

COMPILADO CORRIGIDO DE AV2 - UNIDADE 01-04

TENTATIVA 01-UNIDADE 01

Leia o trecho a seguir:

“O modelo de casos de uso é importante, pois direciona diversas tarefas posteriores do processo de desenvolvimento de um sistema de software. Além disso, esse modelo força os desenvolvedores a moldarem o sistema de acordo com as necessidades do usuário.”Fonte: BEZERRA, E. Princípios de análise e projeto de sistemas com UML. 2 ed. Rio de Janeiro: Elsevier Editora, 2007. p. 54.

Considerando o trecho apresentado e o conceito estudado sobre casos de uso, analise as afirmativas a seguir.

- I. A modelagem de casos de usos foi incorporada como técnica no processo de desenvolvimento de software, denominado *objectory*, no início da década de 1990.
- II. O modelo de casos de uso é uma representação das funcionalidades internamente observáveis do sistema e da mecânica de funcionamento de cada componente do sistema.
- III. Os casos de uso identificam elementos externos ao sistema que interage com ele.
- IV. Um caso de uso pode ser iniciado por um elemento humano, modelado com notação própria, chamado de criador, do termo em inglês *creator*.

Está correto apenas o que se afirma em:

RESPOSTA CORRETA: I e III.

=====

Leia o trecho a seguir:

“Ao fazer a modelagem da arquitetura do sistema, você captura decisões sobre os requisitos do sistema, seus elementos lógicos e seus elementos físicos. Você também fará a modelagem tanto de aspectos estruturais como comportamentais dos sistemas e dos padrões que dão forma a essas visões. Por fim, você desejará focalizar as costuras existentes entre os subsistemas e fazer o acompanhamento desde os requisitos até a entrega.”Fonte: BOOCH, G.; RUMBAUGH, J.; JACOBSON, I. UML, Guia do Usuário. 2. ed. Rio de Janeiro: Editora Campus, 2005. p. 629.

Considerando essas informações e o conteúdo estudado sobre as cinco visões complementares utilizadas na modelagem, analise-as a seguir e associe-as com as suas respectivas funções.

- 1) Visão de casos de uso.
- 2) Visão de projeto.
- 3) Visão de processo.
- 4) Visão de implementação.
- 5) Visão de implantação.

RESPOSTA

CORRETA:

5, 3, 2, 1, 4



=====

Leia o trecho a seguir:

“Um sistema de software orientado a objetos consiste em objetos em colaboração com o objetivo de realizar as funcionalidades desse sistema. Cada objeto é responsável por tarefas específicas. É graças à cooperação entre objetos que a computação do sistema se

desenvolve.”Fonte: BEZERRA, E. Princípios de análise e projeto de sistemas com UML. 2. ed. Rio de Janeiro: Elsevier Editora, 2007. p. 7.

Considerando essas informações e o conteúdo estudado sobre análise e projeto orientado a objeto, analise as afirmativas a seguir.

I. O objetivo básico da Análise Orientada a Objetos é identificar classes a partir das quais objetos serão representados como instâncias e uma de suas tarefas é a identificação de atores.

II. Para cada classe devem ser anotadas, em modelos, as suas dependências, indicando o nível que cada uma estará na arquitetura, na escala bottom, middle e top.

III. No momento da definição de requisitos no texto obtido, verbos são potenciais candidatos a classes, e nomes (substantivos) são potenciais candidatos a métodos.

IV. Na especificação de atributos, será feita uma lista de todos os detalhes da classe, que serão as variáveis dos objetos.

Está correto apenas o que se afirma em:

RESPOSTA CORRETA: I e IV.

Integradores são os profissionais da equipe de configuração responsáveis por realizar a integração dos itens modificados no sistema. Realizar a integração dos itens do sistema, nesse caso, seria o processo de retirar uma determinada função do sistema, realizar a alteração necessária e, então, devolver esse item de volta ao sistema.

Fonte: BERSOFF, E.H. Elements of Software Configuration Management. IEEE Transactions on Software Engineering, v. 10, n. 1, 1984.

Considerando essas informações e o conteúdo estudado sobre o processo de integração de software, pode-se afirmar que esse procedimento é conhecido como:

RESPOSTA CORRETA: a entrada e a saída de qualquer item relacionado ao produto de software para fins de controle de configuração e mudanças é conhecido como “checkin e checkout”.

Leia o trecho a seguir:

“[...] a modelagem de caso de uso é amplamente usada para apoiar a elicitação de requisitos. Um caso de uso pode ser tomado como um cenário simples que descreve o que o usuário espera de um sistema. Cada caso de uso representa uma tarefa discreta [...]”.

Fonte: Sommerville, I. Engenharia de Software. 9 ed. São Paulo: Pearson Prentice Hall, 2011. Pág. 86.

Considerando essas informações e o conceito estudado sobre casos de uso, analise as afirmativas a seguir.

I. Um caso de uso deve compreender um fluxo de eventos completo e consistente, com colaborações entre elementos.

II. O caso de uso apresenta relações de quatro componentes essenciais: elementos internos, elementos externos, um relógio para disparar eventos e uma trilha de conexão, para passagem de dados.

III. Devem ser feitos e modelados casos de uso irreais de estouro de limites previstos dadas memórias limitantes, chamados de casos de saturação, para que os projetistas possam programar situações futuras.

IV. O conjunto de casos de uso representa as situações possíveis de utilização do software modelado.

Está correto apenas o que se afirma em:

RESPOSTA CORRETA: I e IV

Leia o trecho a seguir:

“Para entregar um software que satisfaça ao propósito pretendido, será preciso reunir-se e interagir com os usuários de uma maneira disciplinada, com a finalidade de expor os requisitos reais do sistema. Para desenvolver software de qualidade duradoura, será necessário criar uma arquitetura de fundação sólida que aceite modificações.”

Fonte: BOOCH, G., RUMBAUGH, J., JACOBSON, I. UML, Guia do Usuário. 2 ed. Rio de Janeiro: Editora Campus, 2005. p. 31.

Considerando essas informações e o conteúdo estudado sobre tipos de requisitos, analise os requisitos a seguir e associe-os com suas respectivas características.

- 1) Requisitos de segurança.
- 2) Requisitos de negócio.
- 3) Requisitos de confiabilidade.
- 4) Requisitos de aspectos legais.
- 5) Requisitos de usabilidade.

RESPOSTA CORRETA: 2, 1, 5, 3, 4.

Entende-se como artefato de software os diversos tipos de subprodutos concretos que são produzidos durante o desenvolvimento de software. Em outras palavras, é tudo aquilo que é produzido pelos profissionais da Engenharia de Software ao longo do desenvolvimento dos sistemas.

Considerando essas informações e o conteúdo estudado sobre artefatos de software, analise as asserções a seguir e a relação proposta entre elas

I. A lista de requisitos do software, construída na etapa de levantamento de requisitos, reúne a descrição de todas as funcionalidades e as ações que o sistema deve executar, sendo assim um exemplo de artefato do sistema.

Porque:

II. Ao ser construída pelos analistas de requisitos, tem também a finalidade de definir que mudanças podem ocorrer no sistema na fase de implementação.

A seguir, assinale a alternativa correta:

RESPOSTA

CORRETA:



A asserção I é uma proposição verdadeira, e a II é uma proposição falsa.

O processo de desenvolvimento de software pode ser dividido em até nove atividades distintas, estruturadas e organizadas de maneira a obter o máximo de produtividade dos profissionais, e o sucesso na execução das atividades, buscando entregar software de qualidade e de valor para o cliente. De certa forma, essas atividades são divididas de acordo com o tempo total do projeto e organizadas de forma que possam acontecer em paralelo. Entretanto, algumas dessas atividades acontecem em maior intensidade no início do projeto, enquanto algumas outras tem foco maior no final.

Fonte: BOURQUE, P.; FAIRLEY, R. E. Guide to the software engineering body of knowledge. Los Alamitos. IEEE Computer Society Press, 3ª Ed., 2014.

Considerando essas informações e o conteúdo estudado sobre processos de software, pode-se afirmar que:

RESPOSTA CORRETA: o levantamento de requisitos é uma atividade realizada no início do projeto e visa construir uma lista de funcionalidades que deve ser desenvolvida no sistema

=====

Leia o trecho a seguir:

“Muitos profissionais consideram um tanto complexo o conceito do paradigma de orientação a objetos. No entanto, esse conceito é apenas diferente do enfoque procedural ao qual estão acostumados. Na realidade, o ser humano, no início de sua infância, aprende e pensa de uma maneira orientada a objetos, representando seu conhecimento por meio de abstrações e classificações (na verdade, continuamos fazendo isso mesmo quando adultos, mas desenvolvemos outras técnicas que também utilizamos em paralelo)”. Fonte: GUEDES, G. T. A. UML 2: uma abordagem prática. 2 ed. São Paulo: Novatec Editora, 2011. p. 43.

Considerando essas informações e o conteúdo estudado sobre orientação a objetos, analise os princípios a seguir e associe-os com suas respectivas funções.

RESPOSTA

CORRETA:



2, 1, 4, 5, 3.

=====

A gerência de configuração pode ser entendida como uma das atividades do desenvolvimento de software cujo principal papel é o de planejar e executar as ações necessárias para garantir o correto rastreamento dos artefatos de software que são produzidos ao longo do desenvolvimento. Sendo assim, sua principal motivação é evitar e resolver os problemas que ocorrem em projetos de software devido a alterações realizadas inadequadamente.

Fonte: SANCHES, R. Gerencia de configuração. In: Qualidade de Software [S.l: s.n.], 2001.

Considerando essas informações e o conteúdo estudado sobre os objetivos da gerência de configuração, analise as asserções a seguir e a relação proposta entre elas.

I. A gerência de configuração é vista como uma atividade essencial para promover o alto grau de controle de mudanças em sistemas de software, e assim, acompanhar o processo de evolução de um produto.

Porque:

II. A gerência de configuração consegue promover a rastreabilidade dos documentos, modelos e demais itens produzidos ao longo do desenvolvimento do software pelos profissionais envolvidos no processo.

A seguir, assinale a alternativa correta:

RESPOSTA CORRETA: As asserções I e II são proposições verdadeiras, mas a II não é uma justificativa correta da I.

TENTATIVA 02-UNIDADE 01

Durante o desenvolvimento de software mudanças são praticamente inevitáveis. Neste contexto, a gerência de configuração e suas ações durante o desenvolvimento possuem grande influência no sucesso de um software e no processo de desenvolvimento de uma maneira geral. Por este motivo, a gerência de configuração é sempre considerada uma boa prática de desenvolvimento ligada à qualidade do sistema

Fonte: BOURQUE, P.; FAIRLEY, R. E. Guide to the software engineering body of knowledge. 3. ed. Los Alamitos: IEEE Computer Society Press, 2014.

Considerando essas informações e o conteúdo estudado sobre a gerência de configuração, pode-se afirmar que:

RESPOSTA

CORRETA:



a gerência de configuração permite que qualquer alteração no sistema seja revista antes da entrega.

Leia o trecho a seguir:

“O modelo de casos de uso é importante, pois direciona diversas tarefas posteriores do processo de desenvolvimento de um sistema de software. Além disso, esse modelo força os desenvolvedores a moldarem o sistema de acordo com as necessidades do usuário.” Fonte: BEZERRA, E. Princípios de análise e projeto de sistemas com UML. 2 ed. Rio de Janeiro: Elsevier Editora, 2007. p. 54.

Considerando o trecho apresentado e o conceito estudado sobre casos de uso, analise as afirmativas a seguir.

- I. A modelagem de casos de usos foi incorporada como técnica no processo de desenvolvimento de software, denominado *objectory*, no início da década de 1990.
- II. O modelo de casos de uso é uma representação das funcionalidades internamente observáveis do sistema e da mecânica de funcionamento de cada componente do sistema.
- III. Os casos de uso identificam elementos externos ao sistema que interage com ele.
- IV. Um caso de uso pode ser iniciado por um elemento humano, modelado com notação própria, chamado de criador, do termo em inglês *creator*.

Está correto apenas o que se afirma em:

RESPOSTA

CORRETA:



I e III.

Especialidades profissionais são extremamente importantes no desenvolvimento do software. Na gerência de configuração, isso não é diferente, uma vez que cada papel profissional executa atividades específicas. Por exemplo, na gerência de configuração em uma equipe completa, em que diferentes profissionais estão trabalhando, existe o papel do gerente de controle de mudança.

Considerando essas informações e conteúdo estudado sobre o gerente de controle de mudanças, analise as afirmativas a seguir.

- I. O gerente de controle de mudança é o profissional responsável por supervisionar o processo de mudanças em uma determinada parte do sistema.

II. O gerente de controle de mudanças é o profissional responsável por entender quais serão os impactos caso uma mudança seja autorizada, considerando tempo e custo.

III. O gerente de controle de mudanças é o profissional responsável por monitorar como as mudanças realizadas são finalizadas e reportadas.

IV. O gerente de controle de mudanças é o profissional responsável por definir como o código do sistema deve ser reestruturado após uma mudança.

Está correto apenas o que se afirma em:

Ocultar opções de resposta

RESPOSTA CORRETA: I, II e III.

Leia o trecho a seguir:

“Para entregar um software que satisfaça ao propósito pretendido, será preciso reunir-se e interagir com os usuários de uma maneira disciplinada, com a finalidade de expor os requisitos reais do sistema. Para desenvolver software de qualidade duradoura, será necessário criar uma arquitetura de fundação sólida que aceite modificações.”

Fonte: BOOCH, G., RUMBAUGH, J., JACOBSON, I. UML, Guia do Usuário. 2 ed. Rio de Janeiro: Editora Campus, 2005. p. 31.

Considerando essas informações e o conteúdo estudado sobre tipos de requisitos, analise os requisitos a seguir e associe-os com suas respectivas características.

1) Requisitos de segurança.

2) Requisitos de negócio.

3) Requisitos de confiabilidade.

4) Requisitos de aspectos legais.

5) Requisitos de usabilidade.

() São específicos de cada software como, por exemplo, um cliente efetuando uma compra on-line em um site de e-commerce. São do tipo funcionais.

() Dizem respeito à proteção dos dados e das funções do software, sendo do tipo não funcionais.

() Referem-se à facilidade de utilização do software a partir de interfaces amigáveis, sendo do tipo não funcionais.

() Estão relacionados à probabilidade de operação livre de falhas de um software, em um ambiente e tempo especificados, sendo do tipo não funcionais.

() Definem restrições e condições de legislação do ambiente relacionadas ao campo de atuação do software, sendo do tipo normativos.

Agora, assinale a alternativa que apresenta a sequência correta:

RESPOSTA

CORRETA:



2, 1, 5, 3, 4.

Entende-se como artefato de software os diversos tipos de subprodutos concretos que são produzidos durante o desenvolvimento de software, como por exemplo, todo o material que é produzido durante o trabalho pelos analistas, engenheiros, gerentes e demais profissionais de Engenharia de Software. Esses artefatos serão analisados pela gerência de configuração e parte deles ou todos eles poderão ser incluídos como itens de configuração, para que sejam monitorados.

Fonte: BOURQUE, P.; FAIRLEY, R. E. Guide to the software engineering body of knowledge. 3. ed. Los Alamitos: IEEE Computer Society Press, 2014.

Considerando essas informações e o conteúdo estudado sobre as categorias que melhor representam os artefatos de software que podem ser incluídos na lista de itens de configuração, analise as alternativas a seguir.

I. Artefatos do programa, produzidos pela implementação, como código do banco de dados, podem ser um item de configuração.

II. Diagramas UML, produzidos pelos arquitetos e analistas de software, podem ser itens de configuração.

III. Documentos de teste, contendo resultados de execução, podem ser itens de configuração.

IV. Relatório de ambiente, com especificações sobre a localização dos profissionais durante o desenvolvimento do sistema, podem ser itens de configuração.

Está correto apenas o que se afirma em:

RESPOSTA

CORRETA:



I, II e III.

=====

A gerência de configuração pode ser entendida como uma das atividades do desenvolvimento de software cujo principal papel é o de planejar e executar as ações necessárias para garantir o correto rastreamento dos artefatos de software que são produzidos ao longo do desenvolvimento. Sendo assim, sua principal motivação é evitar e resolver os problemas que ocorrem em projetos de software devido a alterações realizadas inadequadamente.

Fonte: SANCHES, R. Gerencia de configuração. In: Qualidade de Software [S.l: s.n.], 2001.

Considerando essas informações e o conteúdo estudado sobre os objetivos da gerência de configuração, analise as asserções a seguir e a relação proposta entre elas.

I. A gerência de configuração é vista como uma atividade essencial para promover o alto grau de controle de mudanças em sistemas de software, e assim, acompanhar o processo de evolução de um produto.

Porque:

II. A gerência de configuração consegue promover a rastreabilidade dos documentos, modelos e demais itens produzidos ao longo do desenvolvimento do software pelos profissionais envolvidos no processo.

A seguir, assinale a alternativa correta:



As asserções I e II são proposições verdadeiras, mas a II não é uma justificativa correta da I.

=====

Leia o trecho a seguir:

“Um sistema de software orientado a objetos consiste em objetos em colaboração com o objetivo de realizar as funcionalidades desse sistema. Cada objeto é responsável por tarefas específicas. É graças à cooperação entre objetos que a computação do sistema se

desenvolve.”Fonte: BEZERRA, E. Princípios de análise e projeto de sistemas com UML. 2. ed. Rio de Janeiro: Elsevier Editora, 2007. p. 7.

Considerando essas informações e o conteúdo estudado sobre análise e projeto orientado a objeto, analise as afirmativas a seguir.

I. O objetivo básico da Análise Orientada a Objetos é identificar classes a partir das quais objetos serão representados como instâncias e uma de suas tarefas é a identificação de atores.

II. Para cada classe devem ser anotadas, em modelos, as suas dependências, indicando o nível que cada uma estará na arquitetura, na escala bottom, middle e top.

III. No momento da definição de requisitos no texto obtido, verbos são potenciais candidatos a classes, e nomes (substantivos) são potenciais candidatos a métodos.

IV. Na especificação de atributos, será feita uma lista de todos os detalhes da classe, que serão as variáveis dos objetos.

Está correto apenas o que se afirma em:

RESPOSTA CORRETA I e IV.

Todas as atividades do processo de desenvolvimento de software são executadas por um time de profissionais. Nesse esquema, cada membro do time usa sua especialização para agregar valor ao que está sendo construído, uma vez que, mesmo dividindo um objetivo em comum, a natureza dos trabalhos no desenvolvimento de software é diferente e, por isso, requer diferentes habilidades.

Considerando essas informações e o conteúdo estudado sobre os papéis profissionais na gerência de configuração, pode-se afirmar que:

RESPOSTA CORRETA: a equipe de configuração pode apresentar tamanho variável, dependendo do tamanho do software que está sendo desenvolvido.

A gerência de configuração pode ser entendida como o controle da evolução dos artefatos durante todo o desenvolvimento do projeto. Por exemplo, as várias versões do código do sistema, ou as várias versões dos diagramas do software. Neste esquema, todo o conjunto de itens que são armazenados, rastreados e controlados pela atividade de gerência de configuração são chamados, coletivamente, de configuração do software.

Considerando essas informações e o conteúdo estudado sobre itens de configuração, pode-se afirmar que:

RESPOSTA CORRETA: configuração de software é o estado atual de todos os itens que formam o software.

Leia o trecho a seguir:

“A maioria dos softwares profissionais é desenvolvida por equipes de projeto que variam em tamanho, desde duas até várias centenas de pessoas. Como é impossível para todos os membros de um grupo grande trabalharem juntos em um único problema, as grandes equipes são geralmente divididas em vários grupos. “Fonte: SOMMERVILLE, I. Engenharia de Software. 9 ed. São Paulo: Pearson Prentice Hall, 2011. p. 423 e 424.

Considerando essas informações e o conteúdo estudado sobre participantes do processo de análise e desenvolvimento de software, analise as funções a seguir e associe-as a seus respectivos detalhes de atuação.

- 1) Gerente de projeto.
- 2) Analista.
- 3) Projetista.
- 4) Programador.
- 5) Avaliador de qualidade.

RESPOSTA

CORRETA:



4, 5, 2, 1, 3.

TENTATIVA 01-UNIDADE 02

Leia o excerto a seguir:

“A gerência de configuração de software é a atividade que trata de mudanças que acontecem nos artefatos do projeto que está sendo desenvolvido. Nesta atividade trabalham diversos profissionais, especialistas em atividades de seleção de itens, versionamento de sistemas, controle de mudanças e diversas ações em um ambiente de trabalho que requer diversas habilidades técnicas e profissionais específicas da gerência de configuração.

”Fonte: SANCHES, R. Gerencia de configuração. In: Qualidade de Software [S.l: s.n.], 2001.

Considerando essas informações e o conteúdo estudado sobre versionamento de itens de configuração e o trabalho na gerência de configuração, analise as afirmativas a seguir.

I. “Baseline” é o termo usado pelos profissionais da gerência de configuração para se referir a um software recém-liberado para uso.

II. “Tag” é o termo atribuído ao profissional responsável por realizar o versionamento dos itens de configuração.

III. “Build” é como é denominada uma versão incompleta, porém estável, do sistema.

IV. “Merge” é o processo de unificar duas versões distintas do mesmo sistema ou item de configuração.

Está correto apenas o que se afirma em:



III e IV.

Leia o trecho a seguir:

“A Visual Paradigm é uma ferramenta multiplataforma poderosa e fácil de usar para gerenciamento e design de sistemas de TI. O Visual Paradigm fornece aos desenvolvedores de software uma plataforma de desenvolvimento de ponta para criar aplicativos de qualidade de modo mais barato, rápido e melhor!”Fonte: VISUAL PARADIGM. Manual do usuário da Visual Paradigm. Disponível em <<https://www.visual-paradigm.com/support/documents/vpuserguide>>. Acesso em: 05 nov. 2019.

Considerando essas informações e o conteúdo estudado, pode-se afirmar que a Visual Paradigm:

é uma ferramenta CASE do tipo vertical com módulo Web e com versões pagas e uma versão chamada Community, gratuita e limitada em recursos para uso não comercial.

Leia o trecho a seguir:

“A UML disponibiliza uma forma padrão de modelagem de projetos de sistemas, incluindo seus aspectos conceituais, tais como processos de negócios e funções do sistema, além de itens concretos como as classes escritas em determinada linguagem de programação, processos de banco de dados e componentes de software reutilizáveis.”

Fonte: PICHILIANI, M. C. Mapeamento de Software para permitir a colaboração síncrona. Dissertação de mestrado. 2006. (Mestrado em Engenharia Eletrônica e Computação) – Instituto Tecnológico de Aeronáutica, São José dos Campos, 2006. Disponível em: <<http://www.comp.ita.br/~pichilia/argo/TeseVersaoFinal.pdf>>. Acesso em: 08 nov. de 2019. p. 31.

Considerando essas informações e o conteúdo estudado sobre UML, pode-se afirmar que ela é:

uma linguagem com sintaxe bem definida para especificar, visualizar e documentar modelos de software no paradigma orientado a objetos, utilizando uma notação padrão.

leia o trecho a seguir:

“Pode-se fazer uma analogia da UML como uma caixa de ferramentas. Um pedreiro usa sua caixa de ferramentas para realizar suas tarefas. Da mesma forma, a UML pode ser

vista como uma caixa de ferramentas utilizada pelos desenvolvedores de sistemas para realizar a construção de modelos.”

Fonte: BEZERRA, E. Princípios de análise e projeto de sistemas com UML. 2. ed. Rio de Janeiro: Elsevier Editora, 2007. p. 16.

Considerando o conceito estudado sobre utilizações da UML, analise as afirmativas a seguir.

I. A UML é um auxílio ao projeto de software, permitindo criar e visualizar as estruturas de algoritmos em modelos que contribuem para a boa comunicação entre os membros da equipe.

II. A aceitação da UML é comprovadamente forte na área acadêmica por ter base teórica formal, o que a deixa complexa, enfraquecendo sua aceitação na indústria, que tem priorizado linguagens simples de modelagem.

III. A parte gráfica da UML propicia um ágil e facilitado entendimento por humanos, porém ocasiona complexidades de interpretações computacionais, trazendo ambiguidades que devem ser tratadas com expressões textuais.

IV. A UML se associa a processos, sendo utilizada como instrumental robusto em atividades para a escrita e visualização de modelos de software.

Está correto apenas o que se afirma em:

I e IV.

Leia o trecho a seguir:

“A UML é o trabalho de várias pessoas, e as ideias que ali se encontram vêm de muitos trabalhos anteriores. Seria um trabalho importante de pesquisa histórica reconstruir uma lista completa das fontes e ainda mais difícil identificar os muitos precursores que influenciaram a UML, de maneira mais ou menos relevante. Como em qualquer pesquisa científica e prática de engenharia, a UML é uma pequena colina sobre uma grande montanha de experiência anterior.” Fonte: BOOCH, G., RUMBAUGH, J., JACOBSON, I. UML, Guia do Usuário. 2. ed. Rio de Janeiro: Editora Campus, 2005. Versão digital. p. 15. Considerando essas informações e o conteúdo estudado, analise as afirmações a seguir sobre a evolução da UML e assinale V para a(s) verdadeiras e F para a(s) falsa(s).

I. () Para isentar a UML de estratégias empresariais, há, no OMG, a preferência em aceitar sugestões de melhorias provenientes de pesquisadores de universidades do que de pesquisadores ligados a empresas.

II. () Dada a reconhecida qualidade e boa utilização no mercado, a linguagem UML foi adotada, em 1997, pelo OMG como uma linguagem padrão de modelagem de software feito no paradigma orientado a objetos.

III. () O padrão de data para lançamento de novas versões da UML é em todo mês de setembro de cada ano, com cerimônias feitas em diferentes países. Essa melhoria feita de modo anual permite atualização constante da linguagem.

IV. () Antes de existir, a UML era intensa a problemática de escolha de padrão de notação dentre várias técnicas existentes, que podiam possuíam notações distintas para modelar as mesmas perspectivas de um software.

Agora, assinale a alternativa que apresenta a sequência correta:



F, V, F, V.

A crise do software na década de 1970 refletia os problemas e as dificuldades que as empresas de software enfrentavam para produzirem seus produtos em um ambiente em que havia a ausência de técnicas, padrões e modelos bem definidos que pudessem ser seguidos de maneira efetiva. Sendo assim, a Engenharia de Software surgiu buscando sanar tais problemas.

Fonte: BOURQUE, P.; FAIRLEY, R. Guide to the software engineering body of knowledge. Los Alamitos: IEEE Computer Society Press, 2014.

Considerando essas informações e o conteúdo estudado sobre artefatos de software, pode-se afirmar que:

após a crise do software, o processo de desenvolvimento de sistemas foi dividido em atividades específicas, que produziam resultados bem definidos.

A computação está cada vez mais próxima de ser uma atividade centenária que vem sendo utilizada com sucesso em diversos seguimentos da sociedade para atingir vários objetivos específicos. Atualmente, as empresas de desenvolvimento de software têm à disposição um conjunto de ferramentas, técnicas e modelos que permitem que produtos de software sejam construídos cada vez com mais qualidade.

Essa realidade é bem diferente de quando a computação dava os seus primeiros passos e o desenvolvimento de software ainda era novidade para a indústria.

Fonte: BOURQUE, P.; FAIRLEY, R. Guide to the software engineering body of knowledge. Los Alamitos: IEEE Computer Society Press, 2014.

Considerando essas informações e o conteúdo estudado sobre engenharia de software, analise as asserções a seguir e a relação proposta entre elas.

I. A crise do software se refletia nas adversidades e nos problemas enfrentados por empresas de desenvolvimento de software durante a década de 1970.

Porque:

II. Por se tratar de uma atividade nova, na época o desenvolvimento de software não apresentava padronização ou regras que determinassem como um sistema deveria ser efetivamente construído e entregue para os usuários.

A seguir, assinale a alternativa correta:



As asserções I e II são proporções verdadeiras, e a II é uma justificativa correta da I

Leia o trecho a seguir:

“A especificação da linguagem UML 2 é definida por meio da utilização de uma abordagem de metamodelagem que adapta técnicas de especificação formal. Embora essa abordagem necessite de um pouco do rigor de um método de especificação formal, ela oferece as vantagens de ser mais intuitiva e pragmática.”Fonte: GUEDES, G. T. A. UML 2: uma abordagem prática. 2. ed. São Paulo: Novatec Editora, 2011. p. 466.

Considerando essas informações e o conteúdo estudado sobre especificação da UML a partir de sua versão 2.0, analise os itens de seus componentes a seguir e associe-os a suas respectivas funções:

- 1) Superestrutura.
- 2) Infraestrutura.
- 3) OCL (Object Constraint Language).
- 4) Relacionamentos.
- 5) UML Diagram Interchange.



2, 1, 4, 5, 3.

Leia o trecho a seguir:

“Um esquema é a especificação de um modelo usando uma determinada linguagem, a qual pode ser formal ou informal (por exemplo, linguagem natural), textual ou gráfica. Quando a representação do esquema é gráfica, designa-se usualmente por diagrama.”Fonte: DA SILVA, A. M. R. e VIDEIRA, C. A. E. UML, Metodologias e Ferramentas CASE. Lisboa: Editora Centro Atlântico. 2001. p. 34.

Considerando essas informações e o conteúdo estudado, analise as afirmações a seguir sobre a taxonomia de diagramas da UML e assinale V para a(s) verdadeira(s) e F para a(s) falsa(s).

I. () O Diagrama de Perfil, ou Profile Diagram, está dentro da subcategoria de diagramas de interação, permitindo a definição de novos elementos UML e suas relações dinâmicas.

II. () A subcategoria de diagramas chamada de Interação possui 4 tipos de diagramas que descrevem como grupo de elementos de modelos colaboram em um determinado comportamento.

III. () A categoria de diagramas de Estrutura possui 7 diagramas que apresentam detalhes do sistema de modo estático, incluindo o Diagrama de Classes, representando as classes de objetos e suas ligações.

IV. () O Diagrama de Casos de Uso é um dos mais conhecidos da UML, sendo da categoria de Comportamento e estando na subcategoria de Interação, por modelar os contatos interativos de atores com itens do sistema de modo externo.

Agora, assinale a alternativa que apresenta a sequência correta:



F, V, V, F.

Leia o trecho a seguir:

“A introdução de ferramentas CASE numa organização pressupõe uma predisposição para a aplicação de regras e princípios a todo o processo de desenvolvimento, sendo esta pré-condição já de si um aspecto positivo no processo de melhoria do desenvolvimento de software numa organização.” Fonte: DA SILVA, A. M. R. e VIDEIRA, C. A. E. UML, Metodologias e Ferramentas CASE. Lisboa: Editora Centro Atlântico. 2001. p. 409.

Considerando o trecho apresentado e o conceito estudado sobre ferramentas CASE, analise as afirmativas a seguir.

I. A escolha correta da ferramenta CASE em uma organização é essencial para o sucesso de um projeto de desenvolvimento de software.

II. Existe um fato que dificulta a concorrência comercial no universo das ferramentas CASE: as opções gratuitas são completas, abrangendo toda a UML, e tem bom suporte, dominando o mercado.

III. Na especificação da UML, tem-se a recomendação de 10 ferramentas CASE que são homologadas pelo OMG para uma boa modelagem, facilitando a escolha.

IV. O risco de um baixo desempenho de ferramenta CASE deve ser analisado nos projetos de desenvolvimento de software por interferir negativamente no trabalho da equipe.

Está correto apenas o que se afirma em:

Mostrar opções de resposta



I e IV.

TENTATIVA 01-UNIDADE 02

1.

Leia o trecho a seguir:

“O UML providencia um número elevado de conceitos e notações particularmente concebidos de forma a satisfazer os requisitos típicos de modelagem de software. Contudo, podem surgir situações em que se torna desejável a introdução de conceitos e/ou de notações adicionais para além dos definidos originalmente no momento da definição do padrão.”

Fonte: Da Silva, A. M. R. e Videira, C. A. E. UML, Metodologias e Ferramentas CASE. Lisboa: Editora Centro Atlântico. 2001. p. 261. Tradução nossa.

Considerando essas informações e o conteúdo estudado sobre mecanismos de uso geral e de extensão da UML, analise os itens da UML a seguir e associe-os a suas respectivas descrições:

- 1) Estereótipos.
- 2) Notas.
- 3) Pacotes.
- 4) Tagged Values
- 5) Restrições.

() Adorno contendo comentários textuais anexados a um elemento ou a uma coleção de elementos. Permite anexar informações a um modelo, como requisitos, revisões e explicações.

() Recurso de separação que organiza elementos de modelagem em conjuntos maiores que possam ser manipulados como grupos. Realiza, então, o agrupamento de itens semanticamente relacionados.

() Conjunto de valores pré-definidos para um elemento, formando um par de valores que pode ser usado para adicionar propriedades a elementos de um modelo.

() Especificação de regras que delimitam conjunto de valores ou situações possíveis para um determinado elemento. É um recurso, então, utilizado para definir condições que devem ser mantidas como verdadeiras para que o modelo seja bem formado.

() Ampliação do vocabulário da UML, permitindo a criação de novos tipos de blocos de construção que são derivados dos já existentes, mas específicos a determinados problemas. Eles personalizam itens por meio de construções específicas para um domínio, plataforma ou método de desenvolvimento.

Agora, assinale a alternativa que apresenta a sequência correta:

1. ☐

2, 3, 4, 5, 1.

2.

Pergunta 2

0 ponto

A evolução da computação significou também uma evolução na forma como os sistemas são construídos e entregues para os clientes. Essa evolução pode ser explicada por muitos fatores, entre eles a separação do hardware e do software, dois elementos que formam um computador e que, na maioria das vezes, são desenvolvidos em paralelo.

Considerando essas informações e o conteúdo estudado sobre a evolução do desenvolvimento de software, pode-se afirmar que:

1. ☐

os sistemas embarcados são uma categoria de sistemas em que o software é encapsulado e dedicado a um hardware específico.

Pergunta 3

0 ponto

Leia o trecho a seguir:

“Os projetistas da UML não estavam preocupados com os detalhes semânticos da linguagem, mas com sua expressividade. Eles introduziram noções úteis, como diagramas de caso de uso, que ajudam com o projeto, mas são demasiadamente informais para apoiar a execução.” Fonte: SOMMERVILLE, I. Engenharia de Software. 9. ed. São Paulo: Pearson Prentice Hall, 2011. p. 100.

Considerando o trecho apresentado e o conceito estudado sobre objetivos da UML, analise as afirmativas a seguir.

-
- I. A UML visa encorajar o crescimento do mercado de ferramentas de software orientadas a objeto.
- II. A UML objetiva ser um guia simplificado de desenvolvimento, para atrair cada vez mais desenvolvedores que desejam usá-la como um método conciso de desenvolvimento.
- III. A UML tem como alvo estar unida a órgãos de checagem automática de código para a comercialização de selos de qualidade e aderência na linguagem.
- IV. A UML mira em suportar conceitos de desenvolvimento de alto nível como componentes, colaboração, frameworks e padrões.
- Está correto apenas o que se afirma em:

2. ☐

I e IV

3.

Pergunta 4

0 ponto

Leia o trecho a seguir:

“Para compreender a UML, você precisará formar um modelo conceitual da linguagem e isso pressupõe aprender três elementos principais: os blocos de construção básicos da UML, as regras que determinam como esses blocos poderão ser combinados e alguns mecanismos comuns aplicados na UML.”Fonte: BOOCH, G., RUMBAUGH, J., JACOBSON, I. UML, Guia do Usuário. 2. ed. Rio de Janeiro: Editora Campus, 2005. Versão digital. p. 52.

Considerando o conceito estudado sobre a arquitetura de modelos da UML, analise as afirmativas a seguir.

- I. Em sua versão 2.5.1, a UML contém 16 tipos de diagramas, divididos em duas categorias: de estrutura, com 7 diagramas, e de comportamento, com 8 diagramas, e 1 transversal, que está nas duas categorias.
- II. Em um diagrama de estrutura, pode ser apresentada a composição de classes de objetos e seus relacionamentos em um sistema.
- III. A UML prevê a modelagem do software em camadas, com diferentes visões, sendo que alguns diagramas são mais gerais, com uma visão externa do sistema.
- IV. Os diagramas de estrutura, pelo seu caráter de arquitetura flexível, podem mostrar detalhes do comportamento de um objeto ao longo do tempo.
- Está correto apenas o que se afirma em:

1. ☐

2. ☐

II e III.

Pergunta 5

0 ponto

A Implementação é a atividade central do desenvolvimento de software. É nesta atividade que o sistema será efetivamente construído, em um processo no qual todas as funcionalidades presentes no documento de requisitos são transformadas em um sistema funcional.

Considerando essas informações e o conteúdo estudado sobre artefatos do projeto de software e itens de configuração, pode-se afirmar que:

3. ☐

o código-fonte é o conjunto de instruções construído de forma lógica e ordenada.

Pergunta 6

0 ponto

A empresa de celulares Ligapramim é líder no mercado latino-americano. Esta empresa utiliza a tecnologia Roboid como sistema operacional de seus celulares. O sistema Roboid está na versão 3.0, e se prepara para lançar uma versão completamente nova, com um novo conjunto de funcionalidades para chamar a atenção dos usuários. Sendo assim, a Ligapramim está se preparando para lançar essa nova versão em seus celulares. Considerando essas informações e o conteúdo estudado sobre versionamento na gerência de configuração, pode-se afirmar que:

4. ☐

segundo o versionamento X.Y.Z, a nova versão do sistema será lançada como a versão 4.0.

4.

Pergunta 7

0 ponto

Artefatos de software são comumente conhecidos como subprodutos do processo de desenvolvimento de software. Define-se como artefato do processo todo material produzido pelos engenheiros de software, obtido como resultado de uma atividade específica e que pode ser consumido por outras atividades ao longo do desenvolvimento do projeto.

Fonte: BOURQUE, P.; FAIRLEY, R. Guide to the software engineering body of knowledge. Los Alamitos: IEEE Computer Society Press, 2014.

Considerando essas informações e o conteúdo estudado sobre artefatos do projeto de software, analise as asserções a seguir e a relação proposta entre elas.

I. As atividades iniciais do processo de desenvolvimento de software estão focadas em levantar as necessidades dos usuários para serem transformadas em software. Assim, o documento de requisitos é um dos principais artefatos produzidos no início do desenvolvimento.

Porque:

II. O documento de requisitos reúne a lista completa de funcionalidades que um sistema deve possuir, sendo utilizado posteriormente pelas demais atividades do processo, como na programação e nos testes.

A seguir, assinale a alternativa correta:

1. ☐

2. ☐

3. ☐

4. ☐

As assertões I e II são proporções verdadeiras, e a II é uma justificativa correta da I.

5.

Pergunta 8

0 ponto

Ao dividir o desenvolvimento de um sistema em atividades específicas, é possível dividir tarefas e trabalhar com profissionais especialistas executando funções que visam atingir um objetivo específico do projeto, como entender o problema do usuário, construir modelos do software ou planejar a estrutura das telas do sistema, por exemplo.

Fonte: BOURQUE, P.; FAIRLEY, R. Guide to the software engineering body of knowledge. Los Alamitos: IEEE Computer Society Press, 2014.

Considerando essas informações e o conteúdo estudado sobre artefatos do projeto de software, pode-se afirmar que:

1. ☐
2. ☐

os diagramas UML são artefatos da atividade de Análise e Projeto de Software.

6.

Pergunta 9

0 ponto

Leia o trecho a seguir:

“Ferramentas CASE (Computer-Aided Software Engineering ou Engenharia de Software Auxiliada por Computador) são softwares que, de alguma maneira, colaboram para a execução de uma ou mais atividades realizadas durante o processo de engenharia de software.” Fonte: GUEDES, G. T. A. UML 2: uma abordagem prática. 2. ed. São Paulo: Novatec Editora, 2011. p. 41.

Considerando o trecho apresentado e o conceito estudado sobre Bootstrap, analise as afirmativas a seguir.

- I. Escolher uma ferramenta CASE é uma tarefa relativamente fácil e rápida, já que existem poucas opções no mercado com recursos de apoio ao desenvolvimento semelhantes.
 - II. A maioria das ferramentas CASE disponíveis no mercado provê suporte para a UML, já que essa linguagem proporciona regras de sintaxe clara adequadas para serem apoiadas por ferramentas de software.
 - III. Ferramentas convencionais de edição gráfica, como o Microsoft Paint, apoiam o desenho de modelos de modo similar ao das ferramentas CASE de modelagem de software, já que possibilitam diagramação de atores, casos de uso e classes com checagem de regras da UML.
 - IV. Além de auxiliar na diagramação de desenhos, as ferramentas CASE podem emitir relatórios, gerar códigos fonte e controlar de atividades de desenvolvimento.
- Está correto apenas o que se afirma em:

1. ☐
2. ☐

II e IV.

7.

Pergunta 10

0 ponto

Leia o trecho a seguir:

“Um esquema é a especificação de um modelo usando uma determinada linguagem, a qual pode ser formal ou informal (por exemplo, linguagem natural), textual ou gráfica. Quando a representação do esquema é gráfica, designa-se usualmente por diagrama.” Fonte: DA SILVA, A. M. R. e VIDEIRA, C. A. E. UML, Metodologias e Ferramentas CASE. Lisboa: Editora Centro Atlântico. 2001. p. 34.

Considerando essas informações e o conteúdo estudado, analise as afirmações a seguir sobre a taxonomia de diagramas da UML e assinale V para a(s) verdadeira(s) e F para a(s) falsa(s).

I. () O Diagrama de Perfil, ou Profile Diagram, está dentro da subcategoria de diagramas de interação, permitindo a definição de novos elementos UML e suas relações dinâmicas.

II. () A subcategoria de diagramas chamada de Interação possui 4 tipos de diagramas que descrevem como grupo de elementos de modelos colaboram em um determinado comportamento.

III. () A categoria de diagramas de Estrutura possui 7 diagramas que apresentam detalhes do sistema de modo estático, incluindo o Diagrama de Classes, representando as classes de objetos e suas ligações.

IV. () O Diagrama de Casos de Uso é um dos mais conhecidos da UML, sendo da categoria de Comportamento e estando na subcategoria de Interação, por modelar os contatos interativos de atores com itens do sistema de modo externo.

Agora, assinale a alternativa que apresenta a sequência correta:

1. ☐

F, F, V, V.

2. ☐

F, V, V, V.

3. ☐

F, V, V, F.

4. ☐

V, V, F, V.

5. ☐

V, F, V, F.

TENTATIVA 01-UNIDADE 03

Leia o excerto a seguir:

“Em seus diagramas, os objetos ativos poderão aparecer sempre que houver objetos passivos. Você pode fazer a modelagem da colaboração de objetos ativos e passivos, incluindo diagramas de colaboração e de sequências. Um objeto ativo poderá aparecer como o destino de um evento em uma máquina de estados.”Fonte: BOOCH, G.; RUMBAUGH, J.; JACOBSON, I. UML: guia do usuário. Rio de Janeiro: Elsevier, 2012. p. 95.

Considerando essas informações e o conteúdo estudado sobre communication diagram, pode-se afirmar que o diagrama de colaboração:



faz parte dos diagramas de interação

Leia o excerto a seguir:

“O estado de um objeto é, portanto, dinâmico. Assim, ao visualizar seu estado, você está realmente especificando o valor de seu estado em um determinado momento no tempo e no espaço. É possível exibir o estado de alteração do objeto, mostrando-o várias vezes no mesmo diagrama de interação, mas com cada ocorrência representando um estado diferente.”Fonte: BOOCH, G.; RUMBAUGH, J.; JACOBSON, I. UML: guia do usuário. Rio de Janeiro: Elsevier, 2012. p. 183.

Considerando essas informações e o conteúdo estudado sobre o interaction overview diagram, pode-se afirmar que um dos exemplos do uso do diagrama de interação é:

a inserção de mensagens enviadas e recebidas entre os objetos.

Leia o excerto a seguir:

“Como classificadores, especificações de instância têm atributos que são referidos como slots. Uma especificação de instância pode ter um slot para cada recurso estrutural de seu classificador, incluindo recursos herdados. Você pode especificar valores para cada slot em uma especificação de instância, pois um tipo válido é definido para o slot.”

Fonte: IBM. Especificações em instâncias na UML. Disponível em: <https://www.ibm.com/support/knowledgecenter/pt-br/SS5JSH_9.5.0/com.ibm.xtools.modeler.doc/topics/cinstancespec.html>. Acesso em: 05 nov. 2019.

Considerando essas informações e o conteúdo estudado sobre o object diagram, pode-se afirmar que as especificações de instâncias são:

componentes que simbolizam uma instância presente em um sistema moldado.

UX, do inglês User Experience, ou Experiência do Usuário, em português, é um termo que dentro do desenvolvimento de sistemas define o conjunto de ações que visa entender como o usuário final se sente ao utilizar um sistema que será lançado.

Considerando essas informações e o conteúdo estudado sobre geração de baselines e releases, pode-se afirmar que:

a user experience é uma atividade focada na interação do usuário com o sistema.

O ciclo de vida de mudança define o conjunto de atividades que deve ser realizado para que as alterações em uma versão de um sistema sejam efetivamente analisadas, realizadas e encerradas. Com isso, é esperado que, mesmo com a mudança, um sistema continue funcionando sem problemas.

Considerando essas informações e o conteúdo estudado sobre ciclo de vida de mudanças, analise as afirmativas a seguir.

- I. Uma solicitação de mudança pode ser negada imediatamente após o pedido ser realizado, então a solicitação é finalizada e o ciclo de vida da mudança é encerrado.
 - II. Uma solicitação de uma mudança simples pode ser aceita e o ciclo de vida da mudança é reduzido devido à baixa complexidade da solicitação.
 - III. Uma vez que a mudança é realizada, o procedimento requer o encerramento do ciclo e a imediata liberação da nova versão atualizada.
 - IV. O ciclo de vida de mudança não permite que modificações paralelas sejam realizadas ao mesmo tempo na mesma parte do sistema.
- Está correto apenas o que se afirma em:

I e IV.

Leia o excerto a seguir:

“Você pode utilizar o diagrama de estrutura composta para mostrar os detalhes internos de um classificador e descrever os objetos e funções que trabalham juntos para executar o comportamento do classificador contido. Um diagrama de estrutura composta é similar a um diagrama de classe, mas ele representa peças individuais em vez de classes inteiras.”

Fonte: IBM. Diagramas de estrutura composta. Disponível em: <https://www.ibm.com/support/knowledgecenter/pt-br/SS5JSH_9.5.0/com.ibm.xtools.modeler.doc/topics/ccompstruc.html>. Acesso em: 05 nov. 2019.

O trecho apresentado trata do uso dos diagramas de estrutura composta. Considerando essas informações e o conteúdo estudado sobre composite structure diagram, pode-se afirmar que as peças são:

componentes de diagrama que simbolizam um grupo de diversas instâncias apresentados em um classificador estruturado contido.

Leia o excerto a seguir:

“Uma das coisas para as quais os objetos serão utilizados será a modelagem de instâncias concretas que existem no mundo real. Por exemplo, para fazer a modelagem da topologia da rede de uma empresa, você usará diagramas de implantação.” Fonte: BOOCH, G.; RUMBAUGH, J.; JACOBSON, I. UML: guia do usuário. Rio de Janeiro: Elsevier, 2012. p. 285.

Considerando essas informações e o conteúdo estudado sobre o component diagram, pode-se afirmar que, para utilizar uma visão estática adequada para a implantação de um determinado sistema, é necessário:

adotar uma modelagem de componentes físicos que se localizam em um nó.

Leia o excerto a seguir:

“O diagrama de classes, especificamente em UML 2, considera os diagramas de estrutura como uma classificação; não existe um diagrama em si chamado ‘diagrama de estrutura’. No entanto, o diagrama de classes oferece um ótimo exemplo do tipo de diagrama de estrutura e fornece um conjunto inicial de elementos de notação que todos os outros diagramas de estrutura usam.”

Fonte: BELL, D. O diagrama de classes. 2016. Disponível em: <<https://www.ibm.com/developerworks/br/rational/library/content/RationalEdge/sep04/bell/index.html>>. Acesso em: 05



apresentar pacotes ou subsistemas utilizados para reunir elementos do seu modelo dentro de um grupo maior.

Leia o excerto a seguir:

“No Visual Studio, você pode desenhar um diagrama de atividade para descrever um processo de negócios ou um algoritmo de software como um fluxo de trabalho por meio de uma série de ações. Pessoas, componentes de software ou dispositivos podem executar

essas ações.”Fonte: MICROSOFT. Diagramas de atividade UML: diretrizes. 2016. Disponível em: <<https://docs.microsoft.com/pt-br/visualstudio/modeling/uml-activity-diagrams-guidelines?view=vs-2015>>. Acesso em: 05 nov. 2019.

O trecho apresentado trata de uma das funcionalidades dos diagramas de atividades com o uso do Visual Studio. Considerando essas informações e o conteúdo estudado sobre o activity diagram, analise as afirmativas a seguir e assinale V para a(s) verdadeira(s) e F para a(s) falsa(s).



F, F, V, V.

A gerência de configuração é uma atividade do desenvolvimento de software focada no controle das mudanças que podem ocorrer no sistema durante o seu desenvolvimento. Nesse processo, o ciclo de vida de mudanças define de maneira estruturada como uma mudança deve acontecer em um sistema em desenvolvimento ou após ser lançado.

Considerando essas informações e o conteúdo estudado sobre ciclo de vida de mudanças, analise as afirmativas a seguir.

- I. Avaliar a necessidade de uma mudança faz parte do ciclo de vida de mudanças.
- II. Definir o ambiente de programação onde uma mudança é realizada faz parte do ciclo de vida de mudanças.
- III. Planejar uma mudança e seus impactos no sistema faz parte do ciclo de vida de mudanças.
- IV. Estabelecer os custos financeiros de uma mudança faz parte do ciclo de vida de mudanças.

Está correto apenas o que se afirma em:



I e III.