



# **TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS**

**MATERIAL INSTRUCIONAL ESPECÍFICO**

**TOMO 5**

# **CQA - COMISSÃO DE QUALIFICAÇÃO E AVALIAÇÃO**

## **TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS**

### **MATERIAL INSTRUCIONAL ESPECÍFICO**

#### **TOMO 5**

*Material instrucional específico, cujo conteúdo integral ou parcial não pode ser reproduzido ou utilizado sem autorização expressa, por escrito, da CQA/UNIP – Comissão de Qualificação e Avaliação da UNIP – UNIVERSIDADE PAULISTA.*

## Questão 1

### Questão 1.<sup>1</sup>

Leia o texto a seguir.

*Várias técnicas relacionadas à programação extrema (XP) são diretamente ligadas ao código, incluindo a refatoração, programação em pares e integração contínua. A programação em pares é a prática preferida dos desenvolvedores XP trabalhando em pares em um computador.*

SCOTT, K. *O Processo Unificado Explicado*. Porto Alegre: Artmed, 2002. BECK, K. *Programação Extrema (XP) Explicada*. Porto Alegre: Artmed, 2000.

A programação em pares auxilia no desenvolvimento de código de melhor qualidade quando os pares

- A. elaboram e utilizam padrões de codificação conjuntamente, os quais, quando utilizados corretamente e apropriadamente, reduzem problemas individuais.
- B. estão acostumados ao desenvolvimento e à propriedade coletiva, limitando-se a fazer pequenas mudanças na ocorrência de erros em tempo de execução.
- C. minimizam os riscos de insucesso no projeto através da utilização de ferramentas para a geração automática de testes funcionais e protótipos de interface.
- D. escrevem testes em separado e discutem os resultados posteriormente, o que dá a eles a chance de se sintonizarem antes de começarem a implementação.
- E. trabalham em projetos complexos onde a codificação seja desenvolvida de forma conjunta, minimizando erros e agregando valor onde quer que o sistema necessite.

### 1. Introdução teórica

#### Programação extrema (XP)

Programação extrema (XP) é um processo de desenvolvimento de software que tem, como princípios, a comunicação (com uso mínimo de documentação e uso máximo de interação entre as pessoas envolvidas no projeto), a simplicidade (o objetivo é buscar as soluções mais simples para resolver os problemas, contribuindo também para minimizar os custos com as manutenções futuras), o feedback (contribui para que todas as pessoas envolvidas no projeto aprendam cada vez mais sobre o sistema) e a coragem (é necessária para aplicar o processo corretamente).

Todo código criado em XP é escrito por um par de programadores: um coloca seu foco na codificação, pensando no algoritmo e na lógica de programação; e o outro observa o

---

<sup>1</sup>Questão 10 - Enade 2011.

código e tenta encontrar uma forma estratégica de simplificá-lo, apontando erros decorrentes das possíveis falhas.

## **2. Análise da questão**

O trabalho de elaboração de código e o trabalho da determinação dos padrões de codificação são feitos em conjunto, o que contribui para reduzir problemas individuais.

Alternativa correta: A.

## **3. Indicações bibliográficas**

- PRESSMAN, R. S. *Engenharia de software*. 6. ed. São Paulo: McGraw-Hill, 2006.
- SOMMERVILLE, I. *Engenharia de software*. 8. ed. São Paulo: Pearson, 2007.

## Questão 2

### Questão 2.<sup>2</sup>

A linguagem gráfica EPC/ARIS é utilizada para modelar processos de negócios. Para isso, utiliza diversos recursos para descrever, representar ou indicar, por exemplo, atividades, funções, processos e fluxos. Nesse contexto, avalie as afirmativas a seguir.

- I. A ligação entre dois processos é indicada por um conector.
- II. A descrição de um processo deve iniciar e terminar em um evento.
- III. As funções, ou atividades, são representadas por um retângulo com bordas arredondadas.

É correto apenas o que se afirma em

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

## 1. Introdução teórica

### Processo

Processo é a colaboração de várias atividades comuns para a realização de um objetivo global, focado no cliente final.

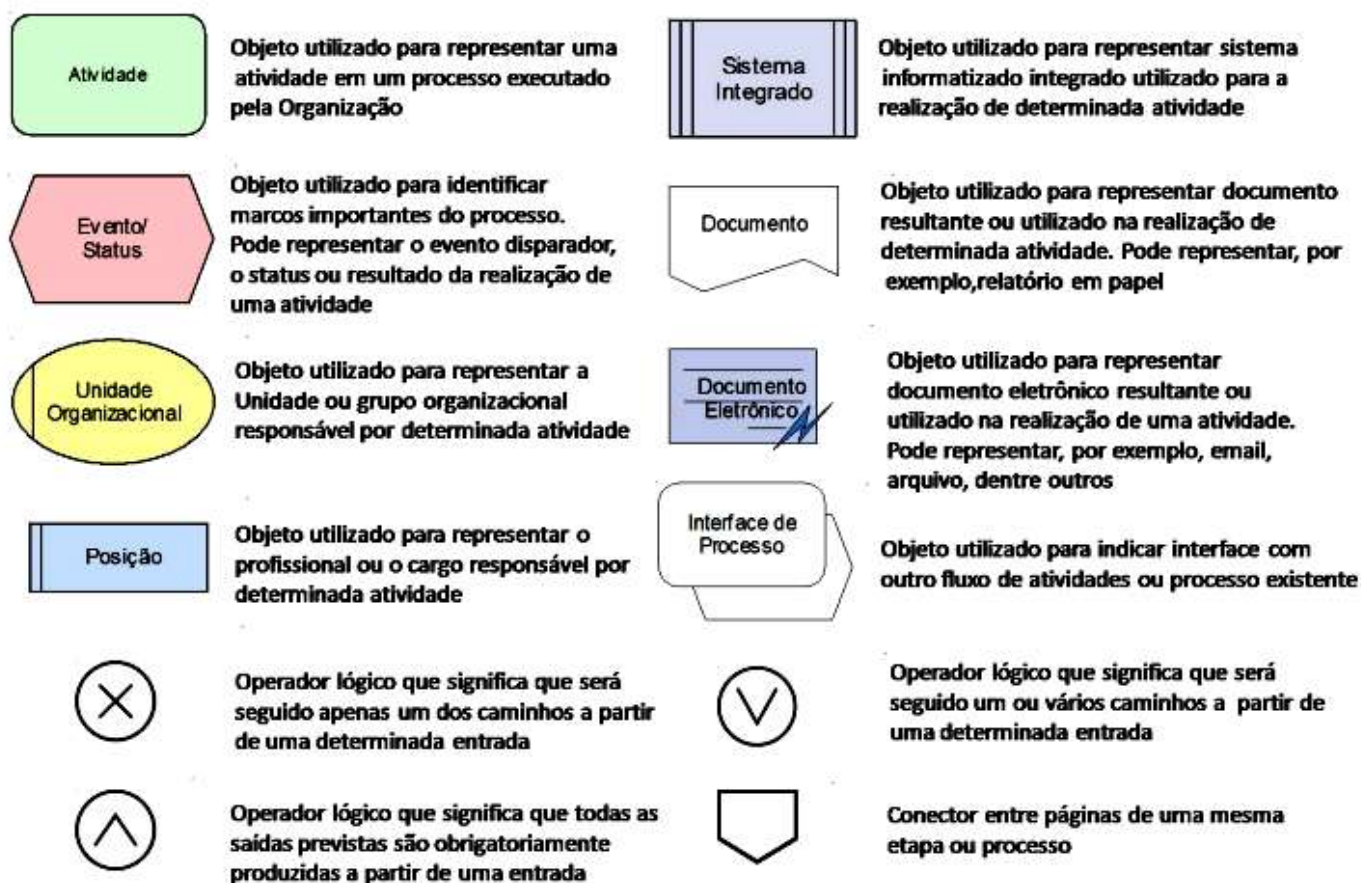
O EPC (Event-Driven Process Chain) é um modelo de representação de processos baseado no controle de fluxos de atividades e eventos. As atividades correspondem às tarefas executadas e os eventos correspondem ao início e ao fim da realização de uma atividade. A cadeia de processos orientada por eventos faz parte da arquitetura ARIS (Architecture of Integrated Information Systems).

No EPC, as atividades (representadas por retângulos com bordas arredondadas) e os eventos (representados por hexágonos) podem ser conectados utilizando os operadores lógicos, que apresentam as dependências de uma ou de mais atividades para a realização da próxima atividade, para o paralelismo na execução das atividades e para a determinação dos caminhos opcionais que um processo pode seguir após realizar uma atividade.

Na figura 1, apresentamos os principais elementos utilizados na modelagem EPC.

---

<sup>2</sup>Questão 13 – Enade 2011.



**Figura 1.** Símbolos utilizados no EPC do ARIS.

## 2. Análise das afirmativas

I – Afirmativa incorreta.

JUSTIFICATIVA. O EPC é um modelo de representação de processo baseado no fluxo das atividades e dos eventos. Dessa forma, não há ligações entre processos, mas entre atividades e eventos.

II – Afirmativa correta.

JUSTIFICATIVA. No modelo EPC, ao descrevermos um processo, devemos iniciar e terminar com um evento.

III – Afirmativa correta.

JUSTIFICATIVA. As atividades (funções) são representadas por um retângulo com bordas arredondadas.

Alternativa correta: E.

### 3. Indicações bibliográficas

- ALVES, A. *Sistema Web para apoio às análises de dados de um calorímetro de altas energias*. Rio de Janeiro: UFRJ, 2009.
- CAMEIRA, R. Hiper-integração: Engenharia de processos, arquitetura integrada de sistemas componetizados em agentes e modelos de negócios tecnologicamente habilitados. *Tese de Doutorado em Engenharia de Produção* – Rio de Janeiro: COPPE-UFRJ, 2003.
- DAVIS, R. *Business process modelling with ARIS*. London: Springer, 2001.

### Questão 3

#### Questão 3.<sup>3</sup>

Leia o texto a seguir.

*Os mapas de navegação enfocam como as pessoas se movimentam por um site ou aplicação. Cada página do site ou local da aplicação é representado com uma caixa ou cabeçalho e todas as páginas que puderem ser acessadas a partir dela devem usá-la como referência. Uma técnica bastante utilizada é colocar todos os fluxos possíveis no mapa de navegação, já que isso destacará seções onde há necessidade de uma revisão no projeto da interface.*

BENYON, D. *Interação humano-computador*. 2. ed. São Paulo: Pearson Prentice Hall, 2011.

Com relação ao tema, avalie as asserções e a relação proposta entre elas.

I. Os mapas de navegação são redesenhados muitas vezes no decorrer do ciclo de vida do projeto

PORQUE

II. A interface deve ser projetada para atender pessoas (capacidades e limitações motoras, neurológicas, cognitivas etc.), atividades que as pessoas querem realizar (questões temporais, com ou sem cooperação, se são críticas em termos de segurança etc.), contextos nos quais a interação acontece (ambiente físico, contexto social ou organizacional etc.), e ater-se às tecnologias empregadas (*hardware*, plataformas, normas, linguagens de programação, políticas de acesso em ambientes de trabalho e residencial etc.). As combinações desses elementos são muito diferentes, por exemplo, em um quiosque público, em um sistema de agenda compartilhada, em uma cabine de avião ou em um telefone celular.

Acerca dessas asserções, assinale a opção correta.

- A. As duas asserções são proposições verdadeiras, e a segunda justifica a primeira.
- B. As duas asserções são proposições verdadeiras, e a segunda não justifica a primeira.
- C. A primeira asserção é uma proposição verdadeira, e a segunda é uma proposição falsa.
- D. A primeira asserção é uma proposição falsa, e a segunda é uma proposição verdadeira.
- E. As duas asserções são proposições falsas.

---

<sup>3</sup>Questão 15 - Enade 2011.



## 1. Introdução teórica

### Mapa de navegação

O mapa de navegação é uma ferramenta gráfica que representa as interfaces gráficas dos sites ou das aplicações, os produtos e serviços, o fluxo da navegação da interface e as transações remotas e locais acessados. Ele deve ser orientado pelo fluxo do processo de negócio.

Existem vários modelos de ciclo de vida do projeto, que incluem a coleta de dados, a análise de informações, o projeto inicial, a implementação, os testes, a validação, a implantação e a manutenção.

A interface para o usuário é muito importante na área de aplicação computacional, pois ela envolve toda a parte visível ao usuário. Utilizando a interface, os usuários têm acesso às funções da aplicação. Fatores como satisfação, eficiência, segurança, custo de treinamento e de retorno de investimento dependem de um bom design de interface.

## 2. Análise das asserções

As duas asserções são proposições verdadeiras, mas a segunda não é uma justificativa correta da primeira, pois podemos redesenhar os mapas de navegação muitas vezes ao longo do ciclo de vida do projeto. Mas, para corrigirmos as interfaces gráficas dos sites ou das aplicações, os produtos e serviços, o fluxo da navegação da interface, as transações remotas e os locais acessados, o mapa de navegação deve ser orientado pelo fluxo do processo de negócio.

Alternativa correta: B.

## 3. Indicações bibliográficas

- PREECE, J.; ROGERS, Y.; SHARP, H. *Design de interação: além da interação homem-computador*. Porto Alegre: Bookman, 2005.
- ROCHA, H. V.; BARANAUSKAS, M. C. *Design e avaliação de Interfaces Humano-Computador*. Campinas: Nied-Unicamp, 2003.
- PRESSMAN, R. S. *Engenharia de software*. 6. ed. São Paulo: McGraw-Hill, 2006.

## Questão 4

### Questão 4.<sup>4</sup>

*Eclipse*, *Netbeans*, *Jdeveloper* são exemplos de ambientes integrados de desenvolvimento, chamados de IDE, que têm por objetivo facilitar o desenvolvimento de *softwares*, provendo maior produtividade e gerenciamento de projetos. A especificação *JavaBeans* foi criada para ser um padrão de desenvolvimento de componentes que possam ser facilmente usados por outros desenvolvedores em diferentes IDE.

Com relação ao tema, avalie as asserções e a relação proposta entre elas.

I. Seja para o *Netbeans* ou para o *Eclipse*, é possível adquirir componentes de terceiros que facilitem a implementação do seu projeto

PORQUE

II. Como o código desses componentes está em linguagem intermediária, ou seja, independente da arquitetura de um computador real, só é necessário que a máquina virtual esteja instalada no computador onde o aplicativo será executado e a máquina virtual será a responsável pela interpretação do código para a linguagem de máquina do computador em execução.

Acerca dessas asserções, assinale a alternativa correta.

- A. As duas asserções são proposições verdadeiras, e a segunda justifica a primeira.
- B. As duas asserções são proposições verdadeiras, e a segunda não justifica a primeira.
- C. A primeira asserção é uma proposição verdadeira, e a segunda é uma proposição falsa.
- D. A primeira asserção é uma proposição falsa, e a segunda é uma proposição verdadeira.
- E. As duas asserções são proposições falsas.

## 1. Introdução teórica

### JavaBeans

O modelo de componentes JavaBeans foi desenvolvido para reutilizar componentes em outros componentes JavaBeans e applets, podendo rodar em um web browser usando uma Java Virtual Machine (JVM).

A JVM é um programa que executa e carrega os aplicativos Java, converte o bytecodes (formato intermediário de código projetado para permitir que vários hardwares e

---

<sup>4</sup>Questão 16 - Enade 2011.

softwares executem o mesmo código) em códigos de máquina executável e gerencia os aplicativos conforme são executados.

Dessa forma, os programas em Java podem funcionar em qualquer plataforma de software ou hardware que tenha uma JVM instalada.

## **2. Análise das asserções**

As duas asserções são proposições verdadeiras, mas a segunda não é uma justificativa correta da primeira porque é possível adquirir componentes de terceiros que facilitem a implementação de um projeto, com ganho de tempo e custo mais baixo do que ser desenvolvido internamente.

Alternativa correta: B

## **3. Indicações bibliográficas**

- ARNOLD, K.; GOSLING, J.; HOMES, D. *A linguagem de programação Java*. 4. ed. Porto Alegre: Bookman, 2007.
- HORSTMANN, C. *Padrões e projeto orientados a objetos*. 2. ed. Porto Alegre: Bookman, 2007.
- MENDES, D. R. *Programação Java com ênfase em orientação a objeto*. Rio de Janeiro: Novatec, 2009.

## Questão 5

### Questão 5.<sup>5</sup>

Leia o texto a seguir.

*Os processos de negócio aumentam o entendimento de "o que precisa ser feito" e de "como precisa ser feito" dentro de uma organização.*

BALDAM, R. 2009, 2. ed.

Sabendo disso, a fase da gestão de processos de negócio que evita a estagnação dos processos organizacionais é

- A. a fase de gerenciamento.
- B. a fase de planejamento.
- C. a fase de monitoração.
- D. a fase de modelagem.
- E. a fase de otimização.

### 1. Introdução teórica

#### Processo de negócio

O processo de negócio é um conjunto de ações e de atividades que são executadas para atingir determinado objetivo. Os processos são caracterizados pelas suas entradas, pelas técnicas que podem ser aplicadas e pelos resultados obtidos.

A gestão de processos de negócio (Business Process Management ou BPM) é um conceito que une gestão de negócios e tecnologia da informação, com a aplicação de ferramentas, técnicas e conhecimentos adequados às atividades do projeto, com foco na otimização dos resultados das organizações por meio da melhoria dos processos de negócio.

As fases da gestão de processos de negócio são as descritas a seguir.

- **Fase de planejamento.** Nessa fase, definem-se os processos mais importantes para a organização e verificam-se as possíveis falhas podem causar prejuízos à empresa. Estabelece-se um plano de ação para a implantação e a priorização dos processos que necessitam de ação imediata.
- **Fase de modelagem.** Nessa fase, executam-se atividades, como a compreensão e a documentação de processos e a coleta de dados relativos ao custo, aos recursos e ao

---

<sup>5</sup>Questão 17 - Enade 2011.

tempo, com o objetivo de obterem-se informações sobre o processo atual e sobre o modelo de processo futuro.

- **Fase de otimização.** Nessa fase, aplicam-se otimizações nos processos da organização. Nesse momento, podem ser removidos gargalos e eliminados custos não necessários ao seu negócio. Trata-se da fase em que se evita a estagnação dos processos organizacionais.
- **Fase de execução de processos.** Essa fase engloba as atividades de implementação e de execução dos processos, como a implantação dos planos de transferência de tecnologia e de treinamentos.
- **Fase de controle e de análise de dados.** O ideal é que o controle seja feito durante todas as fases. É importante o uso de indicadores, gerando informações que posteriormente realimentarão as atividades de otimização e planejamento.

## 2. Análise da questão

A gestão de processos estimula a otimização e, conseqüentemente, melhorias contínuas dos processos organizacionais. Utilizando técnicas gerenciais atuais, a gestão de processos faz com que essa abordagem se torne rotineira para muitas organizações, evitando a estagnação dos processos.

Alternativa correta: E.

## 3. Indicações bibliográficas

- BALDAM, R.; VALLE, R.; PEREIRA, H.; HILST, S.; ABREU, M.; SOBRAL, V.; *Gerenciamento de processo de negócio: BPM – Business Process Management*. 2. ed. São Paulo, Érica, 2009.
- PRESSMAN, R. S. *Engenharia de software*. 6. ed. São Paulo: McGraw-Hill, 2006.

## Questão 6

### Questão 6.<sup>6</sup>

Leia o texto a seguir.

*A Norma ISO/IEC FDIS 14764 (2006) estabelece definições de vários tipos de manutenção e fornece um guia para gerenciar o processo de manutenção, que pode ser aplicado no planejamento, execução e controle, revisão e avaliação, e fechamento do processo de manutenção. Segundo essa Norma, solicitações de modificação são classificadas como corretiva, preventiva, adaptativa ou perfectiva. Os detalhes de como implementar ou realizar as atividades e tarefas de manutenção não são especificadas pela Norma, sendo de responsabilidade do mantenedor.*

ISO/IEC FDIS 14764. *Software Engineering – Software Life Cycle Processes - Maintenance*. 2006.

Considerando os tipos de manutenção e as atividades de implementação do processo, avalie as afirmativas.

- I. O mantenedor deve desenvolver, documentar e executar planos e procedimentos para realizar as atividades e tarefas do processo de manutenção.
- II. O mantenedor deve alterar a configuração do sistema para corrigir erros identificados pelos usuários usando a manutenção perfectiva.
- III. O mantenedor deve estabelecer procedimentos para receber, registrar e rastrear solicitações de modificação/registro de problemas dos usuários, e também prover realimentação para os usuários.
- IV. O mantenedor deve documentar a estratégia a ser usada para melhorar a manutabilidade futura do sistema, usando a manutenção corretiva.

É correto apenas o que se afirma em

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

### 1. Introdução teórica

#### Norma ISO/IEC 14764

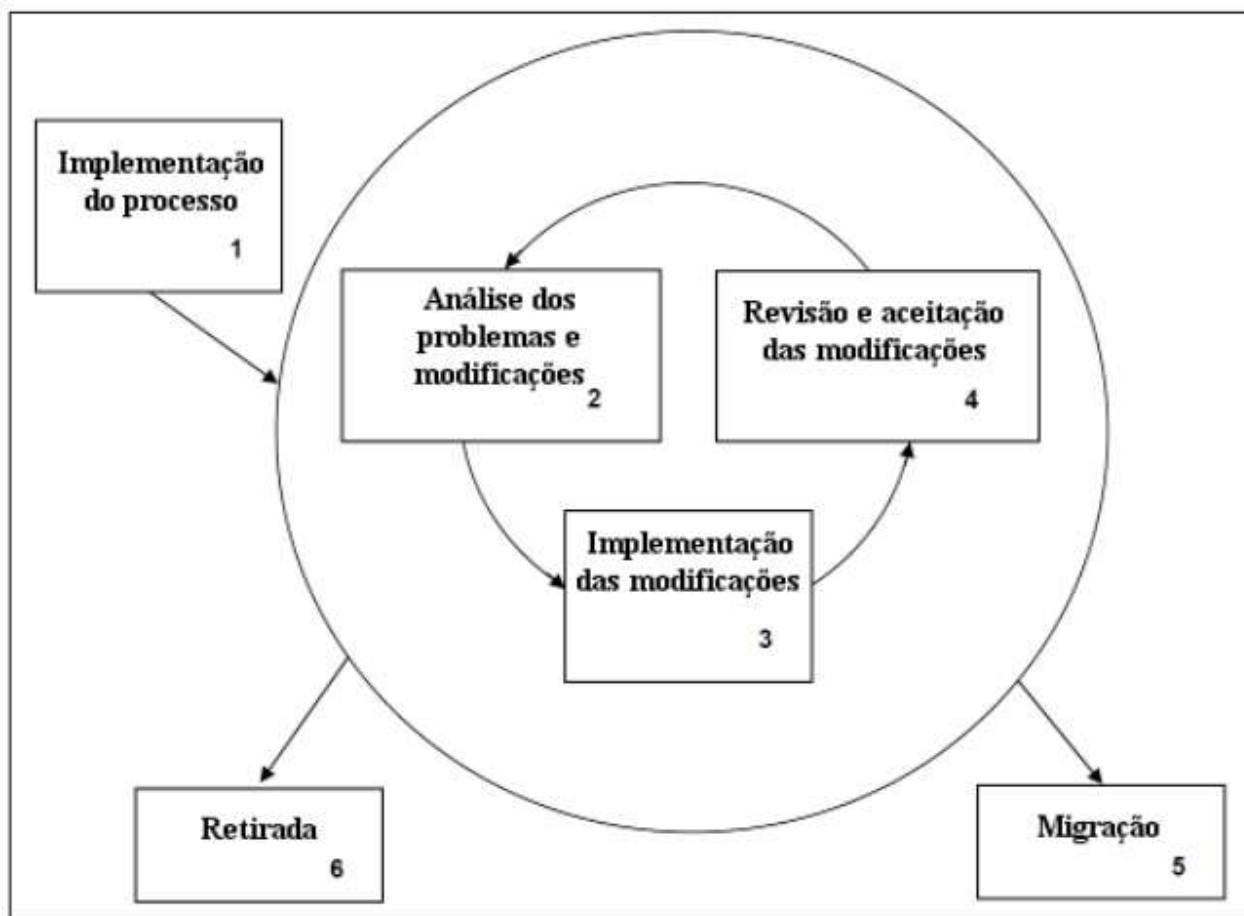
A norma ISO/IEC 14764 (ISO/IEC, 2006) descreve um padrão internacional para a manutenção de software, cujo objetivo é apoiar o gerenciamento, o planejamento e a execução das atividades de manutenção, independentemente de características como

---

<sup>6</sup>Questão 18 - Enade 2011.

tamanho, complexidade ou domínio do software. Essa norma trata desde o planejamento da fase da manutenção até a retirada do software do ambiente de produção no final de sua vida útil.

Na figura 1, ilustramos o processo de manutenção proposto pela norma, dividido em duas fases. A primeira fase concentra-se nas atividades executadas no desenvolvimento inicial, que visam a auxiliar a transferência do conhecimento entre as equipes de manutenção e de desenvolvimento. A segunda fase preocupa-se com o período após o desenvolvimento inicial e visa a garantir a evolução e a operação do software para atender às necessidades do usuário.



**Figura 1.** Processo de manutenção de software da norma ISO/IEC 14764

**Fonte.** Norma ISO/IEC 14764 (ISO/IEC, 2006).

Os objetivos das atividades do processo proposto pela norma são os descritos a seguir.

- **Implementação do processo:** aprovação dos planos e procedimentos que deverão ser executados na fase de manutenção.
- **Análise dos problemas e modificações:** análise das requisições de manutenção. As modificações devem ser propostas, documentadas e aprovadas.

- **Implementação das modificações:** depois de aprovadas, as modificações devem ser desenvolvidas e testadas pelos mantenedores.
- **Revisão e aceitação das modificações:** as modificações feitas devem ser revisadas, avaliadas e aceitas se estiverem conforme o escopo aprovado anteriormente.
- **Migração:** as etapas necessárias para atender à migração do software devem ser definidas, documentadas, aprovadas e executadas.
- **Retirada:** análise e execução da retirada do software do ambiente de produção no final de sua vida útil, para que não ocorram problemas futuros.

## 2. Análise das afirmativas

I – Afirmativa correta.

JUSTIFICATIVA. São de responsabilidade do mantenedor o desenvolvimento, a documentação e a execução dos planos e procedimentos para a realização das atividades e tarefas do processo de manutenção.

II – Afirmativa incorreta.

JUSTIFICATIVA. As manutenções perfectivas envolvem as manutenções responsáveis por inserir ou modificar os requisitos funcionais e não funcionais do software e melhorar a manutenção do sistema.

III – Afirmativa correta.

JUSTIFICATIVA. É de responsabilidade do mantenedor o estabelecimento de procedimentos para receber, registrar e rastrear solicitações de modificação/registro de problemas dos usuários. Além disso, ele deve prover a realimentação para os usuários.

IV – Afirmativa incorreta.

JUSTIFICATIVA. A manutenção corretiva contempla as mudanças acarretadas em virtude de falhas no processamento das funcionalidades ou em virtude da falta de atributos especificados.

Alternativa correta: C.



### 3. Indicações bibliográficas

- APARECIDO, G. Avaliação e formalização de um processo para agrupamento de solicitações de manutenção de software. *Tese de Mestrado em Informática*. Belo Horizonte. Belo Horizonte: PUC-MG, 2011.
- PRESSMAN, R. S. *Engenharia de software*. 6. ed. São Paulo: McGraw-Hill, 2006.
- SOMMERVILLE, I. *Engenharia de software*. 8. ed. São Paulo: Pearson, 2007.

## Questão 7

### Questão 7.<sup>7</sup>

Leia o texto a seguir.

*Padrões de criação (creational patterns) abstraem a forma como objetos são criados, tornando o sistema independente de como os objetos são criados, compostos e representados. Um padrão de criação de classe usa a herança para variar a classe que é instanciada, enquanto que um padrão de criação de objeto delegará a instanciação para outro objeto. Há dois temas recorrentes nesses padrões. Primeiro, todos encapsulam conhecimento sobre quais classes concretas são usadas pelo sistema. Segundo, ocultam o modo como essas classes são criadas e montadas. Tudo que o sistema sabe no geral sobre os objetos é que suas classes são definidas por classes abstratas. Os padrões de criação são classificados em Abstract Factory, Builder, Factory Method, Prototype e Singleton.*

GAMMA, E. et al. *Design Patterns: Elements of Reusable Object-Oriented Software*. Reading, MA: Addison-Wesley, 1994  
(com adaptações).

O padrão *Abstract Factory* é usado quando

- A. o sistema deve ser independente da maneira como seus produtos são criados, relacionados e representados.
- B. o algoritmo de criação de um objeto deve ser independente das suas partes e da maneira como ele é montado.
- C. houver uma única instância de uma classe e esta for acessada a partir de um ponto de acesso conhecido.
- D. classes delegam responsabilidade a alguma das subclasses e deseja-se localizar qual é a subclasse acessada.
- E. as classes utilizadas para instanciação são especificadas em tempo de execução ou carregadas dinamicamente.

## 1. Introdução teórica

### Design Pattern e Abstract Factory

Os Design Patterns (Padrões de Projetos) são uma coleção de padrões de projetos de software que reúnem soluções para problemas recorrentes e conhecidos, descrevendo uma solução legítima para os problemas comuns.

O padrão de criação Abstract Factory encapsula a criação de objetos. Esse padrão é muito útil para a construção de objetos individuais com propósito específico, sem a necessidade de serem conhecidas as classes específicas instanciadas.

---

<sup>7</sup>Questão 20 – Enade 2011.

Um exemplo de utilização do Abstract Factory é o caso de um sistema que precisa de suporte para múltiplos tipos de interfaces gráficas (Windows, Mac etc). Informa-se à factory que o sistema deve se parecer com o Windows. Após esse suporte, o sistema retorna à fábrica GUI (Graphical User Interface) dos objetos relativos ao Windows. Quando utilizamos um objeto (botão, check-box e janelas), a fábrica de GUI retorna às instâncias desses componentes para o Windows.

## 2. Análise da questão

O padrão Abstract Factory é utilizado quando o sistema opera independentemente da maneira como seus produtos são criados, relacionados e representados.

Alternativa correta: A.

## 3. Indicações bibliográficas

- ARNOLD, K.; GOSLING, J.; HOMES, D. *A linguagem de programação Java*. 4. ed. Porto Alegre: Bookman, 2007.
- HORSTMANN, C. *Padrões e projeto orientados a objetos*. 2. ed. Porto Alegre: Bookman, 2007.
- MENDES, D. R. *Programação Java com ênfase em orientação a objeto*. Rio de Janeiro: Novatec, 2009.

## Questão 8

### Questão 8.<sup>8</sup>

O paradigma de programação orientado a objetos tem sido largamente utilizado no desenvolvimento de sistemas.

Considerando o conceito de herança, avalie as afirmativas.

- I. Herança é uma propriedade que facilita a implementação de reuso.
- II. Quando uma subclasse é criada, ela herda todas as características da superclasse, não podendo possuir propriedades e métodos próprios.
- III. Herança múltipla é uma propriedade na qual uma superclasse possui diversas subclasses.
- IV. Extensão é uma das formas de se implementar herança.

É correto apenas o que se afirma em

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

## 1. Introdução teórica

### Paradigma de orientação a objetos

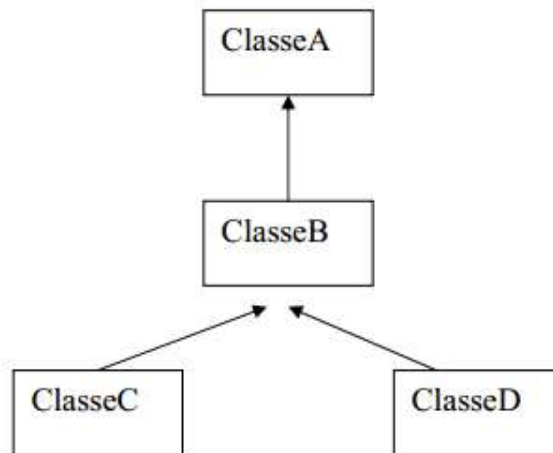
O paradigma de orientação a objetos é uma forma de analisar, projetar e programar um sistema de software com base na interação e na composição de várias unidades de software, chamadas de objetos. Um objeto apresenta atributos (características) e pode executar tarefas reagindo (por meio dos métodos) a mensagens enviadas a ele e enviar mensagens a outros objetos.

O conceito de herança afirma que uma classe pode herdar operações de uma superclasse e as suas operações podem ser herdadas de uma ou de mais subclasses. A herança permite definir uma nova classe com base em uma classe já existente, com modificações e/ou extensões de comportamento, pois isso facilita a implementação de reuso.

---

<sup>8</sup>Questão 21 – Enade 2011.

Na figura 1, podemos observar que a Classe A é superclasse da Classe B e a Classe B é subclasse da Classe A.



**Figura 1.** Herança de superclasse e subclasse.

## 2. Análise das afirmativas

I – Afirmativa correta.

JUSTIFICATIVA. Herança é uma propriedade que facilita a implementação de reuso, pois podemos utilizar uma classe inicial existente e criar várias outras classes a partir dela.

II – Afirmativa incorreta.

JUSTIFICATIVA. Quando uma subclasse é criada, ela herda todas as características da superclasse. A subclasse pode apresentar propriedades e métodos próprios.

III – Afirmativa incorreta.

JUSTIFICATIVA. Herança múltipla é a propriedade em que uma subclasse tem diversas superclasses, isto é, sempre que uma subclasse apresentar duas ou mais superclasses imediatas, ela é "filha" de mais de uma classe. Essa propriedade possibilita a combinação das características de diversas superclasses.

IV – Afirmativa correta.

JUSTIFICATIVA. A herança torna a extensão ou especialização de classe existente. Extensão é uma das formas de se implementar herança.

Alternativa correta: C.

### 3. Indicações bibliográficas

- MARQUES, P.; PEDROSO, H. *C# 2.0*. Rio de Janeiro: LTC, 2007.
- SHARP, J. *Microsoft visual C# 2008 passo a passo*. Porto Alegre: Bookman, 2008.

## ÍNDICE REMISSIVO

<b>Questão 1</b>	Programação extrema (XP).
<b>Questão 2</b>	Processo.
<b>Questão 3</b>	Mapa de navegação.
<b>Questão 4</b>	JavaBeans.
<b>Questão 5</b>	Processo de negócio.
<b>Questão 6</b>	Norma ISSO/IEC 14764.
<b>Questão 7</b>	Design Pattern e Abstract Factory.
<b>Questão 8</b>	Paradigma de orientação a objetos.