentre as alternativas, o trecho de código que complementa a estrutura de repetição dada e implementa a operação de interseção corretamente. É um problema de indexação. E, nesse caso, apenas a alternativa **A** indexa corretamente os *arrays* **a** e **b**. O *array* **a**, por ter 30 posições, deve ser acessado pelo índice **i**, que varia de 0 a 29; e o *array* **b**, por ter 40 posições, deve ser acessado pelo índice **j**, que varia de 0 a 39.

A alternativa **B** é incorreta por duas razões. A primeira é que **j** é usado erroneamente como índice do *array* **a**. Como **j** varia de 0 a 39, ele acessará indevidamente as posições de 30 a 39 que não existem em **a**. A segunda é o uso do **i** como índice de **b**. Nesse caso, como **i** varia de 0 a 29, ele deixará de acessar 10 elementos do vetor **b** (aquelas das posições de 30 a 39).

Na alternativa **C**, o erro está na instrução *print* (linha 4). A comparação entre os elementos (linha 3) está correta, mas a escrita do elemento não está. Em vez de escrever $a[i]$, escreve $a[j]$, que, além de ser outro elemento, contém um erro de indexação, pois **a** possui apenas 30 posições (de 0 a 29) e **j** varia de 0 a 39.

Na alternativa **D**, o erro é que cada elemento de **a** não é comparado com todos os elementos em **b**, já que **i** é usado para indexar os dois *arrays*. O trecho em **D** incorretamente compara o elemento $a[i]$ com o elemento $b[i]$ 40 vezes. Além disso, **i**, por variar apenas de 0 a 29, não consegue acessar todos os elementos em **b**.

Por fim, a alternativa **E** está errada por razões semelhantes às da alternativa **D**. Ela faz uso incorreto de **j** para indexar os dois *arrays*. O índice **j** permitirá acesso a posições que não existem em **a** (de 30 a 39). O trecho executará 30 vezes as comparações dos elementos de **a** e **b**, que estão nas mesmas posições.

REFERÊNCIAS

1. AGUILAR, L. J. *Fundamentos de programação: algoritmos, estruturas de dados e objetos*. 3. ed. Porto Alegre: McGraw-Hill, 2008. 720 p.
2. FORBELLONE, A. L. V.; EBERSPACHER, H. F. *Lógica de programação: a construção de algoritmos e estruturas de dados*. 3. ed. São Paulo: Prentice Hall, 2005. 218 p.

QUESTÃO 10

A área de complexidade de algoritmos abrange a medição da eficiência de um algoritmo frente à quantidade de operações realizadas até que ele encontre seu resultado final.

A respeito desse contexto, suponha que um arquivo texto contenha o nome de N cidades de determinado estado, que cada nome de cidade esteja separado do seguinte por um caractere especial de fim de linha e classificado em ordem alfabética crescente. Considere um programa que realize a leitura linha a linha desse arquivo, à procura de nome de cidade.

Com base nessa descrição, verifica-se que a complexidade desse programa é

- A. $O(1)$, em caso de busca sequencial.
- B. $O(N)$, em caso de busca sequencial.
- C. $O(\log_2 N)$, em caso de busca binária.
- D. $O(N)$, em caso de transferência dos nomes para uma árvore binária e, então, realizar a busca.
- E. $O(\log_2 N)$, em caso de transferência dos nomes para uma árvore binária e, então, realizar a busca.

Gabarito: B.

Autores: Alexandre Agustini e Marco Aurélio Souza Mangan.

COMENTÁRIO

A análise de complexidade de algoritmos é realizada através da análise assintótica, que utiliza a notação O . Existe um arquivo em formato texto e ficamos sabendo que: (a) o tamanho da entrada é definido pelo número de cidades N , (b) os nomes estão em ordem alfabética e (c) cada nome de cidade está em uma linha. O programa que deve ser analisado procura o nome de uma cidade no arquivo, realizando a leitura linha a linha.

A alternativa **A** assume que a busca sequencial pode ser realizada em tempo constante ($O(1)$), ou seja, encontrar o elemento da primeira posição do arquivo tem o mesmo tempo de execução que encontrar o elemento da segunda posição ou mesmo a última (posição N). Como o arquivo é lido linha a linha, claramente para encontrar o primeiro elemento será necessário ler apenas a primeira linha; para o segundo elemento, as duas primeiras linhas; e assim sucessivamente até necessitar realizar a leitura das N linhas do arquivo para o último elemento. Desta forma temos um crescimento linear no número de elementos e não constante, correspondendo à alternativa **B**, que é a correta para a questão.

Para a análise da alternativa **C**, assumimos que não é possível fazer acesso aleatório ao arquivo, pois as linhas podem ter tamanho variável e o deslocamento para a linha de posição ($N/2$) teria de ser realizado linha a linha (sequencial) até encontrar $N/2$ marcadores de final de linha (e desta forma não será possível executar o algoritmo em tempo logarítmico). Podemos projetar, ainda, um algoritmo que calcula o ponto médio do arquivo a partir do seu tamanho em número de bytes e que “ajusta” para o marcador de final de linha mais próximo. Este algoritmo poderia executar em tempo logarítmico se o sistema permitisse acesso aleatório por caracteres, o que não é possível, pois o enunciado define que a leitura é realizada linha a linha; logo a alternativa é falsa.

As duas últimas alternativas supõem a transferência dos dados para uma árvore binária, sem especificar qual o seu tipo. Em uma árvore binária de pesquisa balanceada, o tempo de busca é $O(\log_2 N)$ e, em uma implementação simples de árvore binária, o tempo de busca é $O(N)$, que fazem com que essas alternativas pareçam plausíveis. O tempo de construção da árvore balanceada é $O(N \cdot \log_2 N)$, com N operações de leitura ao arquivo e, para cada elemento, no mínimo $\log_2 N$ operações para realizar a inserção. Caso a árvore não seja balanceada, o custo de inserção de cada elemento é $O(N)$ e o tempo de construção $O(N^2)$. Logo as duas últimas alternativas são falsas, pois o tempo total do algoritmo será a soma do tempo de construção mais o tempo de busca.

REFERÊNCIAS

1. CORMEN, Thomas H. et al. *Introduction to algorithms*. 3. ed. Cambridge: The MIT Press, 2009.
2. MCALLISTER, William. *Data structures and algorithms using Java*. 1. ed. Boston: Jones and Bartlett, 2009.
3. RAWLINS, Gregory J. E. *Compared to what? An introduction to the analysis of algorithms*. New York, NY: Computer Science Press, 1992.

QUESTÃO 11

Considere que uma rede de supermercados tenha contratado um analista de informática para elaborar um plano de prevenção de erros na operação do *software* das lojas. Com o objetivo de melhorar a tomada de decisão, o analista classificou os erros quanto ao tempo de emprego do operador (linha) e quanto aos setores que originaram os erros (coluna). Os dados levantados são apresentados na tabela abaixo.

Tabela de erros da operação do *software*

Tempo de emprego do empregador	Setor do supermercado			TOTAL
	Caixa	Padaria	Açougue	
< 2 anos (A)	55	25	10	90
2 a 4 anos (B)	20	6	4	30
> 4 anos (C)	35	15	10	60
TOTAL	110	46	24	180

Considerando essas informações, conclui-se que a probabilidade aproxima de um erro ser cometido por um operador com tempo de emprego correspondente a 4 anos ou menos é de

- A. 17%.
- B. 33%.
- C. 50%.
- D. 67%.
- E. 83%.

Gabarito: D.

Autores: Rafael Matone Chanin e João Batista Souza de Oliveira.

COMENTÁRIO

Esta questão pode ser resolvida de maneira intuitiva, sem o conhecimento formal da Teoria dos Conjuntos ou Probabilidade. Quando a pergunta solicita a probabilidade de um **operador** cometer um erro, ela se refere a operadores de qualquer setor (Caixa, Padaria ou Açougue). Logo, podemos considerar apenas a última coluna, pois nela encontramos o total dos erros de todos os operadores (independentemente do setor).

A segunda informação que a questão traz é o tempo de emprego do operador: devemos calcular a probabilidade de erro para operadores com tempo de emprego igual ou inferior a 4 anos. Essas informações podem ser encontradas nas duas primeiras linhas da tabela.

Logo, podemos saber a quantidade total de erros que esses operadores cometem de duas formas.

1. Usando as três primeiras colunas, podemos somar os erros de todos os operadores que possuem 4 ou menos anos de tempo de emprego:

$$\text{Total de erros} = 55 + 25 + 10 + 20 + 6 + 4 = 120.$$

2. Olhar apenas a última coluna (TOTAL) e somar os erros totais dos operadores com 4 ou menos anos de trabalho (as duas primeiras linhas):

$$\text{Total de erros} = 90 + 30 = 120.$$

Como o total de erros de todos os operadores é 180, para responder à questão, basta dividirmos os 120 pelo total: $120/180 = 0,6667$, ou seja, aproximadamente 67%.

REFERÊNCIAS

1. ALENCAR FILHO, E. *Teoria elementar dos conjuntos*. São Paulo: Nobel, 1976.
2. BUSSAB, W. O; MORETTIN, P. A. *Estatística básica*. 5. ed. São Paulo: Saraiva, 2002.
3. MEYER, P. L. *Probabilidade: aplicações à Estatística*. Rio de Janeiro: LTC, 2000.

QUESTÃO 12

Considere a situação em que uma empresa que comercializa *software* customizado precisa definir a alocação de quantidade de horas de trabalho para um programador e um desenvolvedor. Considere, ainda, que os profissionais estão envolvidos no projeto de um sistema composto por módulos de controle de estoque e vendas e que os recursos financeiros a serem investidos para realizar as atividades de programação e desenvolvimento para cada hora trabalhada nos respectivos módulos são indicados na tabela abaixo. Assim, por exemplo, o custo unitário por hora trabalhada na atividade de programação no módulo de estoque é \$ 3, ao passo que o custo do desenvolvimento do respectivo módulo é \$ 6.

Atividade	Módulo	
	Estoque	Vendas
Programação	\$ 3	\$ 3
Desenvolvimento	\$ 6	\$ 3

O valor previsto em orçamento para a atividade de programação é de \$ 30 e para a de desenvolvimento é de \$ 48. Ao modelar esse problema para ser resolvido com a técnica de pesquisa operacional, define-se V como a quantidade de horas alocadas no módulo de vendas e E como a quantidade de horas alocadas no módulo de estoque. Nesse modelo, uma das restrições a serem utilizadas é representada por

- A. $3E + 3V \leq 30$.
- B. $3E + 3V \geq 30$.
- C. $3E + 3V \leq 48$.
- D. $6E + 3V \leq 30$.
- E. $6E + 3V \geq 48$.

Gabarito: A.

Autores: Paulo Cesar Chiechelski, Gilberto Keller Andrade e Ionara Rech.

COMENTÁRIO

A questão 12 trata de um problema típico de alocação de recursos que pode ser resolvido mediante a utilização de técnica de Pesquisa Operacional denominada de Programação Linear. Um problema de alocação envolve a utilização mais eficiente de recursos mediante a otimização de resultados. Por sua vez, a Programação Linear é uma modalidade de programação matemática na qual a modelagem do problema é inteiramente expressa em termos de relações lineares (equações ou inequações).

No caso específico da Questão 12, a modelagem utilizada para resolver o problema define as variáveis **V** como quantidade de horas alocadas no módulo de vendas e **E** como quantidade de horas alocadas no módulo de estoque. Dessa forma, os módulos de vendas e de estoque do sistema requerem horas de programação e de desenvolvimento para serem elaborados. Essas horas de programação e desenvolvimento apresentam custos unitários, e a empresa dispõe de orçamento limitado para viabilizar cada uma dessas atividades.

Assim, o custo total das horas de programação não pode exceder ao orçamento fixado de \$ 30 (unidades monetárias). Entretanto, as horas de programação podem ser alocadas tanto para o módulo de vendas como para o módulo de estoque. Sabendo-se que **V** corresponde ao número de horas alocadas para o módulo de vendas e que **E** corresponde ao número de horas alocadas para o módulo de estoque, então o custo total das horas de programação, para qualquer quantidade de horas empregadas em qualquer um dos módulos, pode ser expresso, de uma forma genérica, como $3V+3E$. Isto é, o custo total das horas de programação é igual ao custo unitário das horas de programação para o módulo de vendas (\$ 3) vezes a quantidade de horas alocadas para esse módulo (simbolizada por **V**), mais o custo unitário das horas de programação para o módulo de estoque (\$ 3) vezes a quantidade de horas alocadas para o módulo de estoque (simbolizada por **E**). Ademais, esse custo total das horas de programação não pode exceder o limite orçamentário para essa atividade. Dessa forma, chega-se à seguinte relação linear (inequação): $3V + 3E \leq 30$, que corresponde à alternativa **A**.

De outro lado, a alternativa **B** não poderia ser escolhida, pois enfatiza que o custo total das horas de programação deve ser igual ou superior a \$ 30. Em realidade, o orçamento fixa \$ 30 como limite máximo.

A alternativa **C** não está correta devido ao fato de utilizar argumentos do lado esquerdo da expressão matemática para o custo total de horas de programação e, do lado direito da expressão, empregar limite orçamentário para a atividade de desenvolvimento (\$ 48).

A alternativa **D** está incorreta pela mesma razão da alternativa **C**, mas neste caso houve uma inversão de argumentos. Do lado esquerdo da expressão matemática caracterizou-se o custo total das horas de desenvolvimento; e, do lado direito, o limite orçamentário da atividade de programação.

Por último, a alternativa **E** não pode ser considerada correta, pois muito embora esteja utilizando argumentos do mesmo recurso (horas de desenvolvimento), inverteu o sentido do sinal, indicando que o valor \$ 48 não seria o limite orçamentário, mas sim o valor mínimo que deveria ser praticado para essa atividade.

QUESTÃO 13

Na Figura 1, abaixo, está representado, esquematicamente, um processo de tradução de um programa (arquivo fonte) em um código binário. Esse processo de compilação clássica é utilizado em compiladores como os das linguagens C e Pascal.

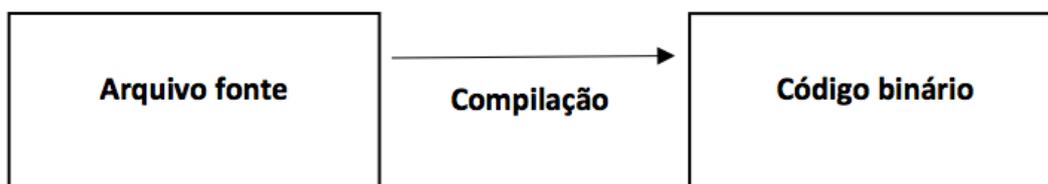


Figura 1 – Processo de compilação clássica

Na Figura 2, abaixo, está representado, esquematicamente, um processo de tradução de um programa (arquivo fonte) em um código intermediário. Esse processo híbrido é utilizado em compiladores como os das linguagens Java e C#.

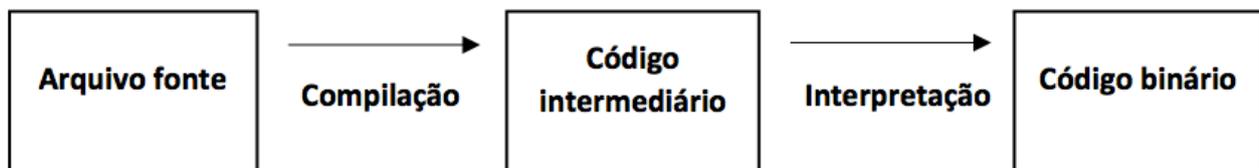


Figura 2 – Processo híbrido

Considerando que, em ambos os processos, o código binário é o que será executado pelo computador e que a execução de dois programas gerados, cada qual por um dos processos apresentados, ocorre em situações equivalentes, avalie as afirmações a seguir.

- I. Há portabilidade para a execução de ambos os programas gerados em cada processo.
- II. Na execução do programa gerado por meio do processo híbrido, o consumo de memória é maior que na execução pelo processo de compilação clássica.
- III. O desempenho na execução do programa gerado pelo processo de compilação clássica é melhor que na execução pelo processo híbrido.

É correto o que se afirma em

- A. I, apenas.
- B. III, apenas.
- C. I e II, apenas.

D. II e III, apenas.

E. I, II e III.

Gabarito: D.

Autores: Sérgio Johann Filho e Júlio Henrique A. P. Machado.

COMENTÁRIO

A questão aborda de maneira comparativa os processos de compilação tradicionalmente utilizados em linguagens de programação, separando esses em processo de compilação clássica e processo híbrido. Ambos os processos possuem vantagens e desvantagens com relação a flexibilidade, portabilidade e uso de recursos computacionais.

Como ilustrado na questão, no processo clássico, um arquivo fonte é transformado diretamente em código binário nativo, que pode ser executado diretamente na arquitetura-alvo (não estamos considerando outras necessidades em termos de recursos como o sistema operacional, bibliotecas, entre outros). No processo híbrido, o arquivo fonte é compilado e transformado em código intermediário que pode ser interpretado e transformado em código nativo, usualmente através do conceito de uma máquina virtual que roda sobre uma arquitetura-alvo.

Considerando os comentários sobre código binário realizados na questão e revendo individualmente cada uma das alternativas, pode-se afirmar o que segue.

I. Há portabilidade para execução de ambos os programas gerados no processo. Falso. Existe portabilidade apenas quando o processo híbrido for utilizado, uma vez que uma máquina virtual que seja compatível com a mesma interface do código intermediário pode existir em qualquer sistema e executar o mesmo programa sem que sejam necessárias modificações. No processo de compilação clássica, a etapa responsável pela geração de código é dependente de arquitetura, ou seja, um programa precisa ser recompilado para a arquitetura-alvo, o que torna a execução do programa não portátil.

II. Na execução do programa gerado por meio do processo híbrido, o consumo de memória é maior que na execução pelo processo de compilação clássica. Verdadeiro. Tradicionalmente o código em linguagem intermediária (a ser executado em uma máquina virtual) é bastante genérico e muitas vezes não explora diretamente os recursos disponibilizados pelo hardware. Por exemplo, o uso da pilha e o acesso a estruturas de dados é feito de maneira mais abstrata. Com isso, são criadas internamente referências adicionais a objetos para que um programa possa executar dinamicamente (interpretado ou compilado em run-time), aumentando o uso de memória. Em um processo de compilação clássica, tais referências podem ser resolvidas estaticamente, reduzindo o consumo de memória durante a execução.

III. O desempenho na execução do programa gerado pelo processo de compilação clássica é melhor que na execução pelo processo híbrido. Verdadeiro. O processo híbrido pode utilizar interpretação ou compilação just-in-time. Na interpretação de código, uma instrução em linguagem intermediária se desdobra em múltiplas instruções nativas. Na compilação just-in-time, trechos de código frequentemente acessados são compilados dinamicamente (em tempo de execução) e executados nativamente. Em ambos os casos, existem custos adicionais em termos de desempenho quando comparado à compilação clássica. O código intermediário não pode aproveitar da melhor maneira possível os recursos de um determinado processador, como instruções especiais e acesso otimizado à memória de acordo com os princípios de localidade. O uso de mais memória e de custos altos de análise de posicionamento de estruturas de dados (inviável em run-time) faz com que o código compilado da maneira clássica possua vantagens com relação ao desempenho, pois diversas análises podem ser realizadas de maneira estática, não gerando penalidades em tempo de execução.

REFERÊNCIA

1. AHO, A.V.; SETHI, S.; ULLMAN, J. D. *Compiladores: princípios, técnicas e ferramentas*. 2. ed. São Paulo: Pearson, 2008. 634 p.

QUESTÃO 14

A tabela abaixo apresenta o ranking das 20 linguagens de programação mais populares.

Jul. 2014	Jul. 2013	Change	Programming Language	Ratings	Change
1	1		C	17.145%	-0,48%
2	2		Java	15.688%	-0,22%
3	3		Objective-C	10.294%	+0,05%
4	4		C++	5.520%	-3,23%
5	7	↗	(Visual) Basic	4.341%	+0,01%
6	6		C#	4.051%	-2.16%
7	5	↘	PHP	2.916%	-4.27%
8	8		Python	2.656%	-1.38%
9	10	↗	JavaScript	1.806%	-0,04
10	12	↗	Transact-SQL	1.759%	+0,19%
11	9	↘	Perl	1.627%	-0,52%
12	13	↗	Visual Basic. NET	1.495%	+0,24%
13	37	↗↗	F#	1.093%	+0,86%
14	11	↘	Ruby	1.072%	-0,51%
15	45	↗↗	ActionScript	1.067%	+0,86%
16	-	↗↗	Swift	1.054%	+1,05%
17	17		Delphi/Object Pascal	1.031%	+0,34%
18	15	↘	Lisp	0,829%	-0,04%
19	18	↘	MATLAB	0,781%	+0,10%
20	20		Assembly	0,777%	+0,20%

Considerando a classificação das linguagens de programação apresentada na tabela, avalie as afirmações a seguir.

- I. O *ranking* é liberado por uma linguagem de terceira geração e de paradigma funcional.
- II. Linguagens com licença livre têm participação inferior a 20% se somados os percentuais de popularidade.
- III. É maior, no *ranking*, a participação de linguagens que utilizam o paradigma orientado a objetos que a das linguagens que utilizam os demais paradigmas.

É correto o que se afirma em

- A. II, apenas.
- B. III, apenas.
- C. I e II, apenas.
- D. I e III, apenas.
- E. I, II e III.

Gabarito: B.

Autores: Marco Antonio Insaurriaga Gonzalez e Flávio Moreira de Oliveira.

COMENTÁRIO

A afirmação **I** está **incorreta**, porque a linguagem C [SCH2006], embora seja de terceira geração, é uma linguagem estruturada (ou procedural), não pertencendo, portanto, ao paradigma funcional.

A afirmação **II** está **incorreta**, porque, com certeza, as seguintes linguagens possuem licença livre e já atingem mais de 20% no percentual (50,290%): C [COD2016], Java [HOR2013], C++ [COD2016], PHP [PHP2016], Python [PYT2016], JavaScript [W3S2016], Perl [PER2016], Ruby [RUB2016], Delphi [DEL2016] e Lisp [LIS2016]. Não possuem licença livre (26,712%): Objective-C [APL2016b], Visual Basic [MIC2016b], C# [MIC2016d], Transact-SQL [MIC2016a], Visual Basic .NET [MIC2016c], F# [FSH2016], ActionScript [ADO2016], Swift [APL2016a], Matlab [MAT2016] e Assembly [IBM2016]. Vale ressaltar, no entanto, que mesmo C# e F#, por exemplo, podem ser utilizadas na versão *community* do IDE Visual Studio, que é livre. Assim, seria preciso definir com mais clareza o que se entende por “licença livre” no enunciado.

A afirmação **III** está **correta**, porque as linguagens que utilizam o paradigma de orientação a objetos, de forma pura ou predominantemente, constituem a grande maioria, sendo elas: Java [HOR2013], Objective-C [KOC2013], C++ [DEI2008], C# [ROB2004], Python [PHI2010], Ruby [SHA2013], ActionScript [BRA2010], Swift [APL2016a] e Delphi [DEL2016]. Ainda PHP [MEL2008], JavaScript [GOO2010], Perl [FLY2006] e Matlab [LIN2016] [MAT2016] podem assumir características de orientação a objetos. Como minoria, temos as linguagens procedurais C [SCH2006] e Assembly [HYD2010], as linguagens orientadas a eventos Visual Basic [EVJ2008] e Visual Basic .NET [MIC2016c], a linguagem de consulta estruturada Transact-SQL, que inclui programação procedural [RAM2001], e as linguagens funcionais F# [FSH2016] e Lisp [MCC1979]. O primeiro grupo (linguagens orientadas a objetos) soma 42,433%, o segundo grupo alcança 7,13%, e o terceiro grupo chega a 27,439%. Mesmo somados, os dois últimos grupos (34,569%) não ultrapassariam o primeiro. Por outro lado, considerando a soma dos dois primeiros grupos, teríamos 49,563%, porcentagem superior ao terceiro, confirmando a supremacia das linguagens orientadas a objetos.

Assim sendo, a alternativa **B** (III, apenas) é a correta.

REFERÊNCIAS

1. [ADO2016] ADOBE. *ActionScript 3.0 – Overview*. Disponível em: <<https://goo.gl/b4iNSJ>>. Acesso em: 10 dez. 2016.
2. [APL2016a] APPLE. *Swift Documentation*. Disponível em: <<https://goo.gl/doBy0d>>. Acesso em: 10 dez. 2016.
3. [APL2016b] APPLE. *Objective-C*. Disponível em: <<https://goo.gl/kch87B>>. Acesso em: 10 dez. 2016.
4. [BRA2010] BRAUNSTEIN, Rober. *ActionScript 3.0 – Bible*. Indianapolis: Wiley, 2010.
5. [COD2016] CODEBLOCKS. *The open source, cross platform, free C, C++ and Fortran IDE*. Disponível em: <<http://www.codeblocks.org/>>. Acesso em: 10 dez. 2016.
6. [DEI2008] DEITEL, Harvey M. et al. *C++: como programar*. Porto Alegre: Bookman, 2008.
7. [DEL2016] DELPHI. *A Beginner's Guide to Delphi Programming*. Disponível em: <<https://goo.gl/S8BHhy>>. Acesso em: 10 dez. 2016.
8. [EVJ2008] EVJEN, Bill. *Professional Visual Basic 2008*. Indianapolis: Wiley Pub, 2008.
9. [FLY2006] FLYNT, John P. *Perl Power!: the Comprehensive Guide*. Boston: Thomson Course Technology PTR, 2006.
10. [FSH2016] FSHARP. *Getting Start*. Disponível em: <<http://fsharp.org/>>. Disponível em: 10 dez. 2016.
11. [GHE1998] GHEZZI, C.; JAZAYERI, M. *Programming language concepts*. 3. ed. New York: John Wiley, 1998.
12. [GOO2010] GOODMAN, Danny. *JavaScript Bible*. 7. ed. Indianapolis: Wiley, 2010.
13. [HOR2013] HORSTMANN, C. *Java for everyone: late objects*. 2. ed. Danvers: Wiley, 2013.
14. [HYD2010] HYDE, Randall. *The Art of Assembly Language*. São Francisco: No Starch Press, 2010.
15. [IBM2016] IBM KNOWLEDGE CENTER. *Assembler Language*. Disponível em: <<https://goo.gl/swjxmA>>. Acesso em: 10 dez. 2016.
16. [KOC2013] KOCHAN, Stephen G. *Programação com Objective-C*. Porto Alegre: Bookman, 2013.
17. [LIN2016] LINGE, Svein. *Programming for Computations: MATLAB/Octave*. S.l.: Springer International Publishing, 2016.
18. [LIS2016] LISPWORKS. *LispWorks for Windows*. Disponível em: <<https://goo.gl/mok9rn>>. Acesso em: 10 dez. 2016.
19. [MAT2016] MATHWORKS. *Features*. Disponível em: <<https://goo.gl/oOBh61>>. Acesso em: 10 dez. 2016.
20. [MCC1979] MCCARTHY, John. *History of Lisp*. 1979. Disponível em: <<https://goo.gl/bhiEw>>. Acesso em: 10 dez. 2016.
21. [MEL2008] MELONI, Julie C.; TELLES, Matthew A. *PHP 6 fast & easy Web development*. Boston: Course Technology, 2008.
22. [MIC2016a] MICROSOFT. *Transact-SQL Reference*. Disponível em: <<https://goo.gl/o42U4n>>. Acesso em: 10 dez. 2016.
23. [MIC2016b] MICROSOFT. *Visual Basic*. Disponível em: <<https://goo.gl/Xnnu0T>>. Acesso em: 10 dez. 2016.
24. [MIC2016c] MICROSOFT. *Visual Basic .NET Language Specification*. Disponível em: <<https://goo.gl/bk6GEz>>. Acesso em: 10 dez. 2016.
25. [MIC2016d] MICROSOFT. *C# Programming Guide*. Disponível em: <<https://goo.gl/wz1ebA>>. Acesso em: 10 dez. 2016.

26. [PER2016] PERL. *Documentação*. Disponível em: <<https://www.perl.org/>>. Disponível em: 10 dez. 2016.
27. [PHP2016] PHP. *Manual*. Disponível em: <<https://secure.php.net/docs.php>>. Acesso em: 10 dez. 2016.
28. [PHI2010] PHILLIPS, Dusty. *Python 3 – object oriented programming: harness the power of Python 3 objects*. Birmingham: Packt Pub, 2010.
29. [PYT2016] PYTHON. *Manual*. Disponível em: <<https://www.python.org/doc/>>. Acesso em: 10 dez. 2016.
30. [RAM2001] RAMALHO, J. A. A. *Transact-SQL: guia de referência*. São Paulo: Berkeley, 2001.
31. [ROB2004] ROBINSON, Simon et al. *Professional C# – programando*. São Paulo: Pearson Education, 2004.
32. [RUB2016] RUBY. Disponível em: <<https://www.ruby-lang.org/en/>>. Acesso em: 10 dez. 2016.
33. [SCH2006] SCHILDT, Herbert. *C Completo e Total*. São Paulo: Pearson Education, 2008.
34. [SEB2011] SEBESTA, R. W. *Conceitos de linguagens de programação*. São Paulo: Bookman, 2011.
35. [SHA2013] SHAUGHNESSY, Pat. *Ruby Under a Microscope: Learning Ruby Internals Through Experiment*. São Francisco: No Starch Press, 2013.
36. [W3S2016] W3SCHOOLS. *Tutorial*. Disponível em: <<http://www.w3schools.com/js/>>. Acesso em: 10 dez. 2016.

QUESTÃO 15

Os sistemas de informação podem ser classificados em sistemas OLTP (*Online Transaction Processing*) e sistemas OLAP (*Online Analytical Processing*).

A respeito desses tipos de sistemas, avalie as asserções a seguir e a relação proposta entre elas.

- I. Em sistemas OLTP, os dados são normalmente estruturados em um modelo relacional normalizado, e em sistemas OLAP, os dados são normalmente estruturados em um modelo multidimensional; contudo, esses sistemas estão inter-relacionados.

PORQUE

- II. Os sistemas OLAP, na maioria das vezes, obtêm dados a partir de processos ETL (*Extract, Transform, Load*), que recuperam e transformam os dados obtidos a partir das bases de dados dos sistemas OLTP.

A respeito dessas asserções, assinale a opção correta.

- A. As asserções I e II são proposições verdadeiras, e a II é uma justificativa correta da I.
B. As asserções I e II são proposições verdadeiras, e a II não é uma justificativa correta da I.
C. A asserção I é uma proposição verdadeira, e a II é uma proposição falsa.
D. A asserção I é uma proposição falsa, e a II é uma proposição verdadeira.
E. As asserções I e II são proposições falsas.

Gabarito: A.

Autores: Renata De Paris e Duncan Dubugras Alcoba Ruiz.

COMENTÁRIO

A questão aborda o assunto sobre processamento analítico de dados e armazéns de dados, os quais são construídos por meio de sistemas OLAP (*Online Analytical Processing*). O desenvolvimento da resposta consiste em identificar se as asserções I e II são proposições verdadeiras ou falsas e, caso as duas asserções forem verdadeiras, examinar se a asserção II é uma justificativa correta para a asserção I.

A asserção I é uma proposição verdadeira. Os sistemas OLTP (*Online Transaction Processing*) apresentam um modelo de dados tipicamente relacional, orientados à aplicação e destinados a armazenar operações do dia a dia de uma organização, tais como compras, vendas, pagamentos, inventários, faturamento e financeiro. Por outro lado, sistemas OLAP (*Online Analytical Processing*) armazenam os dados em modelos do tipo estrela, floco de neve ou constelação de fatos. Segundo Inmon (1997), o conjunto de dados dos sistemas OLAP é orientado a assunto, integrado, não volátil,

variável em relação ao tempo e destinado para o suporte aos processos de tomada de decisão. Embora sistemas OLTP e OLAP tenham características distintas e são concebidos para finalidades bastante diferenciadas, eles são fortemente correlacionados, pois sistemas OLTP são fontes de dados para sistemas OLAP.

A asserção II também é uma proposição verdadeira. Processos de Extração, Transformação e Carga dos dados, conhecidos como ETL, são responsáveis por recuperar e transformar os dados obtidos a partir das bases de dados dos sistemas OLTP e possuem as funcionalidades a seguir.

- Extração: agrupar os dados obtidos a partir de múltiplas bases de dados heterogêneas.
- Transformação: converter os dados a partir do formato de origem para o formato do sistema OLAP.
- Carga: consolidar os dados, processar visões, verificar integridade e inserir índices.

Pode-se também concluir que a asserção I é justificativa da asserção II, pois a correlação entre OLTP e OLAP ocorre a partir da aplicação dos processos ETL. Desta forma a alternativa A é a correta, ou seja, as asserções I e II são proposições verdadeiras, e a II é uma justificativa correta da I.

REFERÊNCIAS

1. HAN, J. *Data Mining: Concepts and Techniques*. São Francisco: Morgan Kaufmann, 2011. 744 p.
2. INMON, W. H. *Como construir o Data Warehouse*. Rio de Janeiro: Campus, 1997. 388 p.
3. KIMBALL, R.; ROSS, M.; MUNDY, J.; BECKER, B. *The data warehouse toolkit: the definitive guide to dimensional modeling*. 3. ed. Indianapolis (IN): Wiley, 2013. 564 p.

QUESTÃO 16

A virtualização de servidores, acompanhada de solução para Gerenciamento Eletrônico de Documentos (GED), trouxe uma série de benefícios para as escolas de uma instituição: acumulados ao longo de 50 anos, os prontuários dos estudantes passaram a ser acessados em tempo real, as atividades pedagógicas realizadas em ambiente mais sólidos e o gerenciamento centralizado do ambiente passou a permitir que a equipe da instituição se concentre em tarefas estratégicas. No médio prazo, a meta da instituição é virtualizar toda a infraestrutura de servidores e eliminar o uso de papel nos processos, trocando documentos em papel por documentos eletrônicos armazenados nos servidores virtual e acessíveis em qualquer unidade da rede de escolas.

COMPUTERWORLD. *Virtualização*: quadro casos de sucesso no Brasil. Disponível em: <http://cio.com.br>. Acesso em: 29 jul. 2014 (adaptado).

Em relação ao estudo de caso mencionado acima, avalie as asserções a seguir e a relação proposta entre elas.

- I. A tecnologia de virtualização de servidores está proporcionando diversos benefícios para as organizações, como redução de custos e de *downtime* e melhoria no desempenho, na segurança e no gerenciamento da infraestrutura de Tecnologia da Informação (TI). No entanto, sozinha, a virtualização não é suficiente para permitir que as organizações se concentrem em tarefas estratégicas.

PORQUE

- II. A virtualização, se associada a outras tecnologias atuais como GED, *Cloud Computing* e *Grid Computing*, possibilita que as organizações foquem no negócio, eliminando os associados à área de TI.
- A. As asserções I e II são proposições verdadeiras, e a II é uma justificativa correta da I.
 - B. As asserções I e II são proposições verdadeiras, mas a II não é uma justificativa correta da I.
 - C. A asserção I é uma proposição verdadeira, e a II é uma proposição falsa.
 - D. A asserção I é uma proposição falsa, e a II é uma proposição verdadeira.
 - E. As asserções I e II são proposições falsas.

Gabarito: E.

Autores: César A. F. De Rose e Tiago C. Ferreto.

COMENTÁRIO

Nesta questão são feitas afirmações sobre os benefícios tanto da virtualização na gerência de servidores como do processo de digitalização de documentos no contexto de uma instituição de ensino.

A segunda parte da asserção I está correta, pois a Virtualização é apresentada no estudo de caso como a principal responsável por permitir que as organizações se concentrem nas tarefas estratégicas em vez de ficarem se preocupando com a gerência da infraestrutura. No entanto, a primeira parte da asserção I está incorreta, pois menciona melhorias no desempenho como um dos benefícios da virtualização, o que não é o caso, já que a camada de virtualização tem um custo e acaba tendo impacto negativo no desempenho final das aplicações. Sendo assim a asserção I no seu todo é falsa.

A asserção II também é falsa, pois apesar de indicar corretamente a virtualização e o Gerenciamento Eletrônico de Documentos (GED) como responsáveis por permitir que as organizações se concentrem nas tarefas estratégicas em vez de ficarem se preocupando com a gerência da infraestrutura, indica também outros temas muito falados hoje em dia como responsáveis, *Cloud* e *Grid Computing*. Computação em Nuvem (*Cloud*) tem relação com o estudo de caso, pois o acesso de documentos de qualquer unidade da rede escolar é indicado como possível, o que remete para um armazenamento dos dados em nuvem. No entanto, o termo Grade Computacional (*Grid*) está mais relacionado com a forma que os recursos dos provedores são interligados e não está diretamente relacionado com o estudo de caso.

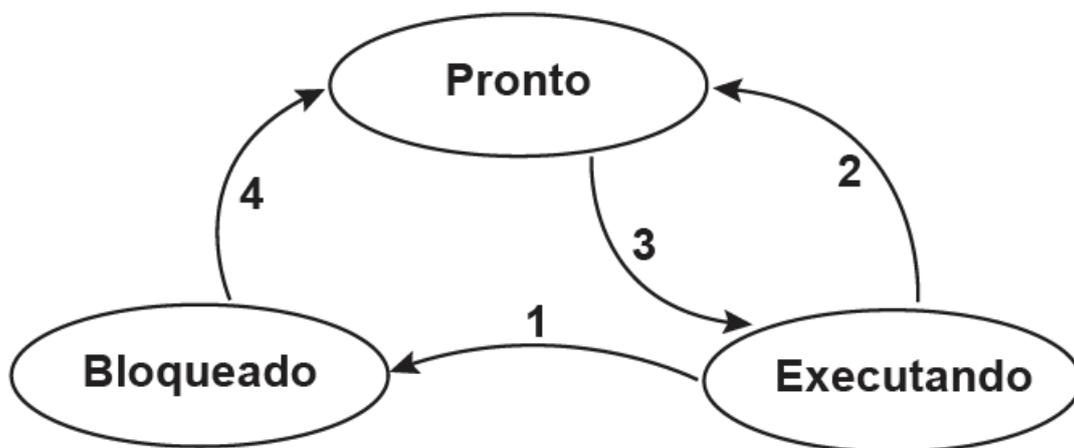
Considerando que as duas asserções são falsas, a alternativa correta é a **E**.

REFERÊNCIA

1. VERAS, Manoel. *Virtualização*. São Paulo: Brasport, 2011.

QUESTÃO 17

Os estados que um processo alcança podem ser modelados por meio do diagrama exibido a seguir.



TANENBAUM, A. S. *Sistemas operacionais modernos*. 3. ed.
São Paulo: Pearson do Brasil, 2010 (adaptado).

No diagrama de estados apresentado, as transições causadas pelo escalonador de processos são

- A. 1 e 2
- B. 1 e 3.
- C. 1 e 4.
- D. 2 e 3.
- E. 2 e 4.

Gabarito: D.

Autores: Carlos Roberto Moratelli e César Augusto Missio Marcon.

COMENTÁRIO

Enquanto um processo é executado, ele transita entre diferentes estados. O estado é definido pela atividade atual do processo. Considere a linha de comando abaixo.

```
cat lista.txt palavras.txt | grep árvore
```

A linha em questão é composta por dois comandos, *cat* e *grep*, que irão gerar dois processos de mesmo nome. O comando *cat* gera como saída a concatenação dos dois arquivos (*lista.txt* e *palavras.txt*). A saída é passada para o comando *grep*, que seleciona todas as linhas com a palavra *árvore*. Veja que o processo *grep* depende do resultado do processo *cat*. Dependendo da velocidade de execução de cada processo, pode ocorrer que o processo *grep* esteja pronto para executar, mas não haja entrada para ele. Assim, ele deve aguardar até que a entrada esteja disponível.

No caso acima, o processo *grep* não pode prosseguir sua execução ficando bloqueado. É possível também que um processo esteja em condições de executar, mas não possa executar imediatamente, pois o sistema operacional decidiu alocar outro processo na CPU. No primeiro caso, o bloqueio é inerente ao problema (não pode processar enquanto não houver entrada). No segundo caso, não existem CPUs suficientes, assim, a CPU precisa ser multiplexada entre os processos existentes. Desta forma, um sistema operacional implementa diferentes estados para representar a atividade atual de seus processos.

A figura mostra que um processo pode estar em um de três estados:

1. em execução (processo executando na CPU);
2. pronto (processo aguardando ser escalonado para executar na CPU);
3. bloqueado (processo aguardando algum evento externo).

Quatro transições são possíveis entre os três estados.

1. O processo não pode prosseguir, pois precisa aguardar por algum evento. Por exemplo, o processo pode ter solicitado uma leitura do disco que apenas pode ser executada pelo sistema operacional. Assim, o processo ficará bloqueado até que o sistema operacional possa atender esta solicitação. **Portanto, a transição 1 é causada pelo próprio processo e não envolve ações do escalonador.**
2. Um processo em execução excedeu o seu tempo na CPU e sofreu uma preempção. Preempção é a interrupção involuntária de um processo para dar lugar a outro processo na CPU. Assim, a transição 2 é causada pela necessidade do sistema operacional de multiplexar a CPU entre os diferentes processos. **A operação de preempção é atribuída e gerenciada pelo próprio escalonador.**
3. A transição 3 ocorre quando o sistema operacional precisa selecionar um processo em estado pronto para ser alocado na CPU. **Esta operação é denominada escalonamento de CPU, onde o escalonador determina o instante da transição, escolhendo o processo pronto conforme uma política de prioridade.**

4. A transição 4 irá ocorrer uma vez que ocorra o evento solicitado por algum processo. Por exemplo, a solicitação de leitura de disco foi finalizada, e os dados solicitados estão disponíveis para o processo solicitante. Durante esta transição, o sistema operacional irá mudar o estado do processo de bloqueado para pronto movendo-o para a fila de escalonamento. **O mecanismo que realiza esta transição não contempla o escalonador.**

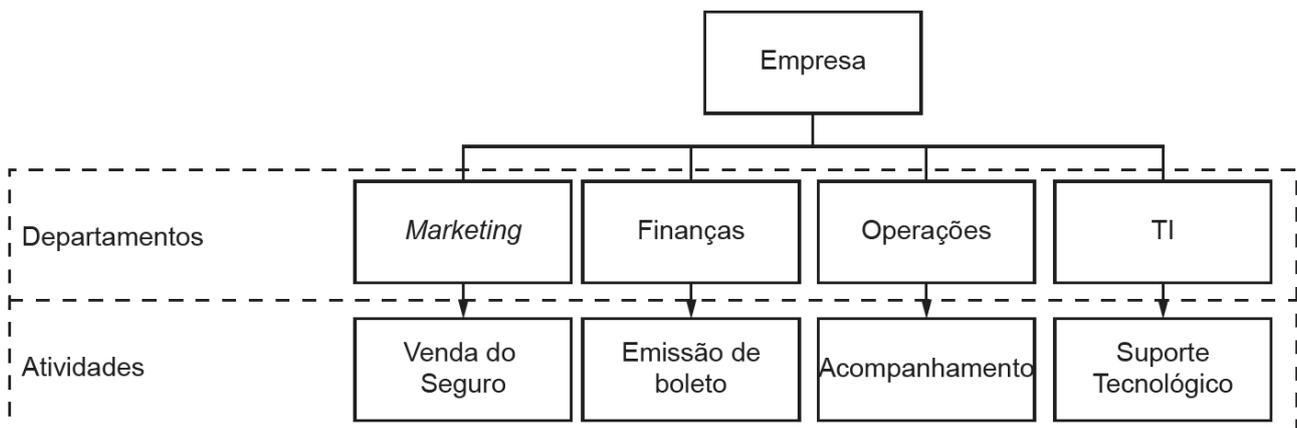
Concluindo, apenas as transições 2 e 3 requerem ações do escalonador.

REFERÊNCIAS

1. SILBERSCHATZ, A.; GALVIN, P.; GAGNE, G. *Sistemas operacionais com Java*. Rio de Janeiro: Elsevier, 2008.
2. TANENBAUM, A. S. *Modern operating systems*. Upper Saddle River: Prentice Hall Press, 2007.

QUESTÃO 18

Uma empresa que comercializa seguros conta atualmente com diferentes sistemas de informação, perfazendo cada um deles uma atividade pelos departamentos de *marketing*, finanças, operações e tecnologia da informação (TI), conforme ilustra a figura a seguir. Considerando a importância estratégica de uma arquitetura empresarial de sistemas de informação funcionalmente integrado, a empresa, após um estudo de viabilidade, identificou que haveria redução significativa de custos caso as atividades fossem gerenciadas como um processo integrado.



Ao estruturar uma arquitetura de negócios orientada à integração do processo, que forma funcional a empresa deve adotar?

- A. Orientação vertical.
- B. Orientação horizontal.
- C. Orientação a processo de TI.
- D. Orientação a processos departamentais.
- E. Orientação a processos colaborativos.

Gabarito: E.

Autor: Sedinei Jose Nardelli Beber.

COMENTÁRIO

Atualmente, a organização (empresa) apresentada possui uma estrutura organizada em quatro funções, representadas pelos seus departamentos: Marketing, Finanças, Operações e TI. O fato de ser uma empresa do ramo de seguros não altera a nova arquitetura, nem é representativa para a resolução da questão, uma vez que a estrutura proposta pode ser útil para inúmeros setores da economia. A questão deixa claro que o estudo de viabilidade identificou uma vantagem (redução

de custos) caso as atividades fossem gerenciadas como processos integrados. Na imagem, a visão dos processos pode ser identificada pela caixa com tracejado englobando as atividades “Venda do Seguro”, “Emissão de Boleto”, “Acompanhamento” e “Suporte Tecnológico”.

Vamos agora às opções.

A. Orientação vertical – representa da forma como está, já vertical, correspondente à cadeia escalar (pirâmide) da organização.

B. Orientação horizontal – em nenhum momento os dados da questão mencionam a possibilidade de começar a diminuir os níveis hierárquicos da organização, de modo que esta não é a alternativa.

C. Orientação a processos de TI – essa área é colocada como um departamento, não alterando de forma estrutural a questão.

D. Orientação a processos departamentais – trata-se da forma como a empresa é organizada inicialmente.

E. Orientação a processos colaborativos – remete a “gerenciadas como um processo integrado”, sendo a opção correta. Os sistemas colaborativos permitem interação indivíduos/grupos, facilita a troca de ideias/recursos e a coordenação de esforços.

REFERÊNCIA

1. O'BRIEN, James; MARAKAS, George. *Administração de Sistemas de Informação*. 15. ed. Porto Alegre: McGrawHill, Bookman, 2013.

QUESTÃO 19

Um analista de sistemas usa um computador para executar apenas dois tipos de operações: processamento lógico/aritmético e acesso a disco.

O computador é um servidor com apenas um processador de um núcleo (core). Além do sistema operacional multitarefa e dos processos do sistema, tal servidor serializa os acessos a disco e é usado para executar apenas os programas que o analista solicita. O analista trabalha com dois tipos de programa: aqueles que realizam em torno de 100% de operações lógicas/aritméticas (quantidade desprezível de acesso a disco), denominados *cpu-bound*, e aqueles que executam em torno de 70% de acesso a disco e 30% de operações lógicas/aritméticas, denominados *io-bound*. Certa vez, o analista executou alguns programas isoladamente e mediu os tempos de execução de cada um. Os dados obtidos foram registrados na Tabela I. Em seguida, ele combinou programas e os submeteu simultaneamente em grupos. Os resultados foram registrados na Tabela II.

Tabela I

Programa	Tipo	Tempo de Execução (ut)
P01	CPU-BOUND	20
P02	CPU-BOUND	40
P03	CPU-BOUND	60
P04	IO-BOUND	50
P05	IO-BOUND	100
P06	IO-BOUND	150

Tabela II

Grupo	Tempo de Execução (ut)
1: P01+P02+P03	123
2: P04+P05+P06	210
3: P03+P05+P06	160
4: Todos	206

Dias depois, revendo as tabelas, ele teve dúvida quanto à correção dos tempos anotados. Considere ut = unidade de tempo.

Com relação a essa situação, avalie as informações a seguir.

- I. É possível que o tempo total do grupo 4 seja 306 ut.
- II. O tempo total do grupo 1 deveria ser, no mínimo, 100 ut.
- III. O tempo total do grupo 2 deveria ser, no mínimo, de 300 ut.
- IV. O tempo total do grupo 3 deveria estar acima de 175 e abaixo de 310 ut.

É correto o que se afirma em

- A. III, apenas.
- B. I e II, apenas.
- C. I e IV, apenas.
- D. II, III e IV, apenas.
- E. I, II, III e IV.

Gabarito: C.

Autores: Edson Ifarraguirre Moreno e Rafael Fraga Garibotti.

COMENTÁRIO

Processos são instâncias de programas aptos a serem executados pelo processador do sistema computacional [1]. Tais processos podem ser classificados como *CPU-bound* ou *IO-bound* de acordo com a utilização dos recursos de computação ou de entrada e saída [2]. Um processo pode ser classificado como *CPU-bound* quando a maior parte de sua execução está vinculada ao uso da unidade de processamento (CPU). Exemplo desse tipo de processo é aquele com aplicação científica, em que a unidade de processamento é muito exigida. Por outro lado, um processo pode ser classificado como *IO-bound* quando passa muito tempo em espera para a realização de operações de leitura e/ou escrita em dispositivos de entrada e saída. Exemplo desse tipo de processo é quando temos uma aplicação que tem bastante iteração com o usuário.

Na Questão 19, os processos *CPU-bound* são referenciadas sendo próximas de 100% de uso de CPU, enquanto as *IO-bound* são referenciadas como tendo aproximadamente 30% de uso de CPU e 70% de operações de entrada e saída. Desta forma, a Tabela I originalmente elaborada, pode ser reescrita, com ilustrado na Tabela 1.

Tabela 1 – Detalhamento dos tempos de CPU e de IO para os diferentes processos.

Programa	Tipo	Tempo de execução (ut)	Tempo de CPU (ut)	Tempo de ES (ut)
P01	CPU BOUND	20	20	-
P02	CPU BOUND	40	40	-
P03	CPU BOUND	60	60	-
P04	IO BOUND	50	15	35
P05	IO BOUND	100	30	70
P06	IO BOUND	150	45	105

Conforme descrito na questão, o sistema computacional é composto por um único núcleo, o que leva a entender que somente um processo pode ser executado por vez. Apesar disto, também é dito que o sistema operacional (SO) é multitarefa, o que significa que o SO é capaz de compartilhar a CPU entre os diferentes processos disponíveis no sistema ao longo do tempo, alternando-os conforme seus estados dentro do sistema computacional [3].

Cada SO define um conjunto de estados, que é utilizado para indicar a possibilidade de uso da CPU. Por exemplo, assumem-se três estados básicos, sendo um processo referente à *execução* da tarefa, outro quando a tarefa está *pronta* para ser executada, e o último quando a tarefa está *bloqueada*, esperando um resultado proveniente de dispositivos de entrada e saída. A Figura 1 ilustra

um exemplo de conjunto de estados e transições entre estes estados [4], consideração básica para o entendimento da questão.

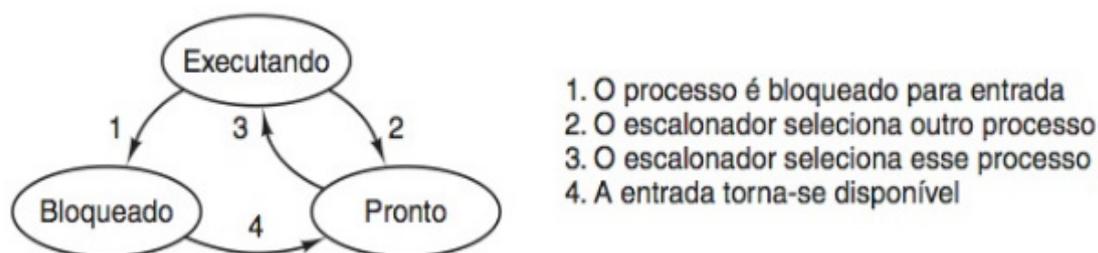


Figura 1 – Figura com a definição de estados e transições entre os estados de um processo [4].

Tais transições apresentadas na Figura 1 são genéricas para o comportamento esperado em sistemas operacionais multitarefa. O estado *executando* representa que um processo está alocando a unidade de processamento. O Estado *bloqueado* pode ser alcançado toda vez que um processo que, por exemplo, estando no estado executando faz uma solicitação de entrada ou saída. O estado de *pronto* é utilizado para sinalizar todos os processos que estão com todas dependências resolvidas e aguardando o uso da CPU.

No sistema computacional proposto para esta questão, apenas um processo pode estar em estado de execução (pois existe somente um *core*), e vários podem estar em estado de pronto ou bloqueado. Toda vez que um processo sai do estado executando, outro pode assumir, ocupando assim a unidade de processamento.

Sendo assim, a Tabela 2 reapresenta os tempos observados (coluna 3) e os tempos que eram esperados para uso de CPU e de entrada e saída (ES) para cada um dos grupos.

Tabela 2 – Tempos de execução observado vs. esperado de uso de CPU e de entrada e saída.

Grupo	Tipo	Tempo de CPU observado (ut)	Tempo de CPU esperado (ut)	Tempo de ES esperado (ut)
1: P01 + P02 + P03	CPU BOUND	123	120	-
2: P04 + P05 + P06	IO BOUND	210	90	210
3: P03 + P05 + P06	CPU + IO BOUND	160	60	-
4: Todos	CPU + IO BOUND	306	210	210

A partir deste ponto, pode surgir uma dúvida com relação à questão elaborada. Como a solicitação de ES é resolvida? Pressupondo a leitura de um arquivo em disco, isto (a) pode ser resolvido pela própria CPU ou (b) pode ser demandado a uma unidade específica de tratamento deste tipo de solução (e.g., DMA) [5]. No primeiro caso, a CPU fica bloqueada resolvendo tal requisição, não permitindo que outro processo acesse a CPU até que a transferência seja finalizada. No segundo caso, a requisição é transferida ao dispositivo de suporte (e.g., DMA) e a CPU fica livre para executar outro processo. Em ambos os casos, o processo requisitante da operação de ES é posto em estado bloqueado. Em sistemas modernos, o uso de DMA é previsto e será pressuposto aqui para esta questão, uma vez que a ocupação da própria CPU para resolver a demanda da tarefa não faria sentido nesta questão.

Na análise das afirmações, temos as conclusões apresentadas a seguir.

I. É possível que o tempo total do grupo 4 seja 306 ut.

Esta afirmação é **verdadeira**, pois, como foi dito, o tempo mínimo de execução é aquele de uso de CPU. A Figura 2 abaixo apresenta uma linha de tempo de uma possível execução. Neste exemplo, é visto que um caso de execução levaria, no mínimo, 210 ut, podendo chegar a 315 ut. Desta forma, 306 ut é totalmente plausível.

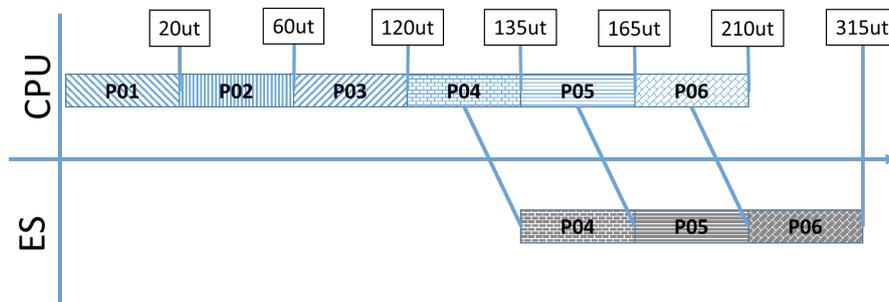


Figura 2 – Proposta de execução de processos para o grupo 4.

II. O tempo total do grupo 1 deveria ser, no mínimo, 100 ut.

Esta afirmação é **falsa**, pois o tempo mínimo de execução é aquele de uso de CPU. Como os três processos são todos *CPU-bound*, a execução é serializada. A Figura 3 apresenta tal execução. Nela, o menor tempo seria de 120 ut.

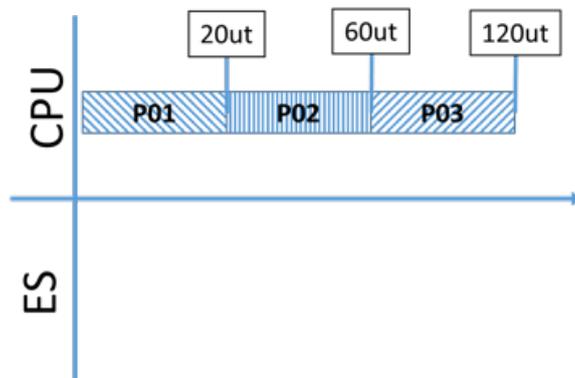


Figura 3 – Tempo de execução do grupo 1.

III. O tempo total do grupo 2 deveria ser, no mínimo, 300 ut.

Esta afirmação é **falsa**. Conforme ilustrado na Figura 4, o grupo 2, formado por processos que demandam tanto processamento quanto entrada e saída podem ser resolvidos em paralelo. Por exemplo, a Figura 4 apresenta uma proposta de execução cujo tempo total consumido foi de 225 ut.

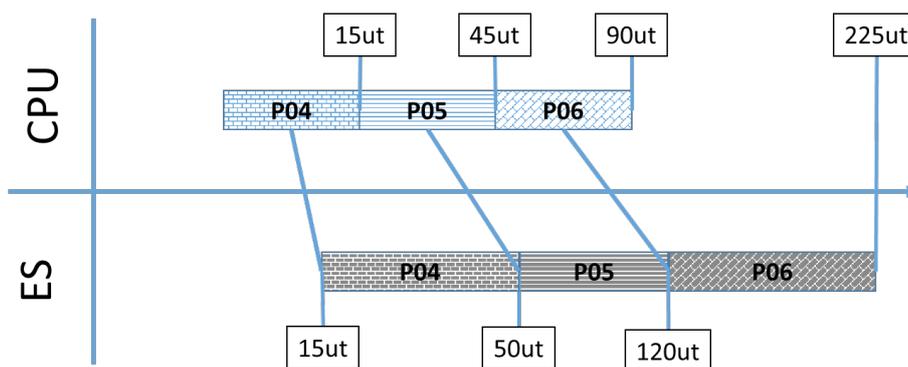


Figura 4 – Tempo de execução do grupo 2.

IV. O tempo total do grupo 3 deveria estar acima de 175 e abaixo de 310 ut.

Esta afirmação é **verdadeira**. A Figura 5 e Figura 6 apresentam, respectivamente, o melhor e o pior caso de execução deste grupo. Na Figura 5, o tempo de execução é de 205 ut. Este caso é considerado o melhor, pois o tempo de fechamento do processo P06, que deve esperar o atendimento à requisição de entrada e saída, paraleliza o uso da CPU para a conclusão do P03.

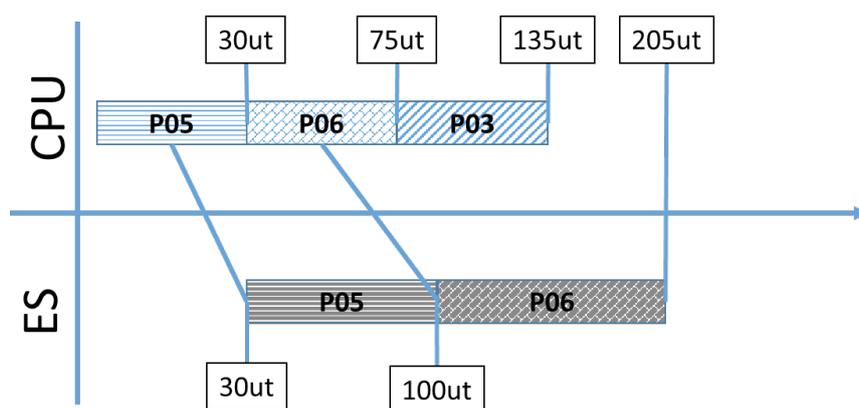


Figura 5 – Melhor caso de execução para o grupo 3.

Na Figura 6, o tempo de execução é de 265 ut. Este caso é considerado um cenário ruim, pois o tempo de fechamento não explora o paralelismo buscado no cenário anterior. Desta forma, o processo P06 deve esperar o atendimento à requisição de entrada e saída, fazendo com que a execução deste grupo seja de 265 ut.

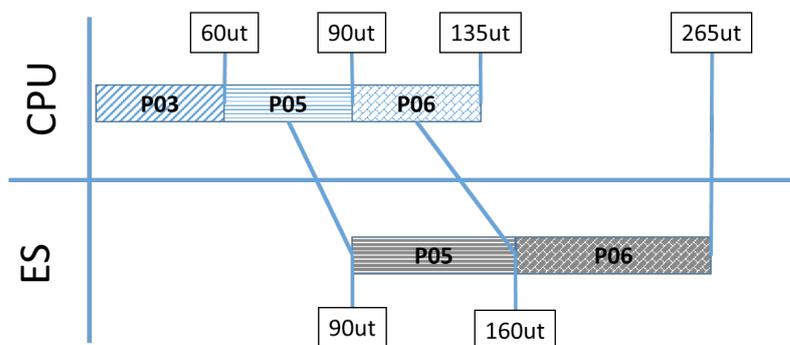


Figura 6 – Caso ruim de execução para o grupo 3.

A questão envolve o entendimento de conceitos relacionados a sistemas operacionais e arquitetura de computadores. O primeiro contribui para o entendimento sobre a gerência de processos. O segundo contribui para o entendimento de arquitetura de *hardware* e as formas de paralelismo entre uso de CPU e recursos de ES. Levando em consideração ambas as áreas, avaliamos que a resposta adequada para esta questão é a opção C, que considera verdadeiras as afirmações **I e IV**.

REFERÊNCIAS

1. MACHADO, F. B.; MAIA, L. P. *Arquitetura de sistemas operacionais*. 5. ed. Rio de Janeiro: LTC, 2013.
2. PATTERSON, D. A.; HENNESSY, J. L.; ASHENDEN, P. J. *Computer organization and design: the hardware/software interface*. 3. ed. Burlington, USA: Morgan Kaufmann, 2007.
3. SILBERCHATZ, A; GALVIN, P. B.; GAGNE, G. *Fundamentos de Sistemas Operacionais*. Rio de Janeiro: LTC, 2013.
4. TANENBAUM, A. S. *Sistemas Operacionais Modernos*. 3. ed. São Paulo: Pearson Prentice Hall, 2009.
5. TANENBAUM, A. S.; WOODHULL, A. S. *Sistemas operacionais: projeto e implementação*. 3. ed. Porto Alegre: Bookman, 2013.

QUESTÃO 20

A internet se encontra organizada em Sistemas Autônomos (AS), que são administrados autonomamente por uma entidade ou organização. Para que uma comunicação ocorra com sucesso entre hospedeiros situados em SAs diferentes, é necessário que pacotes sejam eficientemente transportados do SA origem ao SA destino. Protocolos de roteamento são responsáveis pelo estabelecimento de rotas a serem seguidas pelos pacotes. Para que essas rotas sejam estabelecidas, deve-se usar

- A. *Border Gateway Protocol (BGP).*
- B. *Open Shortest Path First (OSPF).*
- C. *Routing Information Protocol (RIP).*
- D. *Interior Gateway Rounting Protocol (IGRP).*
- E. *Intermediate System to Intermediate System (IS-IS).*

Gabarito: A.

Autoras: Cristina Moreira Nunes e Ana Cristina Benso da Silva.

COMENTÁRIO

Os roteadores são os elementos responsáveis por receber e encaminhar pacotes entre um conjunto de redes de computadores interconectadas. Cada roteador toma decisões de roteamento baseado em informações contidas em sua tabela de roteamento, as quais são atualizadas por protocolos de roteamento.

Na Internet, um conjunto de roteadores e redes podem estar sendo gerenciados por uma única organização (este é o conceito de Sistema Autônomo – AS). Quando roteadores de um mesmo sistema autônomo trocam informações de roteamento entre si, eles utilizam protocolos de roteamento classificados como *Interior Gateway Protocol (IGP)*. Por outro lado, quando roteadores entre sistemas autônomos diferentes trocam informações de roteamento entre si, eles utilizam protocolos de roteamento classificados como *Exterior Gateway Protocol (EGP)*.

Como exemplos de protocolos do tipo IGP, temos:

- *Routing Information Protocol (RIP);*
- *Open Shortest Path First (OSPF);*
- *Interior Gateway Routing Protocol (IGRP);*
- *Intermediate System to Intermediate System (IS-IS).*

Como exemplos de protocolos do tipo EGP, temos:

- *Border Gateway Protocol (BGP);*
- *Exterior Border Protocol (EGP).*

A Questão 20 quer saber qual é o protocolo de roteamento responsável pelo estabelecimento de rotas entre hospedeiros situados em SAs diferentes, ou seja, o protocolo de roteamento do tipo EGP. Dentre as opções apresentadas, a resposta correta é a alternativa **A**, a qual consta o protocolo de roteamento BGP.

REFERÊNCIAS

1. CARISSIMI, A. S.; ROCHOL, J.; GRANVILLE L. Z. *Redes de computadores*. Porto Alegre: Bookman, 2009. 392p.
2. FOROUZAN, B.; FEGAN, S. *Protocolo TCP/IP*. 3. ed. Porto Alegre: Bookman, 2010.
3. KUROSE, J. F.; ROSS, K. W. *Computer networking: a top-down approach*. 6. ed. São Paulo: Pearson, 2014.
4. TANENBAUM, A. *Redes de computadores*. 5. ed. São Paulo: Pearson, 2011. 582p.

QUESTÃO 21

Um gerente de projetos acabou de assumir um projeto em andamento, em que foram identificados vários problemas. Os recursos são limitados e, por isso, é necessário priorizar os problemas a serem solucionados. Nessa situação, a ferramenta que o gerente deve utilizar é

- A. o gráfico de controle.
- B. o diagrama de pareto.
- C. o diagrama de dispersão.
- D. a amostragem estatística.
- E. o diagrama de causa e efeito.

Gabarito: B.

Autoras: Alessandra Costa Smolenaars Dutra e Sabrina dos Santos Marczak.

COMENTÁRIO

Na questão apresentada, o projeto tem vários problemas a serem solucionados, e os recursos são limitados. Nessa situação a ferramenta que o gerente deve utilizar é o diagrama de pareto. **O diagrama de pareto** é construído a partir de um processo de coleta de dados e pode ser utilizado quando se deseja priorizar problemas ou causas relativas a um determinado assunto. O diagrama de pareto leva o nome de Vilfredo Pareto, um economista italiano que desenvolveu a lei de pareto, chamada de Lei 80/20, que determina que, na maioria dos fenômenos conhecidos, 80% das ocorrências originam-se de 20% das causas.

O gráfico de controle é um gráfico que serve para acompanhar a variabilidade de um processo, identificando suas causas comuns (intrínsecas ao processo) e especiais (aleatórias). As causas comuns estão relacionadas ao funcionamento do próprio sistema enquanto que as causas especiais refletem ocorrências fora dos limites de controle. O gráfico permite ainda verificar se os processos estão sendo bem controlados e se há tendência de melhora ou piora de qualidade.

O diagrama de dispersão é um gráfico que permite a identificação entre causas e efeitos, para avaliar o relacionamento entre variáveis. O diagrama de dispersão é a etapa seguinte do diagrama de causa e efeito, pois verifica-se se há uma possível relação entre as causas, isto é, nos mostra se existe uma relação e em que intensidade.

A **amostragem estatística** tem como objetivo fazer generalizações sobre uma população com base nos dados de uma amostra. A população é o conjunto de todos os elementos sob investigação. A amostra é qualquer subconjunto da população.

O **diagrama de causa e efeito**, também conhecido como diagrama de Ishikawa ou diagrama espinha de peixe, é uma ferramenta de representação das possíveis causas que levam a um determinado efeito, utilizada para identificar causas de um problema.

REFERÊNCIAS

1. KOSCIANSKI, A.; SOARES, M. *Qualidade de Software: aprenda as metodologias e técnicas mais modernas para o desenvolvimento de software*. São Paulo: Novatec, 2007.
2. MASHALL, I. *Gestão da qualidade*. 8. ed. Rio de Janeiro: Editora FGV, 2006.
3. ROSE, K. *Project Quality Management: why, what and how*. Fort Lauderdale: J. Ross Publishing, 2005.

QUESTÃO 22

As transações eletrônicas na internet precisam de mecanismo de segurança capazes de garantir autenticidade, confidencialidade e integridade das informações.

Com relação a esse contexto, avalie as afirmações a seguir.

- I. Criptografia assimétrica é um método em que é utilizado um par de chaves: uma pública e uma privada.
- II. Certificado digital é um documento eletrônico assinado digitalmente que permite associar uma pessoa ou entidade a uma chave pública.
- III. Assinatura digital é um método de autenticação de informação digital tipicamente tratado como análogo à assinatura física em papel.
- IV. VPN (*Virtual Private Network*) é um dispositivo de uma rede de computadores por meio do qual se aplica uma política de segurança a determinado ponto de rede.

É correto apenas o que se afirma em

- A. I e II.
- B. I e IV.
- C. III e IV.
- D. I, II e III.
- E. II, III e IV.

Gabarito: D.

Autores: Avelino Francisco Zorzo e Evandro Dalla Vecchia Pereira.

COMENTÁRIO

Esta questão, no seu enunciado, apresenta três das características relacionadas com segurança.

Autenticidade: garante que a mensagem veio de quem realmente se diz ter enviado, sem que venha de alguém diferente do anunciado como fonte.

Confidencialidade: a mensagem só pode ser lida por quem tem o direito à leitura daquela mensagem, ou seja, só pode ler a mensagem quem está autorizado.

Integridade: garante que a mensagem não foi alterada em trânsito, ou seja, possua as mesmas características de quando ela foi enviada.

Ao mesmo tempo, constata-se das informações o que é apresentado a seguir.

- I. A chave pública é divulgada para todos que querem enviar uma mensagem cifrada para o dono daquela chave. Após a mensagem ser cifrada, somente o dono da chave privada, que é

par da chave pública, pode ler a mensagem enviada. Assim, diversas pessoas podem mandar mensagens cifradas para o dono da chave pública, mas somente ele pode ler as mensagens, pois necessita a chave privada para decifrar as mensagens.

- II. O certificado digital é a forma de garantir, por exemplo, que uma chave pública divulgada é realmente da pessoa que está divulgando a chave.
- III. Uma assinatura digital serve para que alguém possa criar uma informação – assinatura, geralmente usando uma chave privada, relacionado a um documento de forma a garantir a autenticidade do documento. Assim, qualquer pessoa que possua a chave pública pode conferir se aquele documento foi realmente assinado pelo dono da chave pública.
- IV. Uma VPN (*Virtual Private Network* – Rede Virtual Privada) é uma rede de comunicação privada construída geralmente sobre uma rede pública, utilizando tunelamento e criptografia para manter os requisitos básicos de segurança, ou seja, autenticidade, confidencialidade e integridade.

Desta forma, as afirmações I, II e III da questão são verdadeiras, enquanto a afirmação IV está é falsa.

REFERÊNCIAS

1. STALLINGS, W. *Cryptography and network security: principles and practice*. 5. ed. New Jersey: Prentice Hall, 2010. 744p.
2. TANENBAUM, A. *Redes de computadores*. 4. ed. Rio de Janeiro: Campus, 2003. 923p.

QUESTÃO 23

Uma companhia com abrangência global decidiu implantar um novo sistema de compartilhamento de informação empresariais. Após estudos de campo, a empresa identificou alguns cenários de utilização das informações. No primeiro cenário, detectou-se que um volume massivo de dados relativos a documentos históricos, normas e regulamentos da organização eram consumidos, em escala global, exclusivamente para leitura. No segundo cenário, verificou-se que documentos eram produzidos de forma colaborativa por equipes regionais de colaboradores. Esses colaboradores, na maioria das vezes, estavam na mesma filial ou geograficamente muito próximos. O terceiro cenário envolvia apenas 10% dos documentos da empresa, acessados concorrentemente por todas as regionais do globo, tanto para leitura quanto para atualização.

Com base nesses cenários e em relação às arquiteturas de *software* distribuídas coerentemente com as necessidades da empresa, avalie as afirmações a seguir.

- I. A utilização de arquitetura *peer-to-peer* (P2P) para disponibilizar os documentos do cenário 1 poderá reduzir o consumo da rede.
- II. O compartilhamento baseado no modelo cliente-servidor com cache no cliente pode aumentar o desempenho da edição colaborativa, sem comprometer a consistência no cenário 2.
- III. A utilização de replicação baseada em cópia primária pode aumentar a escalabilidade do sistema, sem comprometer, de forma significativa, a consistência de dados no cenário 3.
- IV. Se o modelo cliente-servidor de compartilhamento for empregado no cenário 2, a replicação passiva (*backup ou mirroring*) poderá ser utilizada como mecanismo de tolerância a falhas.
- V. A arquitetura *peer-to-peer* baseada em rede *overlay*, tal como o DHT, pode oferecer transparência de migração para os documentos compartilhados no cenário 1.

É correto o que se afirma em

- A. III, apenas.
- B. I e II, apenas.
- C. II, IV e V, apenas.
- D. I, III, IV e V, apenas.
- E. I, II, III, IV e V.

Gabarito: D.

Autores: Tiago Ferreto e Eduardo Henrique P. de Arruda.

COMENTÁRIO

A questão aborda conceitos de sistemas distribuídos referentes ao compartilhamento de informações. São apresentados três cenários variando a quantidade de dados, localização dos acessos e tipo de acesso (leitura ou escrita). As afirmações analisam a eficiência no uso de arquiteturas distribuídas e técnicas de replicação em cada cenário.

A afirmativa **I** indica o uso da arquitetura *peer-to-peer* (P2P) para diminuir o consumo de rede no cenário 1. Este cenário possui como característica uma grande quantidade de dados usados exclusivamente para leitura. Na arquitetura P2P, todos os processos atuam como cliente e servidor. Quando um processo recebe algum dado de outro processo, ele também está apto a servir esse dado para qualquer outro processo da rede. Isso permite que existam diversas cópias do dado na rede. Essa característica dificulta a modificação de dados, pois seria necessário alterar todas as cópias na rede. Desta forma, ela é indicada para acessos do tipo leitura como o do cenário 1. Além disso, a leitura pode ser realizada a partir do processo mais próximo que contenha o dado desejado, minimizando o consumo de rede. Portanto, a afirmativa **I** é **verdadeira**.

A afirmativa **II** indica o uso da arquitetura cliente-servidor com cache no cliente para aumentar o desempenho da edição colaborativa, sem comprometer a consistência, no cenário 2. Este cenário possui como característica a produção de documentos de forma colaborativa por equipes próximas. No modelo cliente-servidor com cache no cliente, o servidor é responsável por prover todos os dados, e o cliente, após realizar o acesso, mantém e atualiza o dado em uma cache local, enviando as modificações em um momento posterior. Essa característica dificulta a edição colaborativa, pois os dados são atualizados inicialmente na cache local, permitindo que existam num certo instante de tempo diversas versões do dado, na cache local, no servidor e nas caches dos outros clientes, comprometendo a consistência. Portanto, a afirmativa **II** é **falsa**.

A afirmativa **III** indica o uso de replicação baseada em cópia primária no cenário 3, visando aumentar a escalabilidade do sistema sem comprometer significativamente a consistência de dados. O cenário 3 consiste em 10% dos documentos que são acessados e modificados de forma concorrente por todas as filiais. A replicação baseada em cópia primária, também chamada de replicação passiva, consiste no uso de diversos servidores para armazenamento redundante dos dados. No acesso para leitura, qualquer servidor pode ser acessado, aumentando o número de acessos suportados e, portanto, a escalabilidade do sistema. No acesso para escrita, existe um servidor que possui a cópia primária do dado (servidor primário), o qual recebe a atualização do cliente e é responsável por replicar a atualização nos demais servidores, chamados de servidores escravos. Durante o processo de atualização dos servidores escravos, existirão servidores com a versão antiga e servidores com a versão nova do dado, o que pode levar ao problema de consistência no acesso ao dado. No entanto, essa inconsistência será temporária, pois eventualmente todos os servidores terão a sua cópia do dado atualizada. Portanto, a afirmativa **III** é **verdadeira**.

A afirmativa **IV** indica que o uso da arquitetura cliente-servidor de compartilhamento usando replicação passiva (*backup* ou *mirroring*) pode ser usada como mecanismo para tolerância a falhas no cenário 2. Este cenário possui como característica a produção de documentos de forma colaborativa por equipes próximas. Na arquitetura cliente-servidor de compartilhamento, o servidor possui todos os dados e qualquer mudança no dado realizada pelo cliente deve ser imediatamente atualizada no servidor. A replicação passiva (abordada na afirmativa **III**) utiliza diversos servidores para armazenamento redundante dos dados. O acesso para leitura pode ser feito em qualquer servidor, enquanto que a atualização deve ser realizada somente no servidor primário. Esse servidor é responsável por manter a consistência entre as cópias do dado nos diversos servidores. No caso de falha de um servidor, os outros servidores poderão servir as requisições de leitura e pode-se definir outro servidor para as atualizações, caso o servidor primário tenha ficado indisponível, aumentando a tolerância a falhas da arquitetura. Portanto, a afirmativa **IV** é **verdadeira**.

A afirmativa V indica que o uso da arquitetura P2P em rede *overlay* do tipo DHT (*Distributed Hash Table*) pode prover a transparência de migração para documentos compartilhados no cenário 1. O cenário 1 possui como característica uma grande quantidade de dados usados exclusivamente para leitura. Conforme já apresentado na explicação da afirmativa I, a arquitetura P2P é propícia para este cenário. O uso de uma rede *overlay* do tipo DHT implica a distribuição de uma tabela de *hashing* pelo conjunto de nós que forma a rede P2P. Essa tabela é usada para definir rapidamente a localização dos documentos e é projetada para suportar a entrada e saída de *peers* da rede, sem gerar inconsistências, provendo a transparência de migração dos documentos compartilhados. Portanto, a afirmativa **V é verdadeira**.

Somente as afirmativas **I, III, IV e V** são **verdadeiras**. Logo, a opção correta é a alternativa **D**.

REFERÊNCIAS

1. COULOURIS, G.; DOLLIMORE, J.; KINBERG, T.; BLAIR, G. *Sistemas distribuídos: conceitos e projetos*. 5. ed. Porto Alegre: Bookman, 2013.
2. TANENBAUM, A. S.; VAN STEEN, M. *Sistemas distribuídos: princípios e paradigmas*. São Paulo: Pearson, 2008.

QUESTÃO 24

Uma empresa mantém um serviço de *software* que atende a centenas de clientes pelo mundo. O serviço é bastante complexo e requer um esforço contínuo de manutenção. Para melhor gerenciar a manutenção do sistema, a empresa decidiu implantar um programa de medição, em que várias métricas foram agregadas, cada uma delas associada a determinada questão a que se desejava responder. O quadro a seguir apresenta as questões e suas métricas associadas.

Questão	Métrica associada
Quantos problemas afetam os clientes?	Número de solicitações de mudanças
Quanto tempo é necessário para se corrigir um problema urgente?	Tempo entre levantamento do problema e aceitação da mudança
Quanto custa uma entrega de manutenção?	Custo por entrega
Como são alocados os custos?	Custo por atividade
Que tipos de mudanças são feitas?	Número de mudanças por tipo
Quanto esforço é despendido por tipo de mudança?	Pessoas-dia gastos, agregados por tipo
Quão difícil é a entrega?	Número de recursos computacionais a serem utilizados

O objetivo atual da empresa é reduzir o prazo de suas entregas, sem comprometer o que foi acordado com o cliente. A métrica que permite avaliar se esse objetivo está sendo atingido é o

- A. custo por entrega.
- B. número de mudança por tipo.
- C. número de solicitação de mudanças.
- D. número de recursos computacionais a serem utilizados.
- E. tempo entre levantamento do problema e aceitação da mudança.

Gabarito: E.

Autores: Leandro Bento Pompermaier e Rodrigo Santos de Espíndola.

COMENTÁRIO

A garantia da qualidade de *software* é uma das principais preocupações no processo de desenvolvimento. Por meio de métricas bem definidas é possível obter informações importantes e com isso auxiliar na condução dos projetos e equipes. Uma métrica pode ser direta ou indireta. As diretas são obtidas usando atributos observáveis, como esforço, tamanho, custo, etc. Enquanto que as indiretas são obtidas por meio de outras métricas (complexidade, confiabilidade, etc.).

A Questão 24 do ENADE pode ser analisada como uma métrica indireta, visto que é necessária a análise de mais de uma métrica para atingir o seu objetivo (reduzir prazo de entrega). Nesse caso, a alternativa **E** indicando uma relação entre o levantamento do problema e o tempo da aceitação da mudança seria a indicada para controlar se o objetivo está sendo atingido.

REFERÊNCIA

1. PRESSMAN, R.; MAXIM, B. *Engenharia de software*. 8. ed. Porto Alegre: McGraw Hill Brasil, 2016.

QUESTÃO 25

No desenvolvimento de um certo produto de *software*, foi determinado que um dos requisitos do produto seria a exibição de informações em planilhas eletrônicas. Contudo, durante uma reunião de revisão, a equipe do projeto verificou que esse formato não facilitava a interpretação das informações e sugeriu que fosse construído um gráfico que permitiria a identificação mais rápida de informações importantes. Diante desse fato, o gerente do projeto deve

- A. autorizar a equipe do projeto a implementar o gráfico e registrar a mudança.
- B. solicitar à equipe do projeto que ignore a sugestão, uma vez que ela está fora do escopo definido.
- C. analisar o impacto da modificação e submetê-la à aprovação do cliente, independentemente da relevância do impacto.
- D. analisar o impacto da modificação e, somente se o impacto no projeto for grande, submetê-la à aprovação do cliente.
- E. autorizar a equipe do projeto a implementar o gráfico e desprezar o registro da mudança, atendendo à recomendação de que sejam rastreadas apenas mudanças cujo impacto seja significativo no custo ou nos prazos do projeto.

Gabarito: C.

Autores: Marcelo Hideki Yamaguti e Jorge Horácio Nicolas Audy.

COMENTÁRIO

O cenário, apresentado na questão, mostra uma situação típica em projetos: mudança de requisitos do produto. A mudança de requisitos pode ter origem externa (por exemplo, proveniente de solicitação do cliente) ou interna (proveniente da própria equipe do projeto). O cenário da questão indica que a origem da mudança veio da equipe.

Em situações de mudança de requisitos, usualmente é realizada uma análise de impacto para se identificar quais alterações serão necessárias no projeto para realizar as ações necessárias. Essas alterações podem impactar, por exemplo, o prazo, o custo ou outra dimensão do projeto, e, por isso, é importante que o cliente tenha ciência desses impactos e aprove (ou não) a mudança. Se a mudança for aprovada, registra-se a mudança, e realizam-se as ações necessárias.

Eventualmente, o gerente de projeto pode ter alçada para aprovar mudanças, dentro de determinados limites, desde que previamente definidos para o projeto. No cenário apresentado não há indicação de que o gerente tem alçada para aprovação de mudanças sem a aprovação do cliente, assim, nessa situação, a alternativa correta é a **C**, “analisar o impacto da modificação e submetê-la à aprovação do cliente, independentemente da relevância do impacto”.

REFERÊNCIAS

1. PROJECT MANAGEMENT INSTITUTE. *A guide to the project management body of knowledge: PMBOK guide*. Newtown Square: PMI, 2013. 589 p.
2. SOMMERVILLE, I. *Engenharia de software*. 9. ed. São Paulo: Pearson Brasil, 2011. 529 p.

QUESTÃO 26

Uma empresa de desenvolvimento de *software* está sendo avaliada no nível E do MPS.BR. O grupo de melhoria de processos fez um levantamento para avaliar as práticas relacionadas à gerência de recursos humanos e obteve as evidências a seguir.

Evidência 1. A empresa possui um processo de avaliação de desempenho e promoção de indivíduos bem estabelecido, em plano de cargos e carreira.

Evidência 2. A empresa contrata externamente instrutores capacitados para realizar treinamentos internos a seus funcionários sempre que verifica a necessidade de ampliar a capacitação da equipe. Esses treinamentos, registrados como parte do portfólio de cada funcionário, são avaliados pelos funcionários em relação à sua qualidade e aderência às necessidades de capacitação de equipe. Cada funcionário também é avaliado ao final de cada treinamento.

De acordo com o Modelo de Referência do MPS.BR, alguns dos resultados esperados para o processo de Gerência de Recursos Humanos são:

GRH 6. Os treinamentos identificados como sendo responsabilidade da organização são conduzidos e registrados.

GRH 7. A efetividade do treinamento é avaliada.

GRH 8. Critérios objetivos para avaliação do desempenho de grupos e indivíduos são definidos e monitorados para prover informações sobre o desempenho e melhorá-lo.

Associação para Promoção da Excelência do Software Brasileiro. SOFTEX. MPS.BR –
Guia Geral MPS de Software. 2012.
Disponível em: <<http://www.softex.br>>. Acesso em: 20 jul. 2014 (adaptado).

Com base no levantamento inicial das práticas da empresa, verifica-se que a evidência 1

- A. atende totalmente ao GRH 8, enquanto a evidência 2 atende totalmente ao GRH 6 e ao GRH 7.
- B. atende parcialmente ao GRH 8, enquanto a evidência 2 atende totalmente ao GRH 6 e ao GRH 7.
- C. atende parcialmente ao GRH 8, enquanto a evidência 2 atende parcialmente ao GRH 6 e ao GRH 7.
- D. atende totalmente ao GRH 8, enquanto a evidência 2 não atende ao GRH 6, mas atende totalmente ao GRH 7.
- E. E. atende parcialmente ao GRH 8, enquanto a evidência 2 atende totalmente ao GRH 6, mas não atende ao GRH 7.

Gabarito: B.

Autoras: Alessandra Costa Smolenaars Dutra e Ana Paula Terra Bacelo.

COMENTÁRIO

Em relação à Evidência 1, “A empresa possui um processo de avaliação de desempenho e promoção de indivíduos bem estabelecido, em planos de cargo e carreira”, verifica-se que atende **parcialmente** ao GRH8 (“Critérios objetivos para avaliação do desempenho de grupos e indivíduos são definidos e monitorados para prover informações sobre este desempenho e melhorá-lo”), porque existe a evidência de que a empresa possui um processo de avaliação, mas não tem evidências de que existam, conforme a NBR ISO, planos de ação para corrigir desvios, ou um plano de treinamento de forma a possibilitar que as habilidades individuais e coletivas da equipe sejam desenvolvidas e, assim, se possa melhorar o desempenho do projeto como um todo (NBR ISO, 2000).

Em relação à Evidência 2, “A empresa contrata externamente instrutores capacitados para realizar treinamentos internos a seus funcionários sempre que verifica a necessidade de ampliar a capacitação da equipe”, esses treinamentos, registrados como parte do portfólio de cada funcionário, são avaliados pelos funcionários em relação à sua qualidade e aderência às necessidades de capacitação da equipe. Cada funcionário também é avaliado ao final de cada treinamento.

Verifica-se que a Evidência 2 atende **totalmente** ao GRH6 (“Os treinamentos identificados como sendo responsabilidade da organização são conduzidos e registrados”), visto que existe a evidência de que a empresa contrata instrutores para realizar e registrar os treinamentos feitos pelos funcionários. Verifica-se ainda que a Evidência 2 atende **totalmente** ao GRH7 (“A efetividade do treinamento é avaliada”), visto que existe a evidência de que o treinamento é avaliado pelo funcionário quanto à sua qualidade e aderência, e também há a avaliação do funcionário em relação ao treinamento. O GRH7 deixa claro que o objetivo desse resultado esperado é avaliar se os treinamentos realizados foram eficientes na transferência de conhecimento para os participantes, atendendo assim totalmente a Evidência 2.

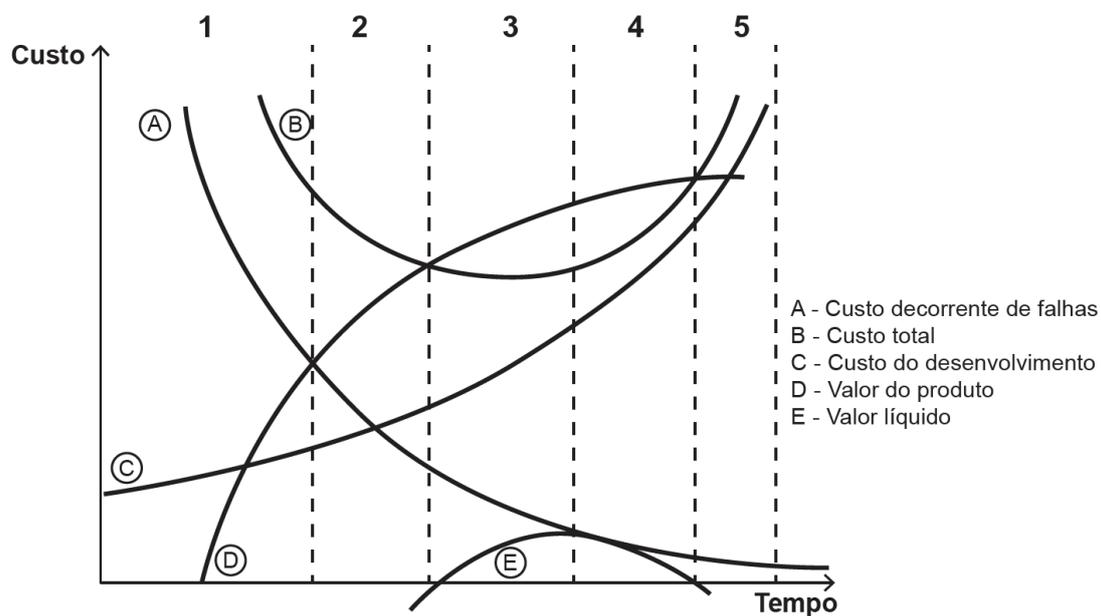
Com isso, a Evidência 1 atende parcialmente a GRH8, enquanto a Evidência 2 atende totalmente o GRH6 e o GRH7.

REFERÊNCIAS

1. ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. NBR ISO 2000, NBR ISO 10006, 2000. *Gestão da qualidade – diretrizes para a qualidade no gerenciamento de projetos*. Rio de Janeiro, 2000.
2. SOFTEX. MPS.BR. *Melhoria do processo de software brasileiro: guia de implementação de software*. Parte 3: Fundamentação para Implementação do Nível E do MR-MPS-SW:2016. Disponível em: <http://www.softex.br/mpsbr/_guias/guias>. Acesso em: 1.º dez. 2016.

QUESTÃO 27

O gráfico a seguir, elaborado pelo diretor de tecnologia da informação de uma empresa, apresenta um modelo para a qualidade de produto de *software* com base no seu custo e no seu valor para a organização.



COHEN, M.; CHARD, J. *Quality Management Best Practices*,
Dr. Dobb's Journal. Novembro 2010.

Disponível em: <<http://drdobbs.com>>. Acesso em: 13 jul. 2014 (adaptado).

A região representada no gráfico em que o produto de *software* deve ser lançado no mercado para permitir que a organização tenha o melhor valor de retorno do investimento é a

- A. 1.
- B. 2.
- C. 3.
- D. 4.
- E. 5.

Gabarito: C.

Autores: Flávio Moreira de Oliveira e Sabrina dos Santos Marczak.

COMENTÁRIO

O gráfico da Questão 27 apresenta a variação de diversos fatores relacionados ao custo e ao valor de um produto de *software* no tempo durante o desenvolvimento. Indicam-se cinco regiões no gráfico, numeradas de 1 a 5 em ordem sequencial no tempo, e pergunta-se qual região corresponderia ao melhor momento para o lançamento do produto no mercado. Analisemos cada região individualmente.

Na região 1, os custos (total e devido a falhas) são máximos e o valor é mínimo; não faz sentido lançar o produto.

Na região 2, o valor aumentou rapidamente, mas ainda não compensa o custo.

Na região 3, o valor compensa o custo total, e o valor líquido é máximo.

Na região 4, o valor ainda compensa o custo total, mas o valor líquido começou a diminuir, devido ao crescimento do custo de desenvolvimento (acumulado).

Na região 5, o custo novamente ultrapassou o valor do produto. Logo, a melhor região é a **3**, indicada na alternativa **C**.

Algumas ressalvas.

- A curva do custo de falhas tem o comportamento esperado para um ciclo de desenvolvimento, lembrando uma distribuição exponencial ou de Weibull. No entanto, se ela continuar ao longo do ciclo de vida do produto, o custo deve voltar a subir perto do final do ciclo.

- Os conceitos de “valor do produto” e “valor líquido” poderiam ser mais bem definidos, visto que podem ser calculados de diferentes maneiras.

- Foi feita uma análise de cada uma das regiões, visando auxiliar a compreensão; no entanto, bastaria observar a curva “E” para responder corretamente, pois corresponde ao valor líquido.

REFERÊNCIAS¹

1. GALIN, D. *Software quality assurance: from theory to implementation*. Harlow: Pearson Education, 2004.
2. KAN, S. H. *Metrics and models in software quality engineering*. 2. ed. Boston: Addison-Wesley Professional, 2009.
3. O'REGAN, Gerard. *Introduction to software quality*. New York: Springer, 2014.

¹ A título de contextualização, as referências abordam o tema da questão. No entanto, cabe ressaltar que a questão, menos que avaliar o conhecimento dos conteúdos, avalia a capacidade de interpretar um gráfico, aliado a um conhecimento mínimo do significado das palavras “custo” e “valor”.

QUESTÃO 28

Conforme a norma ISO 9241, a usabilidade é definida como a capacidade que um sistema interativo de *software* oferece a seus usuários, em um contexto específico de operação, para a realização de tarefas de maneira eficaz, eficiente e agradável. Com fundamento nesse conceito, uma empresa deseja medir a satisfação dos consumidores de seus produtos e, para isso, desenvolverá um sistema de *software*. A equipe de desenvolvimento do sistema de *software* elencou os requisitos listados no quadro abaixo.

Código do Requisito	Descrição do Requisito
RQ1	Cada produto da empresa deverá ser avaliado, por um usuário, com um valor numérico inteiro entre -5 e 5.
RQ2	Cada transação do sistema com o banco de dados para registrar uma avaliação de produto deve ser realizada em um tempo inferior a 20 segundos.
RQ3	Cada operação do usuário deverá ser realizada em até três telas.
RQ4	O sistema deve ter interface com sistemas legados.
RQ5	O usuário consumidor irá identificar-se pelo código do número de Cadastro de Pessoa Física (CPF).

Entre os requisitos listados no quadro, o código do que se refere à usabilidade é

- A. RQ1.
- B. RQ2.
- C. RQ3.
- D. RQ4.
- E. RQ5.

Gabarito: C.

Autoras: Marcia de Borba Campos e Luana Muller.

COMENTÁRIO

RQ1 se refere ao processo de avaliação com usuário utilizando uma escala numérica. Não há indicação de relação com usabilidade.

RQ2 se refere a uma decisão de tempo para coleta de dados, não se refere à usabilidade. Não há explicação de contexto e tão pouco de relação com as metas de usabilidade, por exemplo.

RQ3 se refere ao critério de usabilidade, podendo estar relacionado à eficácia (o quanto um produto é bom em fazer o que se espera dele) e à eficiência (realização de tarefas por meio de um número mínimo de etapas).

RQ4 se refere à integração do sistema com sistemas legados, não se refere à usabilidade.

RQ5 se refere a uma decisão de campo de entrada em um sistema, não se refere à usabilidade.

REFERÊNCIAS

1. ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. *NBR ISO 9241*. São Paulo: Boletim ABNT, 201. 36 p.
2. ROGERS, Y.; SHARP, H.; PREECE, J. *Design de interação: além da interação humano-computador*. Porto Alegre: Bookman, 2013. 599 p.

QUESTÃO 29

Na auditoria externa independente, contratada por uma empresa para verificar o plano de contingência e recuperação de desastres, é necessário analisar

- A. os sistemas de *backup* e a facilidade de recuperação dos dados.
- B. os registros de tentativas de violação de segurança nos sistemas.
- C. os pontos de vulnerabilidade na rede de computadores da empresa.
- D. a documentação dos sistemas de informação da empresa e sua utilidade para os usuários.
- E. os mecanismos de monitoramento dos sistemas de informação que permitam identificar problemas de desempenho.

Gabarito: A.

Autores: Dilnei Venturini e Azriel Majdenbaum.

COMENTÁRIO

A questão apresentada aborda o tema da Segurança da Informação, mais especificamente a gestão da continuidade do negócio através de um plano de contingência e recuperação de desastres. A alternativa **A** está correta. Conforme a ISO/IEC 27002:2013, para a gestão da continuidade do negócio convém que as empresas detalhem e documentem um plano de continuidade do negócio, bem como implementem procedimentos que permitam a recuperação e restauração das operações do negócio e da disponibilidade da informação.

As alternativas **B**, **C**, **D** e **E** estão incorretas, porque relacionam elementos que não estão diretamente relacionados a um plano de contingência e recuperação de desastres, mas a outras áreas da segurança da informação. A alternativa **B** cita registros de tentativa de violação de segurança nos sistemas que estão relacionados à gestão de incidentes de segurança da informação. A alternativa **C** cita pontos de vulnerabilidade na rede de computadores que endereçam o gerenciamento das operações e comunicações. A alternativa **D** cita a documentação dos sistemas de informação da empresa que está ligada a aquisição, desenvolvimento e manutenção de sistemas de informação. E finalmente a alternativa **E** cita os mecanismos de monitoramento dos sistemas de informação que permitam identificar problemas de desempenho, relacionados ao gerenciamento das operações e comunicações.

REFERÊNCIA

1. ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. *ISO/IEC 27002:2013: Tecnologia da Informação – Técnicas de Segurança – Código de Prática para controles de segurança da informação*. São Paulo, 2013.

QUESTÃO 30

Considere que uma empresa que presta serviços de transporte de pacientes em ambulâncias para clientes conveniados disponha de um sistema de controle e gerenciamento de atendimentos e viagens realizados. Considere, ainda, que, em atendimento, é utilizada uma ambulância e são registrados a data e o convênio a que o atendimento está vinculado. Em um atendimento, uma ambulância realiza uma ou mais viagens e, a cada viagem, é incrementado um número sequencial que começa em 1. Nessa situação, o esquema relacional simplificado, mostrado a seguir, foi projetado para suportar um banco de dados que controle a operação. No esquema, as chaves primárias têm seus atributos componentes sublinhados.

Paciente	(<u>CodPaciente</u> , Nome, Endereço)
Convênio	(<u>CodConvênio</u> , Empresa, Plano)
Atendimento	(<u>CodAtendimento</u> , CodPaciente, CodConvênio, Data, Finalidade)
Viagem	(<u>CodAtendimento</u> , Sequência, Origem, Destino)

Com base nas informações e no esquema apresentado, avalie as afirmações a seguir.

- I. CodConvênio é uma chave estrangeira na tabela Atendimento.
- II. CodAtendimento não pode ser chave estrangeira na tabela Viagem, porque faz parte da chave primária.
- III. CodPaciente nunca pode assumir valores nulos na tabela Atendimento, porque é uma chave estrangeira.
- IV. CodPaciente → Nome, Endereço; CodConvênio → Empresa, Plano; CodAtendimento → CodPaciente, CodConvênio, Data, Finalidade; CodAtendimento, Sequência → Origem, Destino; são dependências funcionais corretamente deduzidas.

É correto apenas o que se afirma em

- A. I.
- B. II.
- C. I e IV.
- D. II e III.
- E. III e IV.

Gabarito: C.

Autores: Rodrigo Coelho Barros e Rodrigo Dias Rivera.

COMENTÁRIO

Esta questão trata sobre modelagem de banco de dados relacional. É apresentado um esquema relacional simplificado para um cenário de transporte de pacientes conveniados em ambulâncias, modelado para atender às necessidades de um sistema de controle e gerência de atendimentos e viagens. A alternativa correta é a letra **C**: afirmações **I** e **IV** são as únicas corretas.

A afirmação **I** diz que CodConvênio é chave estrangeira na tabela Atendimento. Tal afirmativa é verdadeira, pois CodConvênio, que é chave primária da tabela Convênio, consta como um dos atributos da tabela Atendimento, sendo responsável pelo relacionamento entre as tabelas Convênio e Atendimento.

A afirmação **II** diz que CodAtendimento não pode ser chave estrangeira da tabela Viagem porque faz parte da chave primária da mesma tabela. Esta afirmativa é falsa, pois não existe nada que impeça que colunas de uma chave primária composta de determinada tabela também atuem como chave estrangeira na modelagem de banco de dados relacional. É exatamente este cenário que ocorre: CodAtendimento é parte de uma chave primária composta da tabela Viagem, mas funciona também como chave estrangeira, relacionando a tabela Viagem com a tabela de Atendimento.

A afirmação **III** diz que CodPaciente nunca pode assumir valores nulos na tabela Atendimento, porque é chave estrangeira. Esta afirmativa é falsa, pois é perfeitamente possível que uma chave estrangeira assuma valores nulos. A única restrição que existe sobre valores nulos é para chaves primárias, cujos valores nunca podem ser nulos.

A afirmação **IV** mostra as dependências funcionais deduzidas a partir das quatro tabelas do problema. O modelo relacional pegou emprestado o conceito de dependências funcionais da teoria de funções da matemática. De maneira simplificada, o conceito de dependência funcional diz que a dependência existe se a partir de determinada informação (ou informações), conseguimos recuperar uma outra informação (informações). Por exemplo: **cpf** → **nome** indica que, a partir de um número de **cpf**, eu consigo encontrar o **nome** correspondente da pessoa. Ou ainda, em outras palavras, que **nome** depende funcionalmente de **cpf**. Perceba que todas as dependências funcionais da alternativa **IV** estão corretas, pois indicam as chaves primárias no lado esquerdo do operador de dependência (→), e os demais atributos das respectivas tabelas no lado direito do operador de dependência. De fato, a partir das chaves primárias de cada tabela conseguimos ter acesso aos demais atributos.

REFERÊNCIAS

1. HEUSER, C. *Projeto de banco de dados*. 6. ed. Porto Alegre: Sagra-DC Luzzatto, 2009. 282 p.
2. SILBERSCHATZ, A.; KORTH, H. F.; SUDARSHAN, S. *Sistema de bancos de dados*. 6. ed. Rio de Janeiro: Campus, 2012. 904 p.

QUESTÃO 31

Um sistema de informação usa um banco de dados relacional que possui tabelas cujos esquemas em SQL estão representados a seguir.

```
CREATE TABLE Artista
(
    id INTEGER PRIMARY KEY,
    nome VARCHAR (40) NOT NULL,
    CPF CHAR (11) NOT NULL,
    UNIQUE (CPF)
);
CREATE TABLE Evento
(
    id INTEGER PRIMARY KEY,
    descricao VARCHAR (60) NOT FULL,
    numMaxConvidados INTEGER DEFAULT 0,
    CHECK (numMaxConvidados >= 0)
);
CREATE TABLE Atuacao
(
    idArtista INTEGER,
    idEvento INTEGER,
    PRIMARY KEY (idArtista, idEvento),
    FOREIGN KEY (idArtista) REFERENCES Artista,
    FOREIGN KEY (idEvento) REFERENCES Evento (id)
);
```

O sistema também possui uma consulta que integra um de seus relatórios, conforme indicado a seguir.

```
SELECT      A. nome, E.descricao
FROM        Evento E FULL JOIN Atuacao T ON E. id = T.idEvento
           FULL OUTER JOIN Artista A ON T. idArtista = A.id
```

Considerando que todas as tabelas possuem dados, o resultado da consulta utilizada no relatório é

- A. o nome de todos os artistas combinados com a descrição de todos os eventos.
- B. a descrição de todos os eventos e, caso haja artistas alocados, os seus nomes.
- C. o nome de todos os artistas e a descrição de todos os eventos em que eles atuam.
- D. o nome de todos os artistas e, caso eles participem de eventos, a descrição do evento.
- E. o nome de todos os artistas, a descrição de todos os eventos e, caso eles se relacionem, os dois combinados.

Gabarito: E.

Autores: Duncan D. Ruiz e Daniel Callegari.

COMENTÁRIO

É uma questão que trata de consultas em SQL e, em especial, junções de tabelas.

É apresentado um esquema SQL onde são definidas as tabelas Artista, Evento e Atuacao, onde esta última faz referência às outras duas, por integridade referencial.

Sobre este esquema, é questionado o resultado da consulta SQL a seguir:

```
SELECT      A.nome, E.descricao
FROM        Evento E FULL JOIN Atuacao on E.id = T.idEvento
           FULL OUTER JOIN Artista A on T.idArtista = A.id
```

Neste comando são selecionadas a coluna nome de Artista e descrição de Evento. A forma de junção das tabelas é por junção externa (FULL JOIN e FULL OUTER JOIN). Como a junção externa é completa (FULL), é feita a junção externa para ambos os lados: esquerdo e direito. A cláusula OUTER não faz diferença na consulta, já que FULL implica junção externa.

Logo, a consulta retorna o nome de todos os artistas, a descrição de todos os eventos e, quando eles estão relacionados por Atuação, artistas combinados com Eventos.

A alternativa correta é a **E**.

Para a alternativa **A**, o comando SQL correspondente seria:

```
SELECT      A.nome, E.descricao
FROM        Evento E, Artista A
```

Nesse caso, é feito o produto cartesiano entre as tabelas Evento e Artista.

Para a alternativa **B**, o comando SQL correspondente seria:

```
SELECT      A.nome, E.descricao
FROM        Evento E LEFT JOIN Atuacao on E.id = T.idEvento
           LEFT JOIN Artista A on T.idArtista = A.id
```

Nesse caso, é feita a junção externa à esquerda entre as tabelas Evento e Artista.

Para a alternativa **C**, o comando SQL correspondente seria:

```
SELECT      A.nome, E.descricao
```

```
FROM      Evento JOIN Atuacao on E.id = T.idEvento
          JOIN Artista A on T.idArtista = A.id
```

Nesse caso, é feita a junção interna entre as tabelas Evento e Artista.

Para a alternativa **D**, o comando SQL correspondente seria:

```
SELECT    A.nome, E.descricao
FROM      Evento E RIGHT JOIN Atuacao on E.id = T.idEvento
          RIGHT JOIN Artista A on T.idArtista = A.id
```

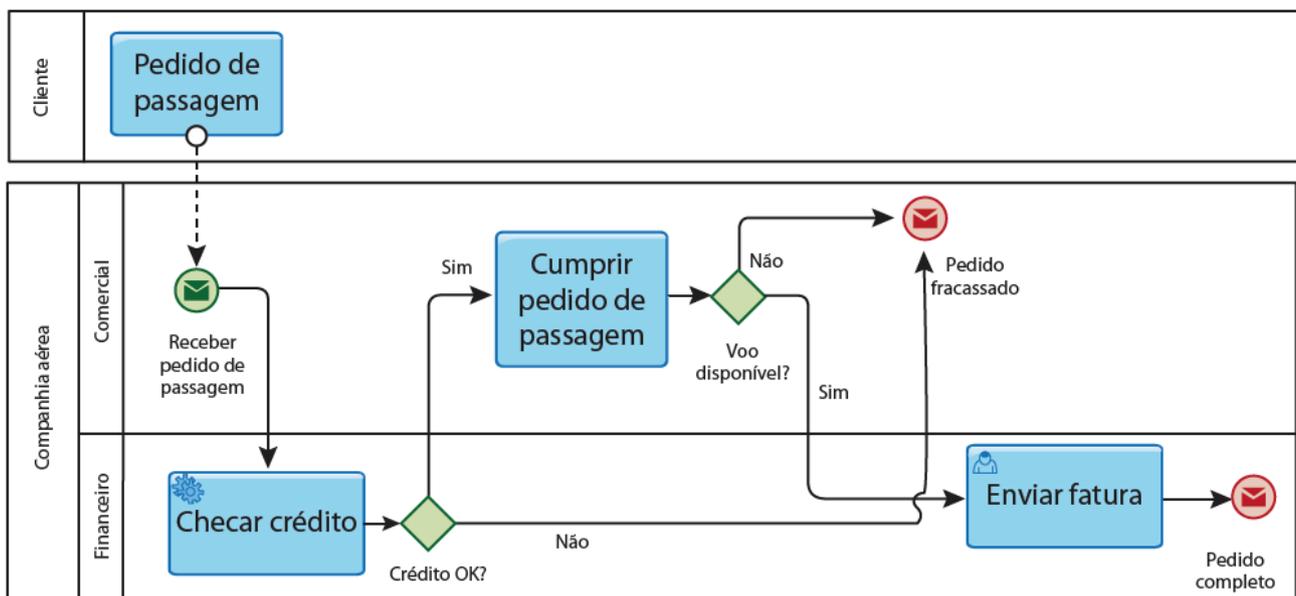
Nesse caso, é feita a junção externa à direita entre as tabelas Evento e Artista.

REFERÊNCIAS

1. BOWMAN, J.; EMERSON, S.; DARNOVSKY, M. *The practical SQL handbook: using SQL variants*. 4. ed. Boston: Addison-Wesley, 2001.
2. DATE, C. J. *Introdução a sistemas de bancos de dados*. Rio de Janeiro: Campus, 2004.
3. ELMASRI, R.; NAVATHE, S. B. *Sistemas de banco de dados*. 6. ed. São Paulo: Pearson Brasil, 2012.
4. PRICE, J. *Oracle database 11g SQL*. Porto Alegre: Bookman, 2008.
5. PUGA, S.; FRANÇA, E.; GOYA, M. *Banco de dados: implementação em SQL, PL/SQL e Oracle 11g*. São Paulo: Pearson, 2014. 332 p.
6. RAMAKRISHNAN, R. *Database management systems*. 3. ed., Boston: McGraw-Hill, 2008.
7. SILBERSCHATZ, A.; KORTH, H. F.; SUDARSHAN, S. *Sistema de bancos de dados*. 6. ed. Rio de Janeiro: Campus, 2012.

QUESTÃO 32

Considere um cenário de venda de passagens aéreas cujo processo de negócio apresenta-se modelado no diagrama abaixo, usando-se BPMN (*Business Process Modeling and Notation*).



Para que essa companhia aérea obtenha mais retorno, será necessário

- A. retirar o evento intermediário de tempo colocado após a tarefa “Pedido de passagem”.
- B. adicionar uma tarefa para oferecimento de outro voo na raia “Comercial”.
- C. modelar a tarefa “Enviar fatura” como tarefa de serviço (*service task*).
- D. transformar o *gateway* “Crédito OK?” em um *gateway* inclusivo.
- E. transformar a tarefa “Checar crédito” em um subprocesso.

Gabarito: B.

Autores: Ricardo Melo Bastos e Rodrigo Espindola.

COMENTÁRIO

A questão apresenta um diagrama em BPMN envolvendo um processo de aquisição de passagem aérea. O foco da questão está associado à Companhia Área, sendo todas as respostas apresentadas referentes a esta piscina no diagrama BPMN.

O questionamento é sobre de que forma a companhia aérea poderia ter mais retorno. Para responder a essa pergunta, é preciso inicialmente identificar onde o processo não se encerra com

sucesso. Isso fica caracterizado quando o processo se encerra no evento final “Pedido fracassado”. Portanto, a resposta correta deve apresentar algum tratamento ao problema que minimize a possibilidade de o processo se encerrar nesse evento final.

Logo o que se deseja é que o processo se encerre no evento final “Pedido completo”. Considerando as alternativas de resposta apresentadas para a questão, a única que traz uma solução plausível para o problema é a alternativa **B** (adicionar uma tarefa para oferecimento de outro voo na raia “Comercial”). Essa resposta propõe a inclusão de uma nova tarefa após o gateway “Voo disponível?”, criando um fluxo alternativo para o caso de não haver disponibilidade no voo inicialmente desejado pelo Cliente. Cria-se assim a possibilidade de oferecer ao cliente um voo alternativo para realização de sua viagem. Todas as demais respostas referem-se a aspectos irrelevantes para o problema em questão.

REFERÊNCIAS

1. FREUND, Jakob; RUCKER, Bernd. *Real-Life BPMN: Using BPMN 2.0 to Analyze, Improve, and Automate Processes in Your Company*. 2. ed. Berlim: Createspace Pub, 2014. 280 p.
2. SILVER, Bruce. *BPMN Method & Style: with BPMN implementer’s guide*. 2. ed. New York: Cody-Cassidy Press, 2011.

QUESTÃO 33

Um dos recursos utilizados no processo de inclusão digital por meio da utilização de Tecnologias de Informação e Comunicação (TIC) é a implantação de Telecentros Comunitários.

A esse respeito, avalie as asserções a seguir e a relação proposta entre elas.

- I. Proporcionar à população mais carente o acesso às tecnologias de informação por meio de Telecentros Comunitários e mediante a capacitação para a utilização desse conhecimento favorece a promoção da cidadania.

PORQUE

- II. Telecentros Comunitários possibilitam à população carente o acesso a informações e conhecimentos, estimulam a criatividade e garantem a inserção profissional no mercado de trabalho.

A respeito dessas asserções, assinale a opção correta.

- A. As asserções I e II são proposições verdadeiras, e a II é uma justificativa correta da I.
B. As asserções I e II são proposições verdadeiras, mas a II não é uma justificativa correta da I.
C. A asserção I é uma proposição verdadeira, e a II é uma proposição falsa.
D. A asserção I é uma proposição falsa, e a II é uma proposição verdadeira.
E. As asserções I e II são proposições falsas.

Gabarito: C.

Autoras: Lúcia Giraffa e Márcia Campos.

COMENTÁRIO

As políticas públicas criadas ao longo dos últimos 20 anos relacionadas à Informática na Educação e Ações inclusivas contemplam a criação destes centros como vetores de Inclusão Digital e promoção da cidadania. Entretanto os Telecentros Comunitários não podem garantir que o participante tenha inserção no mercado de trabalho. O centro pode contribuir para o usuário ter acesso às informações, participar de ações e tarefas que lhe auxiliem a promover processo criativos.

REFERÊNCIAS

1. BARRETO, Angela Maria; PARADELLA, Maria Dulce; ASSIS, Sônia. Bibliotecas públicas e telecentros: ambientes democráticos e alternativos para a inclusão social. *Ciência da Informação*, v. 37, n. 1, 2008.
2. MARTINI, Renato. Inclusão digital & inclusão social. *Inclusão social*, v. 1, n. 1, 2005.

QUESTÃO 34

Nos órgãos públicos, o desafio da gestão do conhecimento é contribuir para a modernização da gestão pública no contexto de um Estado, de forma a garantir-se a participação de toda a sociedade nas atividades desenvolvidas por seus representantes no Executivo, Legislativo e Judiciário.

Considere soluções de TI para a implementação de uma política pública cujo objetivo seja proporcionar maior transparência aos gastos de um órgão público e que, para tal, tenha se optado por uma solução baseada em Web Semântica, sendo uma das fases para seu desenvolvimento a disponibilização dos dados referentes aos gastos do órgão.

Considerando esse cenário, avalie as seguintes asserções e a relação proposta entre elas.

A publicação de dados na forma de dados abertos vinculados (*Linked Open Data – LOD*) é apropriada para uso generalizado dos dados no âmbito da Web Semântica.

PORQUE

Um dos papéis do formato LOD para a Web Semântica é facilitar a integração de dados quando houver ambiguidade de termos utilizados nos diferentes conjuntos de dados.

A respeito dessas asserções, assinale a opção correta.

- A. As asserções I e II são proposições verdadeiras, e a II é uma justificativa correta de I.
- B. As asserções I e II são proposições verdadeiras, mas a II não é uma justificativa correta da I.
- C. A asserção I é uma proposição verdadeira, e a II é uma proposição falsa.
- D. A asserção I é uma proposição falsa, e a II é uma proposição verdadeira.
- E. As asserções I e II são proposições falsas.

Gabarito: C.

Autores: Renata Vieira e Rafael Bordini.

COMENTÁRIO

O *World Wide Web Consortium (W3C)* é uma comunidade internacional que desenvolve padrões Web com a missão de fazer a web explorar seu grande potencial. O W3C desenvolve, para isso, uma base tecnológica para a criação de uma Web de dados, a fim de permitir o desenvolvimento de sistemas que suportem interações confiáveis na rede mundial de computadores. Essa rede de dados interligados em larga escala é chamada *Linked Data*. *Linked Open Data* refere-se a essa integração de dados da web, considerando-se dados disponibilizados por uma licença aberta, que não impede seu reúso.

A Web Semântica refere-se à visão (ampliada e potencial) dessa rede de dados interligados. Um conjunto de tecnologias permite a criação de repositórios de dados na web, construção de vocabulários e regras para manipular esses dados (por exemplo, RDF, SPARQL, OWL, SKOS). Essas tecnologias permitem a conversão ou acesso *on-the-fly* a bases de dados existentes.

Na Web Semântica, são os vocabulários que definem os conceitos e relacionamentos usados para descrever e representar uma área de interesse. Vocabulários podem ser simples ou complexos. Muitas vezes uma ontologia pode ser empregada, nesse caso quando se tem uma coleção de termos mais complexa e formal. Vocabulários mais simples são usados quando esse formalismo mais complexo não é necessário. O papel dos vocabulários, portanto, é o de auxiliar na integração dos dados, tratando os casos de ambiguidade de termos usados em diferentes bases de dados por exemplo.

Conforme os conceitos apresentados acima, podemos observar que a afirmação I é verdadeira. A afirmação II, no entanto, não é verdadeira, uma vez que o *Linked Open Data* em si não resolve o problema de ambiguidade dos dados, para isso é necessário o emprego de vocabulários especializados ou ontologias.

REFERÊNCIA

1. W3C. Disponível em: <<http://www.w3.org/standards/semanticweb/>>. Acesso em: 20 mar. 2017.

QUESTÃO 35

Way back in the late twentieth century, 1997 to be exact, the Object Management Group (OMG) released the Unified Modeling Language (UML). One of the purposes of UML was to provide the development community with a stable and common design language that could be used to develop and build computer applications. UML brought forth a unified standard modeling notation that IT professionals had been wanting for years. Using UML, IT professionals could now read and disseminate system structure and design plans, just as construction workers have been doing for years with blueprints of buildings.

It is now the twenty-first century, 2003 to be precise, and UML has gained traction in our profession. On 75 percent of the resumes I see, there is a bullet point claiming Knowledge of UML. However, after speaking with a majority of these job candidates, it becomes clear that they do not truly Know UML. Typically, they are either using it as a buzz word, or they have had a sliver of exposure to UML.

BELL, D. *UML basics: an introduction to the unified modeling language*. IBM Corporation, 2003. Disponível em: <<http://www.ibm.com>>. Acesso em: 28 jul. 2014 (adaptado).

Com base no texto e nos conceitos de UML, avalie as afirmações a seguir.

- I. Na maioria dos currículos analisados pelo autor, existe alguma indicação de conhecimento da linguagem UML pelo candidato ao emprego.
- II. A UML é composta por diversos diagramas, como diagrama de caso de uso, diagrama de especificação e diagrama de componentes.
- III. A UML não é uma linguagem de programação, mas, sim, uma linguagem de modelagem padronizada para os profissionais de TI cuja notação independe de processos.

É correto o que se afirma em

- A. I, apenas.
- B. II, apenas.
- C. I e III, apenas.
- D. II e III, apenas.
- E. I, II e III.

Gabarito: C.

Autores: Ana Paula Terra Bacelo e Marcelo Hideki Yamaguti.

COMENTÁRIO

A questão apresenta um texto, em inglês, relacionado ao contexto histórico da linguagem de modelagem unificada – UML – e à percepção do autor em relação ao seu uso no mercado e também à sua inserção nos currículos dos cursos de tecnologia da informação. Observa-se que as habilidades avaliadas nesta questão são de cunho técnico e de interpretação de textos na língua inglesa.

Com relação às afirmações:

- I. está correta, pois no texto da questão, em seu último parágrafo, o autor afirma que, em 75% dos currículos de candidatos a empregos, há indicação de conhecimento de UML.
- II. está incorreta, pois na mesma é citado “diagrama de especificação”, sendo que esse não existe na UML. Os diagramas que compreendem a UML, versão 2, são: casos de uso, atividades, comunicação, sequência, classes, objetos, componentes, estados, pacotes, implantação, estruturas compostas, perfil, tempo e interatividade.
- III. está correta, pois, no seu próprio nome (Linguagem de Modelagem Unificada) já se destaca a questão de unificação e padronização para a modelagem (não programação), sendo que ela pode ser aplicada para qualquer tipo de problema, independente da sua complexidade e processo.

Por isso que a resposta correta é a alternativa **C**.

REFERÊNCIAS

1. ARLOW, J.; NEUSTADT, I. *UML 2 and the Unified Process: practical object-oriented analysis and design*. 2. ed. Upper Saddle River: Addison Wesley, 2005. 592 p.
2. BOOCH, G.; RUMBAUGH, J.; JACOBSON, I. *UML: guia do usuário*. 2. ed. Rio de Janeiro: Elsevier, 2006. 474 p.
3. OBJECT MANAGEMENT GROUP. Disponível em: <<http://www.uml.org/>>. Acesso em: 30 mar. 2017.