

**-- CONHECIMENTOS ESPECÍFICOS --****Questão 31**

No contexto do desenvolvimento de *software*, o processo iterativo e incremental é

- A um método com enfoque na integração de componentes reutilizáveis em um sistema já existente, evitando a necessidade de desenvolvê-lo desde o início.
- B uma abordagem que intercala as atividades de especificação, desenvolvimento e validação, criando versões sucessivas que adicionam novas funcionalidades ao sistema.
- C um modelo linear de desenvolvimento em que cada fase deve ser concluída antes que a próxima comece, sem revisões ou interações.
- D uma técnica que utiliza incrementos pequenos, envolvendo clientes para *feedback* rápido e priorizando a comunicação informal em detrimento da documentação formal.
- E um modelo que representa o processo de *software* como ciclos, cada qual abordando fases como viabilidade, requisitos e projeto, com foco em riscos.

**||JUSTIFICATIVAS||**

||A|| - Incorreta. Esse conceito refere-se à engenharia de *software* orientada a reuso, abordagem baseada na existência de um número significativo de componentes reutilizáveis. O processo de desenvolvimento do sistema concentra-se na integração desses componentes em um sistema já existente em vez de desenvolver um sistema a partir do zero.\*/\*

||B|| - Correta. O desenvolvimento iterativo e incremental de *software* facilita e reduz custos ao implementar mudanças durante o processo. Cada versão adiciona funcionalidades necessárias, priorizando as mais importantes ou urgentes. Isso permite ao cliente avaliar o sistema cedo e, se necessário, ajustar apenas o incremento em desenvolvimento, redefinindo funcionalidades para versões futuras. Essa abordagem intercala as atividades de especificação, desenvolvimento e validação.

O sistema é desenvolvido como uma série de versões (incrementos), de maneira que cada versão adiciona funcionalidade à anterior.\*/\*

||C|| - Incorreta. Trata-se do modelo cascata, o qual considera as atividades fundamentais do processo de especificação, desenvolvimento, validação e evolução, e representa cada uma delas como fases distintas, como: especificação de requisitos, projeto de *software*, implementação, teste, e assim por diante.\*/\*

||D|| - Incorreta. Trata-se das metodologias ágeis, que focam no desenvolvimento rápido de *software* e produzem sistemas úteis rapidamente, em incrementos pequenos, com funcionalidades adicionais a cada versão. Os métodos ágeis minimizam a documentação, intercalam especificação, projeto e implementação, e envolvem usuários na avaliação e definição de requisitos em cada etapa, priorizando comunicação informal sobre documentação formal.\*/\*

||E|| - Incorreta. Trata-se do modelo espiral de Boehm, um *framework* de processo de *software* dirigido a riscos proposto por Boehm, no qual o processo de *software* é representado como uma espiral, e não como uma sequência e atividades com alguns retornos de uma para outra. Cada volta na espiral representa uma fase do processo de *software*. Dessa forma, a volta mais interna pode preocupar-se com a viabilidade do sistema; o ciclo seguinte, com definição de requisitos; o seguinte, com o projeto do sistema; e assim por diante.\*/\*

**Questão 32**

Uma equipe de desenvolvimento de *software* está trabalhando em uma aplicação *web* e necessita adotar uma ferramenta de integração de código e criação de *pipelines* de implantação, a fim de automatizar o processo de integração e entrega contínua.

Nessa situação, a ferramenta mais indicada para o atendimento da necessidade da equipe é o

- A Vagrant.
- B Docker.
- C Ansible.
- D Kubernetes.
- E GitLab CI.

**||JUSTIFICATIVAS||**

||A|| - Incorreta. O Vagrant é uma ferramenta de automação para criação e gerenciamento de ambientes de desenvolvimento virtualizados. Ele simplifica o processo de configurar ambientes de máquinas virtuais (VMs) usando provedores como VirtualBox, VMware ou Hyper-V.\*/\*

||B|| - Incorreta. Docker é uma ferramenta de *containerização*, não integração contínua (CI). Ela cria e gerencia *containers*, que são ambientes isolados onde as aplicações podem ser executadas de forma consistente em diferentes ambientes.\*/\*

||C|| - Incorreta. Ansible é uma ferramenta de automação de configuração de infraestrutura e gerenciamento de servidores. Pode ser usado para provisionar e configurar sistemas, mas não foca na integração contínua (CI).\*/

||D|| - Incorreta. Kubernetes é uma ferramenta de orquestração de *containers*, não de CI/CD. Embora facilite o gerenciamento e a automação de *containers*, ele não é especificamente uma ferramenta para automação de integração e entrega contínua.\*/\*

||E|| - Correta. O GitLab CI é uma ferramenta para integração contínua (CI) que permite automatizar o processo de integração de código, criar *pipelines* de integração e entrega contínua (CD) e gerenciar a implantação de *software*. Ele facilita a automação de testes, construção e entrega de código em várias fases de desenvolvimento, proporcionando uma integração eficiente e contínua.\*/\*

**Questão 33**

Uma empresa pode disponibilizar atualizações de *software* mais rápidas e confiáveis aos seus clientes ao implementar práticas de DevOps que garantam

- I a entrega contínua das alterações de código para liberação em produção.
- II publicações esparsas a fim de evitar a geração de grandes quantidades de versões do *software*.
- III a adoção de infraestrutura como código para provisionamento e gerenciamento automatizado da infraestrutura.

Assinale a opção correta.

- A Apenas o item I está certo.
- B Apenas o item II está certo.
- C Apenas os itens I e III estão certos.
- D Apenas os itens II e III estão certos.
- E Todos os itens estão certos.

**||JUSTIFICATIVAS||**

||A|| - Incorreta. Os itens I e III estão certos.\*/  
 ||B|| - Incorreta. O item II está errado.\*/  
 ||C|| - Correta. O item I está certo, pois a entrega contínua é uma prática de desenvolvimento de *software* em que alterações de código são criadas, testadas e preparadas automaticamente para liberação para produção. Ela se expande com base na integração contínua, pela implantação de todas as alterações de código em um ambiente de teste e(ou) ambiente de produção, após o estágio de criação. Quando a integração contínua for implementada adequadamente, os desenvolvedores sempre terão um artefato de criação pronto para ser implantado e que passou por um processo de teste padronizado.

||D|| - Incorreta. O item II está errado.\*/  
 ||E|| - Incorreta. Apenas os itens I e III estão certos.\*/  
 O item II está errado, pois a publicação deve ser frequente e não esparsa. A integração contínua é uma prática de desenvolvimento de *software* em que os desenvolvedores, com frequência, juntam suas alterações de código em um repositório central. Depois disso, criações e testes são executados. Os principais objetivos da integração contínua são encontrar e investigar erros mais rapidamente, melhorar a qualidade do *software* e reduzir o tempo necessário para validar e lançar novas atualizações de *software*.

O item III está certo, pois a infraestrutura como código é uma prática em que a infraestrutura é provisionada e gerenciada mediante técnicas de desenvolvimento de código e *software*, como controle de versão e integração contínua. O modelo controlado por API da nuvem permite que desenvolvedores e administradores de sistema interajam com a infraestrutura de modo programático e em escala, em vez de precisarem instalar e configurar manualmente os recursos.\*/  
 O item II está errado, pois a publicação deve ser frequente e não esparsa. A integração contínua é uma prática de desenvolvimento de *software* em que os desenvolvedores, com frequência, juntam suas alterações de código em um repositório central. Depois disso, criações e testes são executados. Os principais objetivos da integração contínua são encontrar e investigar erros mais rapidamente, melhorar a qualidade do *software* e reduzir o tempo necessário para validar e lançar novas atualizações de *software*.

O item III está certo, pois a infraestrutura como código é uma prática em que a infraestrutura é provisionada e gerenciada mediante técnicas de desenvolvimento de código e *software*, como controle de versão e integração contínua. O modelo controlado por API da nuvem permite que desenvolvedores e administradores de sistema interajam com a infraestrutura de modo programático e em escala, em vez de precisarem instalar e configurar manualmente os recursos.\*/  
 O item II está errado, pois a publicação deve ser frequente e não esparsa. A integração contínua é uma prática de desenvolvimento de *software* em que os desenvolvedores, com frequência, juntam suas alterações de código em um repositório central. Depois disso, criações e testes são executados. Os principais objetivos da integração contínua são encontrar e investigar erros mais rapidamente, melhorar a qualidade do *software* e reduzir o tempo necessário para validar e lançar novas atualizações de *software*.

||D|| - Incorreta. O item II está errado.\*/  
 ||E|| - Incorreta. Apenas os itens I e III estão certos.\*/  
 O item II está errado, pois a publicação deve ser frequente e não esparsa. A integração contínua é uma prática de desenvolvimento de *software* em que os desenvolvedores, com frequência, juntam suas alterações de código em um repositório central. Depois disso, criações e testes são executados. Os principais objetivos da integração contínua são encontrar e investigar erros mais rapidamente, melhorar a qualidade do *software* e reduzir o tempo necessário para validar e lançar novas atualizações de *software*.

||E|| - Incorreta. Apenas os itens I e III estão certos.\*/  
 O item II está errado, pois a publicação deve ser frequente e não esparsa. A integração contínua é uma prática de desenvolvimento de *software* em que os desenvolvedores, com frequência, juntam suas alterações de código em um repositório central. Depois disso, criações e testes são executados. Os principais objetivos da integração contínua são encontrar e investigar erros mais rapidamente, melhorar a qualidade do *software* e reduzir o tempo necessário para validar e lançar novas atualizações de *software*.

**Questão 34**

Considerando que uma equipe de *designers* esteja utilizando *storytelling* para obter *insights* sobre os usuários e, dessa forma, criar empatia com seus usuários-alvo, assinale a opção correta em relação ao *storytelling*.

- A As histórias são apenas descrições de uma cena ou de um personagem.
- B O principal objetivo das histórias é transferir as informações de uma pessoa para outra.
- C As histórias instigam a imaginação dos membros da audiência.
- D Histórias são criadas necessariamente para atender a múltiplos públicos.
- E Uma história deve ser contada, de uma vez, até o final, a fim de não comprometer a compreensão da audiência.

**||JUSTIFICATIVAS||**

||A|| - Incorreta. As histórias não se limitam a descrever cenas ou personagens. A mera descrição de uma cena ou de um personagem não é realmente uma história até que haja eventos, decisões e ações, ou pelo menos uma reação a uma situação ou ambiente.\*/  
 ||B|| - Incorreta. O objetivo das histórias vai além de apenas transferir informações. Elas são criadas para engajar a audiência, estimular a imaginação, criar empatia, provocar reflexões e gerar compreensão compartilhada.\*/  
 ||C|| - Correta. As histórias ganham vida na imaginação dos membros da audiência, seja uma pessoa ou centenas. A ideia central do *storytelling* é envolver a audiência e estimular sua imaginação.\*/  
 ||D|| - Incorreta. As histórias, em *design* de experiência do usuário, geralmente são criadas para um público específico e para atender a um objetivo específico, não necessariamente para atender a múltiplos públicos ao mesmo tempo.\*/  
 ||E|| - Incorreta. As histórias, em *design* de experiência do usuário, geralmente são criadas para um público específico e para atender a um objetivo específico, não necessariamente para atender a múltiplos públicos ao mesmo tempo.\*/  
 O item II está errado, pois a publicação deve ser frequente e não esparsa. A integração contínua é uma prática de desenvolvimento de *software* em que os desenvolvedores, com frequência, juntam suas alterações de código em um repositório central. Depois disso, criações e testes são executados. Os principais objetivos da integração contínua são encontrar e investigar erros mais rapidamente, melhorar a qualidade do *software* e reduzir o tempo necessário para validar e lançar novas atualizações de *software*.

||B|| - Incorreta. O objetivo das histórias vai além de apenas transferir informações. Elas são criadas para engajar a audiência, estimular a imaginação, criar empatia, provocar reflexões e gerar compreensão compartilhada.\*/  
 ||C|| - Correta. As histórias ganham vida na imaginação dos membros da audiência, seja uma pessoa ou centenas. A ideia central do *storytelling* é envolver a audiência e estimular sua imaginação.\*/  
 ||D|| - Incorreta. As histórias, em *design* de experiência do usuário, geralmente são criadas para um público específico e para atender a um objetivo específico, não necessariamente para atender a múltiplos públicos ao mesmo tempo.\*/  
 ||E|| - Incorreta. As histórias, em *design* de experiência do usuário, geralmente são criadas para um público específico e para atender a um objetivo específico, não necessariamente para atender a múltiplos públicos ao mesmo tempo.\*/  
 O item II está errado, pois a publicação deve ser frequente e não esparsa. A integração contínua é uma prática de desenvolvimento de *software* em que os desenvolvedores, com frequência, juntam suas alterações de código em um repositório central. Depois disso, criações e testes são executados. Os principais objetivos da integração contínua são encontrar e investigar erros mais rapidamente, melhorar a qualidade do *software* e reduzir o tempo necessário para validar e lançar novas atualizações de *software*.

||C|| - Correta. As histórias ganham vida na imaginação dos membros da audiência, seja uma pessoa ou centenas. A ideia central do *storytelling* é envolver a audiência e estimular sua imaginação.\*/  
 ||D|| - Incorreta. As histórias, em *design* de experiência do usuário, geralmente são criadas para um público específico e para atender a um objetivo específico, não necessariamente para atender a múltiplos públicos ao mesmo tempo.\*/  
 ||E|| - Incorreta. As histórias, em *design* de experiência do usuário, geralmente são criadas para um público específico e para atender a um objetivo específico, não necessariamente para atender a múltiplos públicos ao mesmo tempo.\*/  
 O item II está errado, pois a publicação deve ser frequente e não esparsa. A integração contínua é uma prática de desenvolvimento de *software* em que os desenvolvedores, com frequência, juntam suas alterações de código em um repositório central. Depois disso, criações e testes são executados. Os principais objetivos da integração contínua são encontrar e investigar erros mais rapidamente, melhorar a qualidade do *software* e reduzir o tempo necessário para validar e lançar novas atualizações de *software*.

||D|| - Incorreta. As histórias, em *design* de experiência do usuário, geralmente são criadas para um público específico e para atender a um objetivo específico, não necessariamente para atender a múltiplos públicos ao mesmo tempo.\*/  
 ||E|| - Incorreta. As histórias, em *design* de experiência do usuário, geralmente são criadas para um público específico e para atender a um objetivo específico, não necessariamente para atender a múltiplos públicos ao mesmo tempo.\*/  
 O item II está errado, pois a publicação deve ser frequente e não esparsa. A integração contínua é uma prática de desenvolvimento de *software* em que os desenvolvedores, com frequência, juntam suas alterações de código em um repositório central. Depois disso, criações e testes são executados. Os principais objetivos da integração contínua são encontrar e investigar erros mais rapidamente, melhorar a qualidade do *software* e reduzir o tempo necessário para validar e lançar novas atualizações de *software*.

||E|| - Incorreta. Em muitos casos de *storytelling*, não é preciso contar a história até o final de uma vez. O objetivo do *storytelling* é engajar a audiência, muitas vezes criando uma jornada emocional ou uma narrativa que pode ser contada de forma fragmentada ou interativa. Em algumas abordagens, contar a história de forma incompleta ou criar suspense pode ser uma técnica eficaz para manter o interesse e provocar mais reflexão. Isso pode ser especialmente útil em contextos como *design* de experiência do usuário, em que uma parte da história pode ser usada para provocar questões ou incentivar a descoberta do problema e da solução.\*/  
 ||A|| - Incorreta. Em muitos casos de *storytelling*, não é preciso contar a história até o final de uma vez. O objetivo do *storytelling* é engajar a audiência, muitas vezes criando uma jornada emocional ou uma narrativa que pode ser contada de forma fragmentada ou interativa. Em algumas abordagens, contar a história de forma incompleta ou criar suspense pode ser uma técnica eficaz para manter o interesse e provocar mais reflexão. Isso pode ser especialmente útil em contextos como *design* de experiência do usuário, em que uma parte da história pode ser usada para provocar questões ou incentivar a descoberta do problema e da solução.\*/  
 ||B|| - Incorreta. Em muitos casos de *storytelling*, não é preciso contar a história até o final de uma vez. O objetivo do *storytelling* é engajar a audiência, muitas vezes criando uma jornada emocional ou uma narrativa que pode ser contada de forma fragmentada ou interativa. Em algumas abordagens, contar a história de forma incompleta ou criar suspense pode ser uma técnica eficaz para manter o interesse e provocar mais reflexão. Isso pode ser especialmente útil em contextos como *design* de experiência do usuário, em que uma parte da história pode ser usada para provocar questões ou incentivar a descoberta do problema e da solução.\*/  
 ||C|| - Correta. Em muitos casos de *storytelling*, não é preciso contar a história até o final de uma vez. O objetivo do *storytelling* é engajar a audiência, muitas vezes criando uma jornada emocional ou uma narrativa que pode ser contada de forma fragmentada ou interativa. Em algumas abordagens, contar a história de forma incompleta ou criar suspense pode ser uma técnica eficaz para manter o interesse e provocar mais reflexão. Isso pode ser especialmente útil em contextos como *design* de experiência do usuário, em que uma parte da história pode ser usada para provocar questões ou incentivar a descoberta do problema e da solução.\*/  
 ||D|| - Incorreta. Em muitos casos de *storytelling*, não é preciso contar a história até o final de uma vez. O objetivo do *storytelling* é engajar a audiência, muitas vezes criando uma jornada emocional ou uma narrativa que pode ser contada de forma fragmentada ou interativa. Em algumas abordagens, contar a história de forma incompleta ou criar suspense pode ser uma técnica eficaz para manter o interesse e provocar mais reflexão. Isso pode ser especialmente útil em contextos como *design* de experiência do usuário, em que uma parte da história pode ser usada para provocar questões ou incentivar a descoberta do problema e da solução.\*/  
 ||E|| - Incorreta. Em muitos casos de *storytelling*, não é preciso contar a história até o final de uma vez. O objetivo do *storytelling* é engajar a audiência, muitas vezes criando uma jornada emocional ou uma narrativa que pode ser contada de forma fragmentada ou interativa. Em algumas abordagens, contar a história de forma incompleta ou criar suspense pode ser uma técnica eficaz para manter o interesse e provocar mais reflexão. Isso pode ser especialmente útil em contextos como *design* de experiência do usuário, em que uma parte da história pode ser usada para provocar questões ou incentivar a descoberta do problema e da solução.\*/  
 O item II está errado, pois a publicação deve ser frequente e não esparsa. A integração contínua é uma prática de desenvolvimento de *software* em que os desenvolvedores, com frequência, juntam suas alterações de código em um repositório central. Depois disso, criações e testes são executados. Os principais objetivos da integração contínua são encontrar e investigar erros mais rapidamente, melhorar a qualidade do *software* e reduzir o tempo necessário para validar e lançar novas atualizações de *software*.

**Questão 35**

No que se refere a histórias de usuário, assinale a opção correta.

- A Nas histórias de usuário, há apenas um artefato de requisitos vinculados à história.
- B A história de usuário deve ser escrita em uma linguagem técnica e formal.
- C A história de usuário é a maior unidade de trabalho baseada na necessidade do cliente final.
- D O objetivo da história de usuário é expressar, de forma clara e leve, as angústias e necessidades do cliente final.
- E Histórias equivalem a casos de uso, já que ambos descrevem requisitos e cumprem o mesmo papel.

**||JUSTIFICATIVAS||**

||A|| - Incorreta. Histórias de usuário podem estar vinculadas a vários artefatos, como critérios de aceitação, diagramas, entre outros.\*/  
 ||B|| - Incorreta. Histórias de usuário devem ser escritas em linguagem simples e compreensível para todos os envolvidos, incluído o cliente.\*/  
 ||C|| - Incorreta. História de usuário é a menor unidade de trabalho com base na necessidade do cliente final que vai utilizar e(ou) interagir com o produto. Histórias de usuário frequentemente representam fragmentos menores de uma necessidade do cliente final.\*/  
 ||D|| - Correta. Uma história é a representação clara e informal que expressa a necessidade e(ou) requisito de um potencial usuário. O objetivo da história é expressar, em poucas palavras, as dores e necessidades, propor critérios mínimos para aceitação e o valor de negócio agregado, de uma forma clara, limpa e leve.\*/  
 ||E|| - Incorreta. Histórias de usuário e casos de uso têm objetivos e estruturas diferentes; histórias são mais focadas no entendimento do valor para o usuário, enquanto casos de uso detalham fluxos de interação específicos.\*/  
 ||A|| - Incorreta. Histórias de usuário podem estar vinculadas a vários artefatos, como critérios de aceitação, diagramas, entre outros.\*/  
 ||B|| - Incorreta. Histórias de usuário devem ser escritas em linguagem simples e compreensível para todos os envolvidos, incluído o cliente.\*/  
 ||C|| - Incorreta. História de usuário é a menor unidade de trabalho com base na necessidade do cliente final que vai utilizar e(ou) interagir com o produto. Histórias de usuário frequentemente representam fragmentos menores de uma necessidade do cliente final.\*/  
 ||D|| - Correta. Uma história é a representação clara e informal que expressa a necessidade e(ou) requisito de um potencial usuário. O objetivo da história é expressar, em poucas palavras, as dores e necessidades, propor critérios mínimos para aceitação e o valor de negócio agregado, de uma forma clara, limpa e leve.\*/  
 ||E|| - Incorreta. Histórias de usuário e casos de uso têm objetivos e estruturas diferentes; histórias são mais focadas no entendimento do valor para o usuário, enquanto casos de uso detalham fluxos de interação específicos.\*/  
 O item II está errado, pois a publicação deve ser frequente e não esparsa. A integração contínua é uma prática de desenvolvimento de *software* em que os desenvolvedores, com frequência, juntam suas alterações de código em um repositório central. Depois disso, criações e testes são executados. Os principais objetivos da integração contínua são encontrar e investigar erros mais rapidamente, melhorar a qualidade do *software* e reduzir o tempo necessário para validar e lançar novas atualizações de *software*.

||B|| - Incorreta. Histórias de usuário devem ser escritas em linguagem simples e compreensível para todos os envolvidos, incluído o cliente.\*/  
 ||C|| - Incorreta. História de usuário é a menor unidade de trabalho com base na necessidade do cliente final que vai utilizar e(ou) interagir com o produto. Histórias de usuário frequentemente representam fragmentos menores de uma necessidade do cliente final.\*/  
 ||D|| - Correta. Uma história é a representação clara e informal que expressa a necessidade e(ou) requisito de um potencial usuário. O objetivo da história é expressar, em poucas palavras, as dores e necessidades, propor critérios mínimos para aceitação e o valor de negócio agregado, de uma forma clara, limpa e leve.\*/  
 ||E|| - Incorreta. Histórias de usuário e casos de uso têm objetivos e estruturas diferentes; histórias são mais focadas no entendimento do valor para o usuário, enquanto casos de uso detalham fluxos de interação específicos.\*/  
 O item II está errado, pois a publicação deve ser frequente e não esparsa. A integração contínua é uma prática de desenvolvimento de *software* em que os desenvolvedores, com frequência, juntam suas alterações de código em um repositório central. Depois disso, criações e testes são executados. Os principais objetivos da integração contínua são encontrar e investigar erros mais rapidamente, melhorar a qualidade do *software* e reduzir o tempo necessário para validar e lançar novas atualizações de *software*.

||C|| - Incorreta. História de usuário é a menor unidade de trabalho com base na necessidade do cliente final que vai utilizar e(ou) interagir com o produto. Histórias de usuário frequentemente representam fragmentos menores de uma necessidade do cliente final.\*/  
 ||D|| - Correta. Uma história é a representação clara e informal que expressa a necessidade e(ou) requisito de um potencial usuário. O objetivo da história é expressar, em poucas palavras, as dores e necessidades, propor critérios mínimos para aceitação e o valor de negócio agregado, de uma forma clara, limpa e leve.\*/  
 ||E|| - Incorreta. Histórias de usuário e casos de uso têm objetivos e estruturas diferentes; histórias são mais focadas no entendimento do valor para o usuário, enquanto casos de uso detalham fluxos de interação específicos.\*/  
 O item II está errado, pois a publicação deve ser frequente e não esparsa. A integração contínua é uma prática de desenvolvimento de *software* em que os desenvolvedores, com frequência, juntam suas alterações de código em um repositório central. Depois disso, criações e testes são executados. Os principais objetivos da integração contínua são encontrar e investigar erros mais rapidamente, melhorar a qualidade do *software* e reduzir o tempo necessário para validar e lançar novas atualizações de *software*.

||D|| - Correta. Uma história é a representação clara e informal que expressa a necessidade e(ou) requisito de um potencial usuário. O objetivo da história é expressar, em poucas palavras, as dores e necessidades, propor critérios mínimos para aceitação e o valor de negócio agregado, de uma forma clara, limpa e leve.\*/  
 ||E|| - Incorreta. Histórias de usuário e casos de uso têm objetivos e estruturas diferentes; histórias são mais focadas no entendimento do valor para o usuário, enquanto casos de uso detalham fluxos de interação específicos.\*/  
 O item II está errado, pois a publicação deve ser frequente e não esparsa. A integração contínua é uma prática de desenvolvimento de *software* em que os desenvolvedores, com frequência, juntam suas alterações de código em um repositório central. Depois disso, criações e testes são executados. Os principais objetivos da integração contínua são encontrar e investigar erros mais rapidamente, melhorar a qualidade do *software* e reduzir o tempo necessário para validar e lançar novas atualizações de *software*.

||E|| - Incorreta. Histórias de usuário e casos de uso têm objetivos e estruturas diferentes; histórias são mais focadas no entendimento do valor para o usuário, enquanto casos de uso detalham fluxos de interação específicos.\*/  
 O item II está errado, pois a publicação deve ser frequente e não esparsa. A integração contínua é uma prática de desenvolvimento de *software* em que os desenvolvedores, com frequência, juntam suas alterações de código em um repositório central. Depois disso, criações e testes são executados. Os principais objetivos da integração contínua são encontrar e investigar erros mais rapidamente, melhorar a qualidade do *software* e reduzir o tempo necessário para validar e lançar novas atualizações de *software*.

**Questão 36**

A respeito da arquitetura *cloud native*, de técnicas de componentização de *software* e da arquitetura orientada a serviços (SOA), assinale a opção correta.

- A Na arquitetura *cloud native*, os aplicativos são divididos em serviços pequenos fracamente acoplados que se comunicam entre si.
- B No desenvolvimento de serviços, o projeto de mensagem tem como objetivo a definição das operações, entradas, saídas e exceções associadas às operações.
- C Um serviço é um componente de *software* reutilizável e de alto acoplamento que encapsula funcionalidades acessíveis e distribuídas por programas.
- D Na arquitetura *cloud native*, os aplicativos podem ser dimensionados horizontalmente, aumentando o tamanho das máquinas virtuais.
- E O uso de componentes exige necessariamente a implementação de interface e o conhecimento do código-fonte para referenciá-lo diretamente em um programa.

**||JUSTIFICATIVAS||**

||A|| - Correta. Em uma arquitetura nativa da nuvem, os aplicativos são divididos em serviços menores, independentes e fracamente acoplados que podem ser desenvolvidos, implementados e gerenciados de forma independente. Esses serviços se comunicam entre si usando APIs e são empacotados e implantados por meio do uso de contêineres. Isso permite criar aplicativos que podem ser facilmente dimensionados, gerenciados e atualizados sem interrupção no sistema geral.\*/

||B|| - Incorreta. O projeto de mensagem no desenvolvimento de serviços tem como objetivo definir a estrutura das mensagens que serão trocadas entre o cliente e o serviço, ou seja, como os dados serão representados e organizados nas mensagens enviadas e recebidas. No estágio de projeto de mensagem, o foco está na maneira como os dados serão embalados, no formato da mensagem e na estrutura das informações, incluindo-se as entradas e saídas dos dados, mas não se foca diretamente na definição das operações e exceções. Essas definições são tratadas principalmente no projeto lógico da interface.\*/

||C|| - Incorreta. Um serviço é um componente de *software* de baixo acoplamento e reutilizável que encapsula funcionalidade discreta, que pode ser distribuída e acessada por meio de programas. Um *web service* é um serviço acessado por meio de protocolos baseados em padrões da Internet e em XML.\*/

||D|| - Incorreta. Na arquitetura *cloud native*, os aplicativos são dimensionados horizontalmente, adicionando-se novas instâncias de serviços ou contêineres, em vez de aumentar o tamanho (capacidade) das máquinas virtuais existentes, que seria um escalonamento vertical.\*/

||E|| - Incorreta. O uso de componentes não exige necessariamente a implementação de interface nem o conhecimento do código-fonte para referenciá-lo diretamente em um programa. Um componente deve ser independente; deve ser possível compor e implantá-lo sem a necessidade de usar outros componentes específicos. Se utilizarem serviços externos, os serviços devem ser definidos explicitamente em uma especificação de interface, como *requires*. O objetivo é permitir o uso dos componentes sem a necessidade de conhecer seus detalhes internos.\*/

**Questão 37**

Uma empresa, ao criar um sistema de rastreamento de pacotes em tempo real, implementou uma integração com diversos serviços de entrega que possuem diferentes requisitos de comunicação. Alguns serviços respondem apenas quando solicitados, enquanto outros não geram resposta.

Assinale a opção que cita o padrão adotado na situação hipotética precedente.

- A XMLHttpRequest
- B XSLT
- C SOAP
- D REST
- E WSDL

**||JUSTIFICATIVAS||**

||A|| - Incorreta. Objetos XMLHttpRequest (XHR) são usados para interagir com servidores. Eles permitem recuperar dados de uma URL sem a necessidade de atualizar toda a página. Isso possibilita que uma página da Web atualize apenas partes específicas sem interromper as atividades do usuário.\*/

||B|| - Incorreta. O XSLT é uma linguagem para transformar documentos XML, como HTML, texto ou XML.\*/

||C|| - Incorreta. SOAP é um protocolo baseado em XML para troca de mensagens, focado em chamadas de serviços de rede, mas não define como estruturar os diferentes tipos de interações de forma tão flexível quanto o WSDL.\*/

||D|| - Incorreta. REST é uma arquitetura de estilo para sistemas distribuídos que usa métodos HTTP para comunicação entre serviços, não focando em padrões de resposta como o WSDL.\*/

||E|| - Correta. A especificação WSDL permite que o sistema de rastreamento se comunique com os diferentes serviços de entrega, utilizando os padrões de troca de mensagens adequados, como "In-Out" para serviços que necessitam de resposta, "Out-Only" para serviços de envio sem retorno, e "In-Optional-Out" para serviços que podem ou não responder. Isso permite flexibilidade na definição de interações entre serviços *web*.\*/

**Questão 38**

```
public class Main {
    public static void main(String[] args)
        throws InterruptedException {
        ExecutorService executor =
            Executors.newThreadPerTaskExecutor(
                Thread.ofVirtual().factory());
        System.out.println("Executor created");
        for (int i = 0; i < 10; i++) {
            int taskNumber = i;
            executor.submit(() -> {
                System.out.println("Task " + taskNumber +
                    " executed by " + Thread.currentThread());
            });
        }
        executor.shutdown();
    }
}
```

A execução do trecho de código precedente, em Java 21, resultará na criação de

- A um *ExecutorService* que utiliza um *pool* fixo de *threads* virtuais.
- B *threads* que serão gerenciadas diretamente pelo sistema operacional.
- C *threads* virtuais que serão automaticamente descartadas após a execução da tarefa.
- D *threads* virtuais que ficarão em estado de execução, de forma contínua.
- E *threads* físicas compatíveis com o método `submit()` utilizado no código.

**||JUSTIFICATIVAS||**

||A|| - Incorreta. O código não está utilizando um *pool* de *threads* fixo. Ele está criando uma nova *thread* para cada tarefa, utilizando

`Executors.newThreadPerTaskExecutor(Thread.ofVirtual().factory()).` Portanto, não há um *pool* fixo de *threads*, e sim *threads* criadas por tarefa.\*/

||B|| - Incorreta. As *threads* virtuais não são gerenciadas diretamente pelo sistema operacional. Elas são gerenciadas pela JVM e não são associadas diretamente a *threads* físicas do sistema operacional. Esse é um dos principais aspectos das *threads* virtuais, que são leves, e a JVM pode otimizar sua execução de forma eficiente, criando muitas delas sem a sobrecarga das *threads* tradicionais do sistema operacional.\*/

||C|| - Correta. As *threads* virtuais são leves e a JVM pode descartá-las automaticamente após a execução da tarefa. Quando a execução de uma tarefa termina, a *thread* virtual é encerrada. O conceito de *threads* virtuais permite que muitas *threads* sejam criadas sem um grande custo, pois elas são automaticamente descartadas quando não são mais necessárias.\*/

||D|| - Incorreta. As *threads* virtuais não ficam em estado de execução de forma contínua. Elas são criadas para cada tarefa e, após a execução, são descartadas. O que acontece é que, enquanto a tarefa está em execução, a *thread* virtual é associada à tarefa e ficará ativa até o término da execução da tarefa.\*/

||E|| Incorreta. O método `submit()` pode ser usado tanto com *threads* físicas quanto virtuais, mas, no caso, o código está utilizando *threads* virtuais. *Threads* virtuais são uma abstração da JVM e podem ser usadas de maneira semelhante às *threads* tradicionais, mas sem a sobrecarga das *threads* físicas.\*/



**Questão 39**

Assinale a opção que apresenta a ferramenta que possibilita, nativamente, a execução paralela de testes de WebDriver em máquinas remotas, em múltiplas plataformas e em larga escala, com diferentes versões de navegadores.

- A JUnit
- B Ranorex
- C Selenium Grid
- D Cucumber
- E Cypress

**JUSTIFICATIVAS**

||A|| - Incorreta. JUnit é uma ferramenta para testes unitários em Java, mas não oferece a funcionalidade de execução paralela de testes em diferentes navegadores e plataformas. JUnit é voltada para testes de unidade, e não para automação de testes em múltiplos navegadores.\*/

||B|| - Incorreta. Ranorex é uma plataforma de automação de testes focada principalmente em testes de interface gráfica (GUI) em *desktop*, *web* e aplicativos móveis. Ele oferece uma solução mais centralizada e integrada para automação de testes, em vez de ser usado para gerenciamento distribuído de testes em múltiplas máquinas.\*/

||C|| - Correta. Selenium Grid é uma ferramenta que permite a execução de testes automatizados em paralelo em múltiplas máquinas e navegadores. Ele distribui comandos do cliente para instâncias remotas de navegadores, facilitando testes em várias versões de navegadores e plataformas diferentes, otimizando o tempo e a cobertura dos testes. Isso é ideal para testar aplicações em diferentes ambientes simultaneamente.\*/

||D|| - Incorreta. Embora seja uma ferramenta de teste de comportamento, Cucumber é voltado para testes baseados em Gherkin e não para a execução paralela de testes em diversas máquinas e navegadores. Cucumber é uma ferramenta que suporta o desenvolvimento orientado a comportamento (BDD), permitindo a execução de testes automatizados a partir de especificações executáveis.\*/

||E|| - Incorreta. Cypress é uma excelente ferramenta de automação de testes para aplicações *web*, mais focado na execução de testes em uma única máquina e em um único navegador de cada vez (principalmente no Chrome). Ele oferece uma abordagem mais simples e rápida para testes *end-to-end*, com uma excelente integração com o próprio navegador, permitindo que os testes interajam diretamente com a aplicação, mas sua execução não é distribuída, o que significa que não é ideal para rodar testes em diferentes navegadores ou em múltiplas plataformas de forma paralela.\*/

**Questão 40**

Julgue os itens a seguir, a respeito de testes de *software*.

- I A herança torna mais simples os testes de classes, pois os testes das operações herdadas contemplam os testes em todos os contextos de uso.
- II O teste de *release* deve ser realizado por uma equipe que não esteve envolvida no desenvolvimento do sistema.
- III Os testes de estresse ajudam a identificar o ponto de degradação de um sistema, permitindo implementar controles para evitar sobrecarga.

Assinale a opção correta.

- A Apenas o item I está certo.
- B Apenas o item II está certo.
- C Apenas os itens I e III estão certos.
- D Apenas os itens II e III estão certos.
- E Todos os itens estão certos.

**JUSTIFICATIVAS**

||A|| - Incorreta. O item I está errado, pois a herança torna os testes de classes mais complicados, pois as operações herdadas podem fazer suposições sobre outras operações e atributos, que podem não ser válidas em subclasses. Assim, é necessário testar as operações herdadas em todos os contextos de uso para garantir que funcionem corretamente em cada caso.\*/

||B|| - Incorreta. Os itens II e III estão certos.\*/

||C|| - Incorreta. O item I está errado.\*/

||D|| - Correta. O item II está certo, pois o objetivo principal do processo de teste de *release* é convencer o fornecedor do sistema de que esse sistema é bom o suficiente para uso. Portanto, uma equipe separada, que não esteve envolvida no desenvolvimento do sistema, deve ser responsável pelo teste de *release*. O item III está certo, pois os testes de estresse são particularmente relevantes para sistemas distribuídos baseados em uma rede de processadores. Esses sistemas frequentemente apresentam degradação severa quando estão muito carregados. Os testes de estresse ajudam a descobrir quando a degradação começa, e, assim, é possível adicionar controles ao sistema para rejeitar operações além desse ponto.\*/

||E|| - Incorreta. Apenas os itens II e III estão certos.\*/

**Questão 41**

```
begin
    integer arr[100], target, inicio, fim, meio;
    integer busca_binaria(integer arr, integer
target);
    begin
        inicio := 0;
        fim := length(arr) - 1;
        while inicio <= fim do
            meio := (inicio + fim) div 2;
            if arr[meio] = target then
                return meio;
            else if arr[meio] < target then
                inicio := meio + 1;
            else
                fim := meio - 1;
            end;
        return -1;
    end;
end;
```

O algoritmo de busca binária apresentado anteriormente possui

- A complexidade de tempo  $O(n)$ , em que  $n$  é o número de elementos no *array*.
- B complexidade de tempo  $O(\log n)$ , em que  $n$  representa o número de elementos no *array*.
- C complexidade espacial  $O(n)$ , já que o algoritmo não usa estruturas de dados adicionais que crescem com o tamanho da entrada.
- D complexidade espacial  $O(\log n)$ , já que o algoritmo não usa estruturas de dados adicionais que crescem com o tamanho da entrada.
- E complexidade de tempo  $O(1)$ , já que o algoritmo não usa estruturas de dados adicionais que crescem com o tamanho da entrada.

**||JUSTIFICATIVAS||**

||A|| - Incorreta. A complexidade de tempo  $O(n)$  significa que o tempo de execução do algoritmo aumenta linearmente com o número de elementos no *array*. Isso ocorre porque, no caso de uma busca linear, por exemplo, o algoritmo verifica cada elemento do *array* até encontrar o valor desejado ou até terminar a busca sem sucesso. Para  $n$  elementos, o algoritmo pode realizar até  $n$  comparações, o que resulta em uma complexidade de  $O(n)$ , o que é muito pior que nos algoritmos de busca binária.\*/\*

||B|| - Correta. A complexidade de tempo  $O(\log n)$  significa que o tempo necessário para executar o algoritmo aumenta de forma logarítmica à medida que o número de elementos no *array* aumenta. No caso da busca binária, a cada iteração, o número de elementos que o algoritmo precisa verificar é reduzido pela metade, o que resulta em uma busca muito eficiente em *arrays* grandes. Essa complexidade é muito melhor do que uma busca linear, que tem complexidade  $O(n)$ .\*/

||C|| - Incorreta. Para um algoritmo de complexidade espacial  $O(n)$ , é necessário que o algoritmo utilize estruturas de dados cujo tamanho aumente proporcionalmente ao número de elementos na entrada (por exemplo, armazenando cópias de dados ou utilizando *arrays*/vetores adicionais). Se o algoritmo não utiliza nenhuma estrutura adicional que cresça com o tamanho da entrada, a complexidade espacial não seria  $O(n)$ . A complexidade espacial em um caso como esse poderia ser  $O(1)$  (ou constante), pois a memória usada não depende do tamanho da entrada.\*/\*

||D|| - Incorreta.  $O(\log n)$  geralmente se aplica a algoritmos recursivos ou divide e conquista, onde a profundidade da pilha de chamadas pode crescer logaritmicamente. No entanto, se o algoritmo não utiliza nenhuma estrutura de dados adicional que cresça com a entrada, a complexidade espacial não seria  $O(\log n)$ . Se não há uso de estruturas que ocupem memória proporcional ao tamanho da entrada (como *arrays* ou listas) e o algoritmo não requer uma pilha de chamadas muito profunda, a complexidade espacial pode ser  $O(1)$ , ou seja, constante.\*/\*

||E|| - Incorreta. O uso de estruturas de dados adicionais não afeta diretamente a complexidade de tempo de um algoritmo. Por exemplo, se o algoritmo for linear, como a busca de um elemento em um vetor, a complexidade de tempo seria  $O(n)$ , e não  $O(1)$ .  $O(1)$  ocorre em algoritmos onde o tempo de execução é constante independentemente do tamanho da entrada, como acessar um elemento específico em uma posição predeterminada de um *array* ou realizar uma operação simples sem interação com todos os dados da entrada.\*/\*

**Questão 42**

Se, durante uma transferência bancária, ocorrer uma falha de energia exatamente no momento da operação, a propriedade que garantirá que a transação seja completamente concluída ou totalmente desfeita, evitando estados intermediários no banco de dados, é denominada

- A isolamento.
- B consistência.
- C atomicidade.
- D durabilidade.
- E completude.

**||JUSTIFICATIVAS||**

||A|| - Incorreta. O isolamento garante que as transações concorrentes não interfiram uma com as outras, fazendo com que cada transação seja executada isoladamente.\*/\*

||B|| - Incorreta. A consistência assegura que uma transação leve o banco de dados de um estado válido para outro estado válido, preservando as regras e restrições definidas.\*/\*

||C|| - Correta. A atomicidade garante que uma transação seja tratada como uma unidade indivisível (atômica) de trabalho. Ou todas as operações são realizadas com sucesso (*commit*), ou nenhuma delas é efetivada (*rollback*).\*/

||D|| - Incorreta. A durabilidade assegura que, uma vez que uma transação seja confirmada (*commit*), suas alterações serão permanentes e sobreviverão a falhas do sistema.\*/\*

||E|| - Incorreta. Não é uma propriedade do banco de dados.\*/\*

**Questão 43**

Assinale a opção que apresenta uma técnica de modelagem e otimização de bases de dados para *business intelligence*.

- A modelo estrela
- B OLTP
- C normalização extrema
- D tabelas planas
- E remoção de dados históricos

**||JUSTIFICATIVAS||**

||A|| - Correta. O modelo estrela é uma técnica amplamente utilizada em BI para modelagem de bases de dados analíticas. Ele organiza os dados em tabelas de fatos e dimensões, facilitando consultas OLAP e análise de dados.\*/\*

||B|| - Incorreta. O banco de dados relacional (OLTP) é modelado com tabelas interligadas onde o processo de consulta pode causar uma indisponibilidade no sistema de gestão empresarial (ERP), pois é baseado em transações.\*/\*

||C|| - Incorreta. A normalização é útil em bancos de dados transacionais, mas, em BI, a desnormalização é preferível para melhorar o desempenho das consultas analíticas, o que é típico em sistemas OLAP.\*/\*

||D|| - Incorreta. Tabelas planas (ou *flat tables*) são limitadas na modelagem de dados analíticos, pois não capturam relações complexas entre dados e podem prejudicar a escalabilidade de consultas em grandes volumes de dados.\*/\*

||E|| - Incorreta. A remoção de dados históricos contradiz a ideia central de BI, que depende de dados históricos para análise de tendências e para suportar decisões informadas.\*/\*

**Questão 44**

Em relação ao modelo de referência OSI da ISO, bem como às suas camadas e respectivas funções, julgue os itens a seguir.

- I O modelo de referência OSI possui sete camadas.
- II O HTTP está presente na camada de rede.
- III O tratamento da transmissão de *bits* normais por um canal de comunicação é função da camada física.
- IV A camada de sessão gerencia estruturas de dados abstratas e permite a comunicação entre computadores com diferentes representações de dados.

Estão certos apenas os itens

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

**||JUSTIFICATIVAS||**

||A|| - Correta. O item I está certo, pois o modelo de referência OSI tem sete camadas.

O item II está errado, pois o HTTP está presente na camada de aplicação do modelo OSI.

O item III está certo, pois a camada física trata da transmissão de *bits* normais por um canal de comunicação.

O item IV está errado, porque a camada de sessão permite que os usuários, em diferentes máquinas, estabeleçam sessões de comunicação entre eles. \*/

||B|| - Incorreta. Apenas os itens I e III estão certos.\*/

||C|| - Incorreta. Apenas os itens I e III estão certos.\*/

||D|| - Incorreta. Apenas os itens I e III estão certos.\*/

||E|| - Incorreta. Apenas os itens I e III estão certos.\*/

**Questão 45**

Para mapear nomes de *hosts* em endereços IP na rede interna, onde as mensagens de consulta e resposta são enviadas como pacotes UDP, é adequado o uso do

A HTTP.

B FTP.

C DHCP.

D DNS.

E SMTP.

**||JUSTIFICATIVAS||**

||A|| - Incorreta. O protocolo HTTP é usado para transportar toda essa informação entre os servidores e os clientes na Web. \*/

||B|| - Incorreta. O protocolo FTP tem o objetivo de transferir programas e arquivos de todos os tipos por meio de redes TCP/IP. Suas principais características são a confiabilidade e a eficiência com que realizam as transferências. Embora seja possível usar outros protocolos para transferir arquivos, como o HTTP e o SMTP, o FTP é o mais adequado para essa tarefa, pois utiliza o protocolo TCP na camada de transporte e escuta as requisições nas portas 20 (dados) e 21 (controle).\*/

||C|| - Incorreta. O protocolo DHCP é responsável pela configuração dinâmica de endereços IP em uma rede de computadores.\*/

||D|| - Correta. O DNS é utilizado para mapear nomes de *hosts* em endereços IP. Para mapear um nome em um endereço IP, um programa aplicativo chama um procedimento de biblioteca denominado resolvidor e repassa a ele o nome como um parâmetro. O resolvidor envia uma consulta contendo o nome para um servidor DNS local, que procura o nome e retorna uma resposta contendo o endereço IP ao resolvidor. Este, em seguida, retorna o endereço IP ao programa aplicativo que fez a chamada. As mensagens de consulta e resposta são enviadas como pacotes UDP. Munido do endereço IP, o programa pode então estabelecer uma conexão TCP com o host ou enviar pacotes UDP até ele. \*/

||E|| - Incorreta. O SMTP é utilizado para enviar *emails* pelas conexões e informa de volta o *status* de entrega e quaisquer erros.\*/

**Questão 46**

A respeito de instalação, configuração e suporte em ambiente Linux, julgue os itens a seguir.

I O comando `# apt-get remove --purge` pode ser usado para a remoção completa de um pacote e de seus arquivos de configuração.

II O uso do comando `# chkconfig dhcpd on` permite iniciar o serviço de DHCP no servidor Linux.

III Em algumas distribuições Linux, para a inicialização do servidor NFS, é preciso ativar o serviço `portmap`.

Assinale a opção correta.

A Apenas o item I está certo.

B Apenas o item II está certo.

C Apenas os itens I e III estão certos.

D Apenas os itens II e III estão certos.

E Todos os itens estão certos.

**||JUSTIFICATIVAS||**

||A|| - Incorreta. O item III também está certo.\*/

||B|| - Incorreta. Apenas os itens I e III estão certos.\*/

||C|| - Correta. O item I está certo porque, para remover um pacote, é usado o comando `apt-get remove`, como em `# apt-get remove apache2`. Ao remover o pacote, todos os componentes são apagados, inclusive as bibliotecas e arquivos diversos. Apenas os arquivos de configuração são preservados, para o caso de usuário querer instalar o pacote novamente depois. Para fazer uma remoção completa, incluindo os arquivos de configuração, deve-se adicionar o parâmetro `--purge`, como a seguir.

`# apt-get remove --purge`

O item II está errado. O item II está errado. Embora o pacote se chame apenas DHCP, o *script* referente ao serviço é `dhcpd`, de forma que os comandos para iniciar e parar o serviço são os seguintes.

`# service dhcpd start`

`# service dhcpd stop`

O item III está certo porque NFS é uma opção para compartilhar sistemas de arquivos entre máquinas Linux, de uma forma prática e estável. O NFS é um servidor que precisa ser habilitado manualmente na maior parte das distribuições. Utiliza outro serviço, o *portmap*, para gerenciar as requisições dos clientes. Este serviço deve estar ativo para que o NFS funcione, ou seja, para inicializar o servidor NFS, é preciso ativar os dois. Nas distribuições derivadas do Debian, há ainda um terceiro serviço, o *nfs-common*, que deve ser ativado antes do módulo servidor, como a seguir.

`# /etc/init.d/portmap start`

`# /etc/init.d/nfs-common start`

`# /etc/init.d/nfs-kernel-server start`

Nas distribuições derivadas do Red Hat, devem-se ativar apenas os serviços `portmap` e `nfs`, conforme a seguir.

`# service portmap start`

`# service nfs start`

A configuração do NFS é feita em um único arquivo, o `/etc/exports`, onde se encontra a configuração dos diretórios compartilhados, um por linha. Originalmente, esse arquivo fica vazio ou contém apenas um comentário. É preciso apenas abri-lo em um editor de textos e adicionar as pastas que deseja compartilhar.\*/

||D|| - Incorreta. Apenas os itens I e III estão certos.\*/

||E|| - Incorreta. Apenas os itens I e III estão certos.\*/



### Questão 47

Acerca da instalação e configuração do DHCP e do DNS no ambiente Windows Server, julgue os itens subsequentes.

- I O comando `Install-WindowsFeature DHCP -IncludeManagementTools` autoriza o servidor DHCP no *Active Directory*.
- II O comando `Get-DhcpServerInDC` verifica se o DHCP está autorizado no *Active Directory*.
- III A existência de um IP estático no servidor é pré-requisito para instalação e configuração do DNS.
- IV O DNSSEC é um pacote de extensão que oferece segurança contra ataques de falsificação de DNS.

Estão certos apenas os itens

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

#### JUSTIFICATIVAS

||A|| - Incorreta. O item I está errado, pois o comando `Add-DhcpServerInDC -DnsName` é executado para adicionar o servidor DHCP à lista de servidores DHCP autorizados no *Active Directory*.\*/

||B|| - Incorreta. O item I está errado.\*/

||C|| - Incorreta. Os itens II, III e IV estão certos.\*/

||D|| - Incorreta. O item I está errado.\*/

||E|| - Correta. O item I está errado, pois o comando `Add-DhcpServerInDC -DnsName` é executado para adicionar o servidor DHCP à lista de servidores DHCP autorizados no *Active Directory*.

O item II está certo porque o comando `Get-DhcpServerInDC` é executado para verificar se o servidor DHCP está autorizado no *Active Directory*.

O item III está certo, porque, antes de se instalar e se configurar o servidor DNS, o computador deve atender aos seguintes pré-requisitos: estar executando uma versão com suporte do Windows Server; ter um IP estático; ter uma conta que seja membro do grupo administradores ou equivalente.

O item IV está certo, pois DNSSEC é um pacote de extensões que adiciona segurança ao protocolo DNS, habilitando respostas de DNS para validação. A DNSSEC oferece autoridade de origem, integridade dos dados e negação de existência autenticada. Com o DNSSEC, o protocolo DNS fica muito menos suscetível a determinados tipos de ataques, particularmente os ataques de falsificação de DNS.\*/

### Questão 48

Se um servidor *Active Directory* está com problema de replicação com outro servidor *Active Directory*, então, para resolver o problema de conexão, a porta a ser liberada no *firewall* é a

- A TCP 135.
- B UDP 389.
- C TCP 21.
- D UDP 67.
- E TCP 443.

#### JUSTIFICATIVAS

||A|| - Correta. A porta correta é a TCP 135, onde o serviço de RPCs é executado. Por padrão, as RPCs (chamadas de procedimento remoto) de replicação do *Active Directory* ocorrem dinamicamente em uma porta disponível por meio do RPCSS (mapeador de pontos de extremidade da RPC) na porta 135. Deve-se verificar se o *firewall* do Windows com segurança avançada e outros *firewalls* estão configurados corretamente para permitir a replicação.\*/

||B|| - Incorreta. Servidor LDAP. Protocolo UDP. Porta 389.\*/

||C|| - Incorreta. O serviço de publicação FTP fornece conectividade FTP. Por padrão, a porta de controle FTP é a 21.\*/

||D|| - Incorreta. Servidor DHCP. Protocolo UDP. Porta 67.\*/

||E|| - Incorreta. O serviço do sistema HTTP SSL permite que o IIS execute funções SSL. O SSL é um padrão aberto para estabelecer um canal de comunicação criptografado, a fim de evitar a interceptação de informações extremamente importantes, por exemplo, números de cartão de crédito. HTTPS. Protocolo TCP. Porta 443.\*/

### Questão 49

No que se refere ao escopo do projeto, julgue os itens que se seguem.

- I Após a aprovação do termo de abertura do projeto, faz-se necessário coletar os requisitos dos principais *stakeholders* e estruturar o escopo do projeto.
- II A matriz de rastreabilidade dos requisitos contribui para o monitoramento dos requisitos e para o gerenciamento das mudanças no escopo do produto.
- III A declaração de escopo de um projeto inclui apenas informações como programação, orçamento, entregas e responsabilidades.

Assinale a opção correta.

- A Apenas o item I está certo.
- B Apenas o item III está certo.
- C Apenas os itens I e II estão certos.
- D Apenas os itens II e III estão certos.
- E Todos os itens estão certos.

#### JUSTIFICATIVAS

||A|| - Incorreta. Apenas os itens I e II estão certos.\*/

||B|| - Incorreta. Apenas os itens I e II estão certos.\*/

||C|| - Correta. O item I está certo, porque desenvolver o termo de abertura do projeto (*project charter*) é o primeiro processo, pois consiste na elaboração desse documento, que autoriza formalmente a existência de um projeto, isto é, autoriza o gerente de projetos a aplicar os recursos da organização no desenvolvimento do projeto.

O item II está certo, pois a matriz de rastreabilidade dos requisitos ajuda a monitorar os requisitos de produto, ao longo do ciclo de vida, gerenciando as mudanças no escopo do produto. Além disso, essa matriz ajuda no alinhamento dos requisitos aos objetivos do projeto e do negócio.

O item III está errado. A declaração de escopo de um projeto deve incluir, essencialmente: justificativa de sua criação (essa informação servirá para que o gerente de projetos e sua equipe possam apresentar as necessidades de negócio que motivaram a criação do projeto. No âmbito de uma empresa, ajuda o gerente de projetos a negociar recursos para o projeto.); objetivos do projeto (uma vez estabelecidas as **justificativas**, é preciso deixar claro as intenções do projeto, ou seja, quais suas metas quantitativas, quais os resultados esperados. Normalmente, projetos com objetivos bem declarados são aqueles que mostram explicitamente os alvos de prazo, custo e qualidade esperados. Quanto aos objetivos mais qualitativos, estes carregam um grau maior de risco em relação ao seu atendimento.); produto do

projeto (aquilo que se espera entregar no final para que o projeto tenha sucesso. Essa informação deverá conter uma descrição sucinta do produto do projeto. Em muitos casos, essa informação poderá envolver desenhos, fluxogramas, esboços, detalhes de construção etc. Ela pode ser complementada com a adição de uma lista de subprodutos, que, quando integrados, devem compor o produto final do projeto. Devem ser incluídos, também, os critérios de aceitação do produto.).\*/

||D|| - Incorreta. Apenas os itens I e II estão certos.\*/

||E|| - Incorreta. Apenas os itens I e II estão certos.\*/

### Questão 50

Julgue os itens subsequentes, a respeito de segurança física e lógica.

- I A organização deve prevenir o acesso físico não autorizado às instalações de processamento críticas e sensíveis, mantendo-as em áreas seguras e protegidas por perímetro de segurança definidos.
- II Os equipamentos para contingência e mídia de *backup* devem ficar a uma distância segura, para que não sejam danificados em eventual desastre que afete o local principal.
- III O cabeamento de telecomunicação que transporta dados deve ser protegido contra interceptação ou danos, mediante o uso de conduítes ou evitando-se trajetos que passem por áreas públicas.

Assinale a opção correta.

- A Apenas o item I está certo.
- B Apenas o item III está certo.
- C Apenas os itens I e II estão certos.
- D Apenas os itens II e III estão certos.
- E Todos os itens estão certos.

#### ||JUSTIFICATIVAS||

||A|| - Incorreta. Todos os itens estão certos.\*/

||B|| - Incorreta. Todos os itens estão certos.\*/

||C|| - Incorreta. Todos os itens estão certos.\*/

||D|| - Incorreta. Todos os itens estão certos.\*/

||E|| - Correta. O item I está certo, pois a organização deve prevenir o acesso físico não autorizado, danos e interferências em suas instalações e informações. As instalações de processamento da informação críticas ou sensíveis precisam ser mantidas em áreas seguras, protegidas por perímetros de segurança definidos, com barreiras de segurança e controles de acesso apropriados, de maneira compatível com os riscos identificados.

O item II está certo, pois os equipamentos para contingência e mídia de *backup* devem ficar a uma distância segura, para que não sejam danificados por um desastre que afete o local principal.

O item III está certo, pois o cabeamento de telecomunicações que transporta dados ou dá suporte aos serviços de informações deve ser protegido contra interceptação ou danos, pelo uso de conduítes ou evitando-se trajetos que passem por áreas públicas. Para sistemas sensíveis ou críticos, deve se considerar o uso de rotas alternativas e(ou) meios de transmissão alternativos que proporcionem segurança adequada, além do acesso controlado aos painéis de controle.\*/

### Questão 51

Assinale a opção em que é citado um sistema capaz de verificar o conteúdo das camadas superiores do TCP/IP e alertar sobre atividades não apropriadas ou nocivas na rede.

- A SSH
- B *firewall*
- C IDS
- D VPN
- E *proxy* reverso

#### ||JUSTIFICATIVAS||

||A|| - Incorreta. SSH é um protocolo de rede da camada de aplicação (camada 4 do TCP/IP) que permite a conexão com outro computador na rede, de forma a executar comandos de uma unidade remota. \*/

||B|| - Incorreta. *Firewall* é um dispositivo de uma rede de computadores que tem como objetivos: regular o tráfego de dados entre uma rede local e a rede externa não confiável, por meio da introdução de filtros para pacotes ou aplicações; impedir a transmissão e(ou) recepção de acessos nocivos ou não autorizados dentro de uma rede local. \*/

||C|| - Correta. Sistemas de detecção de intrusão (IDS) são sistemas que fazem a análise das atividades de uma rede com objetivo de descobrir e alertar os seus administradores sobre atividades não apropriadas, nocivas ou anômalas. Eles são capazes de verificar o conteúdo das camadas superiores do TCP/IP a procura de padrões (assinaturas) que indiquem uma tentativa de ataque.\*/

||D|| - Incorreta. Uma rede privada virtual (*virtual private network* — VPN) é uma rede de comunicações privada normalmente utilizada por uma ou mais empresas e(ou) instituições, que tem como suporte uma rede de comunicações pública (como por exemplo, a Internet). \*/

||E|| - Incorreta. O *proxy* reverso é um *proxy* que funciona como intermediário entre o servidor e a rede externa. A grande diferença entre o *proxy* normal e o *proxy* reverso é que o primeiro se localiza em uma rede local, e o segundo, no lado do servidor. Dessa forma, as requisições são encaminhadas ao *proxy* reverso, que as repassa ao servidor de aplicação.\*/

### Questão 52

Assinale a opção que apresenta o tipo de ataque no qual o atacante, fazendo uso de um *bot-net*, envia vários pacotes ou requisições de uma vez, com o objetivo de sobrecarregar um servidor e, como consequência, impedir o fornecimento do serviço para os demais usuários.

- A DDoS
- B *spoofing*
- C XSS
- D SQL injection
- E *path traversal*

#### ||JUSTIFICATIVAS||

||A|| - Correta. O ataque de negação de serviço (*denial of service*) acontece quando um atacante envia vários pacotes ou requisições de serviço de uma vez, com objetivo de sobrecarregar um servidor e, como consequência, impedir o fornecimento de um serviço para os demais usuários, causando prejuízos. Em dispositivos com grande capacidade de processamento, normalmente, é necessária uma enorme quantidade de requisições para que o ataque seja eficaz. Para isso, o atacante faz o uso de uma *bot-net* (rede de computadores zumbis sob comando do atacante) para bombardear o servidor com requisições, fazendo com que o ataque seja feito de forma distribuída.\*/

||B|| - Incorreta. *Spoofing* é a modificação de campos de identificação de pacotes de forma que o atacante possa atuar se passando por outro *host*. As alterações no campo de identificação podem ser úteis para: o atacante se disfarçar como se fosse outro usuário da rede; envenenar caches de protocolos, para inserir informações falsificadas para diversos fins (*cache poisoning*).\*/

||C|| - Incorreta. *Session hijacking* (sequestro de sessão) é um ataque que permite que um atacante obtenha acesso às informações, se passando por outro *host* em uma rede. O atacante obtém informações de uma sessão aberta pela vítima, e, a partir dessas informações, obtém informações de autenticação que fazem com que o atacante consiga se passar pelo *host* atacado, fazendo operações em nome dele.\*/



||D|| - Incorreta. *SQL injection* é uma técnica de ataque que explora vulnerabilidades de segurança que ocorrem na camada de banco de dados de uma aplicação, que ocorrem quando a entrada do usuário ou está incorretamente filtrada, ou não é rígida o suficiente. \*/

||E|| - Incorreta. Um ataque de travessia de caminho (também conhecido como travessia de diretório) visa acessar arquivos e diretórios que são armazenados fora da pasta raiz da Web. Ao manipular variáveis que referenciam arquivos com sequências ponto-ponto-barras (../) e suas variações, ou usando caminhos de arquivo absolutos, pode ser possível acessar arquivos e diretórios arbitrários armazenados no sistema de arquivos, incluindo código-fonte do aplicativo ou configuração e arquivos críticos do sistema. \*/

### Questão 53

Julgue os seguintes itens, referentes à computação em nuvem.

- I A computação em nuvem permite mudar a forma de operar a tecnologia da informação: em vez de um modelo baseado em aquisição de equipamentos, adota-se um modelo baseado em aquisição de serviços de TI.
- II A elasticidade em nuvem se aplica quando há percepção da existência ilimitada dos recursos que podem ser adquiridos no momento e na quantidade desejada, com ampliação ou diminuição rápida, na medida da solicitação.
- III Um dos benefícios da adoção da computação em nuvem é a segurança das informações sensíveis, garantindo-se a integridade e privacidade dos dados.

Assinale a opção correta.

- A Apenas o item II está certo.
- B Apenas o item III está certo.
- C Apenas os itens I e II estão certos.
- D Apenas os itens I e III estão certos.
- E Todos os itens estão certos.

#### ||JUSTIFICATIVAS||

||A|| - Incorreta. Apenas os itens I e II estão certos. \*/

||B|| - Incorreta. Apenas os itens I e II estão certos. \*/

||C|| - Correta. O item I está certo, pois a computação em nuvem significa mudar fundamentalmente a forma de operar a TI, saindo de um modelo baseado em aquisição de equipamentos para um modelo baseado em aquisição de serviços de TI. Trata-se de um conjunto de recursos virtuais facilmente utilizáveis e acessíveis, tais como *hardware*, *software*, plataformas de desenvolvimento e serviços. Esses recursos podem ser dinamicamente reconfigurados para se ajustarem a uma carga de trabalho (*workload*) variável, permitindo a otimização do uso dos recursos.

O item II está certo, porque o benefício da elasticidade da computação em nuvem permite transferir o risco da baixa utilização e da alta utilização (saturação) para uma situação de ajuste fino entre a carga de trabalho e os recursos disponíveis. Trata-se de uma condição em que há percepção da existência ilimitada dos recursos que podem ser adquiridos no momento e na quantidade desejada, com ampliação ou diminuição rápida, na medida da solicitação.

O item III está errado, pois, apesar dos vários benefícios da computação em nuvem, é preciso considerar as dificuldades para sua utilização que se tornam barreiras para adesão. A segurança é um grande desafio da computação em nuvem. A segurança e a privacidade dos dados estão no topo das preocupações relacionadas a TI. Isso é confirmado pela maioria das pesquisas realizadas no segmento. Tal receio do mercado se deve pelo fato de a computação em nuvem colocar os riscos em outro nível, porque os serviços essenciais são, muitas vezes, subcontratados de terceiros. Nessa condição, pode ser mais difícil manter a integridade e a privacidade dos dados. \*/

||D|| - Incorreta. Apenas os itens I e II estão certos. \*/

||E|| - Incorreta. Apenas os itens I e II estão certos. \*/

### Questão 54

Com relação à criptografia, julgue os itens subsequentes.

- I O princípio básico da disponibilidade só é implementado com uso da criptografia.
- II O tipo de criptografia baseada em sigilo da chave se divide em sistemas criptográficos simétricos (de chave privada) e sistemas criptográficos assimétricos (de chave pública).
- III A principal vantagem do algoritmo de chave pública, em comparação ao algoritmo de chave privada, é a maior velocidade no processo criptográfico, em razão do menor tamanho das chaves.

Assinale a opção correta.

- A Apenas o item I está certo.
- B Apenas o item II está certo.
- C Apenas os itens I e III estão certos.
- D Apenas os itens II e III estão certos.
- E Todos os itens estão certos.

#### ||JUSTIFICATIVAS||

||A|| - Incorreta. Apenas o item II está certo. \*/

||B|| - Correta. O item I está errado, porque os princípios de confidencialidade, integridade, autenticidade e irretratabilidade são, de fato, implementados pela criptografia. Entretanto, a disponibilidade não depende apenas da criptografia para funcionar, pois também pode depender da implantação de outras técnicas, como redundância e de planos de continuidade de negócios.

O item II está certo, uma vez que praticamente todos os algoritmos de criptografia usados atualmente se baseiam no sigilo da chave, pois permite que um algoritmo possa ser divulgado sem comprometer o sistema de segurança. Além disso, a segurança do algoritmo pode ser devidamente testada e confirmada por meio da divulgação. Esses sistemas criptográficos baseados em sigilo da chave são divididos em dois tipos: sistemas criptográficos simétricos (chave privada) e sistemas criptográficos assimétricos (chave pública).

O item III está errado. Os algoritmos de chave pública utilizam pares de chaves, uma pública e outra privada, atribuídas a uma pessoa ou entidade. A chave pública é distribuída livremente para todos os correspondentes, enquanto a chave privada deve ser conhecida apenas pelo seu dono. Num algoritmo de criptografia assimétrica, uma mensagem cifrada com a chave pública pode somente ser decifrada pela sua chave privada correspondente. Além disso, uma mensagem cifrada com a chave privada só pode ser decifrada pela chave pública correspondente. Com isso, é possível fazer assinaturas digitais sobre as mensagens, de forma que uma pessoa possa criptografar a mensagem com sua chave privada e o destinatário possa comprovar a autenticidade por meio da decifração pela chave pública do remetente. Dessa forma, os algoritmos de chave pública fornecem, além da confidencialidade, a garantia de autenticidade da origem. Vantagens: é mais segura que a criptografia simétrica, pois não precisa de meio seguro para enviar a chave para descriptar a mensagem; permite o uso de assinaturas digitais, garantindo a autenticidade da origem; a montagem de uma infraestrutura de chaves utiliza menos informação, visto que basta apenas um par de chaves por pessoa para se fazer a comunicação. Desvantagem: o processo de criptografia é mais lento e o tamanho das chaves é consideravelmente maior do que os algoritmos de chave privada. \*/

||C|| - Incorreta. Apenas o item II está certo. \*/

||D|| - Incorreta. Apenas o item II está certo. \*/

||E|| - Incorreta. Apenas o item II está certo. \*/

### Questão 55

A Lei Geral de Proteção de Dados Pessoais (LGPD)

- I tem como fundamento a liberdade de expressão, de informação, de comunicação e de opinião.
- II aplica-se ao tratamento de dados pessoais realizado para fins exclusivos de defesa nacional.
- III aplica-se ao tratamento de dados pessoais coletados fora do território nacional.
- IV tem como fundamento o desenvolvimento econômico e tecnológico e a inovação.

Estão certos apenas os itens

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

**||JUSTIFICATIVAS||**

||A|| - Incorreta. O item II está errado, pois contradiz o disposto no art. 4.º, III, b, da LGPD.\*/

||B|| - Correta. Os itens I e IV estão certos, conforme o art. 2.º, III e V, da LGPD.\*/

||C|| - Incorreta. O item II está errado, pois contradiz o disposto no art. 4.º, III, b, da LGPD. O item III está errado porque viola o art. 3.º, III, da LGPD.\*/

||D|| - Incorreta. O item III está errado.\*/

||E|| - Incorreta. Os itens II e III estão errados.\*/

### Questão 56

Em relação às árvores de decisão, julgue os itens a seguir.

- I Essa categoria de algoritmos da análise supervisionada realiza a classificação dos dados em um número limitado de classes, de modo a utilizar apenas os valores das variáveis de entrada como base.
- II As árvores de decisão são mais apropriadas para dados categóricos e intervalares, portanto, o uso de variáveis contínuas em árvores de decisão requer a discretização, que consiste na transformação de variáveis numéricas contínuas em intervalos ou categorias.
- III O algoritmo Random Forest utiliza o método bagging para a criação de múltiplas árvores de decisão independentes e combina as previsões para melhorar a precisão e reduzir o risco de overfitting.

Assinale a opção correta.

- A Apenas o item I está certo.
- B Apenas o item III está certo.
- C Apenas os itens I e II estão certos.
- D Apenas os itens II e III estão certos.
- E Todos os itens estão certos.

**||JUSTIFICATIVAS||**

||A|| - Incorreta. Todos os itens estão certos.\*/

||B|| - Incorreta. Os itens II e III também estão certos.\*/

||C|| - Incorreta. Os itens I e III também estão certos.\*/

||D|| - Incorreta. Os itens I e II também estão certos.\*/

||E|| - Correta. O item I está certo, pois o objetivo principal dos algoritmos de análise supervisionada de classificação é prever a classe ou a categoria a que uma amostra pertence com base em variáveis de entrada. Esses algoritmos aprendem um mapeamento entre os valores das variáveis de entrada e as classes previamente conhecidas durante o treino, de modo a utilizar esse mapeamento para a classificação de novos dados em um conjunto finito de categorias.

O item II está certo, pois as árvores de decisão dividem os dados em ramos com base em condições lógicas que são mais naturalmente adequadas para variáveis categóricas e intervalares.

Variáveis contínuas, que assumem valores dentro de um intervalo numérico infinito, não são diretamente compatíveis com essa estrutura hierárquica. Por isso, é necessário aplicar a discretização, um processo que transforma variáveis contínuas em categorias ou intervalos.

O item III está certo, pois o algoritmo *Random Forest* utiliza o método *bagging* para criar múltiplas árvores de decisão independentes a partir de amostras aleatórias dos dados. Cada árvore é treinada com um subconjunto de variáveis, o que aumenta a diversidade entre elas. Ao combinar as previsões, o *Random Forest* reduz o risco de *overfitting* e melhora a precisão, o que torna o modelo mais robusto.\*/

### Questão 57

Certa companhia de abastecimento de água organiza as informações da empresa em três conjuntos de dados: *clientes* (*idCliente*, *nomeCliente*), com dados dos clientes; *imoveis* (*idImovel*, *idCliente*, *endereço*), com dados de imóveis; e *contas* (*idConta*, *idImovel*, *consumo*), com dados do histórico de pagamento de contas e do consumo, em metros cúbicos, de cada conta. Essa organização leva em consideração que cada cliente pode ter vários imóveis (chave estrangeira: *idCliente* em *imoveis*) e que cada imóvel pode ter várias contas (chave estrangeira: *idImovel* em *contas*). Os conjuntos de dados (*clientes*, *imoveis* e *contas*) foram carregados para um *DataFrame* no *Pandas*.

Com base nas informações da situação hipotética precedente, assinale a opção que corresponde ao trecho de código que, caso seja executado, exibirá corretamente o total de consumo por cliente na variável *totalConsumo*.

- A `totalConsumo = contas.groupby('idCliente')['consumo'].sum()`
- B `imoveisContas = pd.merge(imoveis, contas, on='idImovel')`  
`totalConsumo = imoveisContas.groupby('idCliente')['consumo'].sum()`
- C `totalConsumo = clientes.merge(contas, on = 'idCliente').groupby('idCliente')['consumo'].sum()`
- D `totalConsumo = contas.groupby('idImovel')['consumo'].sum()`
- E `imoveisContas = pd.merge(contas, imoveis, on='idCliente')`  
`totalConsumo = imoveisContas.groupby('idImovel')['consumo'].sum()`

**||JUSTIFICATIVAS||**

||A|| - Incorreta. O *DataFrame* *contas* não possui a coluna *idCliente*. Ele está relacionado aos clientes apenas por meio do *DataFrame* *imoveis*, por meio da chave *idImovel*. Por isso, não é possível agrupar diretamente por *idCliente*.\*/

||B|| - Correta. O trecho código realiza a junção entre os *DataFrames* *imoveis* e *contas* para combinar as informações e, em seguida, agrupa os dados de cada cliente (*idCliente*) para calcular o total de consumo (*consumo*).\*/

||C|| - Incorreta. Essa tentativa de junção é falha porque o *DataFrame* *clientes* não está diretamente relacionado ao *DataFrame* *contas*. A relação entre os dois passa pelo *DataFrame* *imoveis*, por meio da chave *idImovel*.\*/

||D|| - Incorreta. Esse código calcula o total de consumo por imóvel (*idImovel*), mas não agrupa os valores por cliente (*idCliente*), de modo que ignora a relação entre imóveis e clientes.\*/

||E|| - Incorreta. A junção entre *contas* e *imoveis* a partir da utilização de *idCliente* está incorreto, porque não há a coluna *idCliente* em *contas*. Além disso, o agrupamento final é feito por

idImovel, o que não atende ao objetivo solicitado de calcular o consumo por cliente.\*/

### ❏ Questão 58

Durante uma auditoria em certa empresa de manutenção de hidrômetros, verificou-se que existem dados duplicados na entidade `clientes` (`cpf`, `nome_cliente`, `endereço`) do banco de dados, ou seja, o mesmo cliente aparece mais de uma vez com o mesmo `cpf` e `nome_cliente`, mas com endereços diferentes. Os dados do banco de dados foram carregados em um `DataFrame` `clientes` do `Pandas`.

A partir dessa situação hipotética, assinale a opção que corresponde ao trecho de código que, caso executado, identificará os clientes duplicados ao utilizar como filtro apenas o `cpf` e o `nome_cliente`.

- A `clientes[clientes.duplicated()]`
- B `clientes[clientes.duplicated(subset = ['cpf', 'nome_cliente'])]`
- C `clientes[clientes.duplicated(subset = ['nome_cliente'], keep='last')]`
- D `clientes.drop_duplicates(subset = ['cpf', 'nome_cliente'], keep = 'first')`
- E `clientes.drop_duplicates()`

#### ||JUSTIFICATIVAS||

||A|| - Incorreta. Esse código identifica as duplicatas em todas as colunas do `DataFrame`, mas não especifica quais colunas devem ser verificadas. O contexto da questão considera apenas as colunas relevantes (`cpf` e `nome_cliente`).\*/

||B|| - Correta. O método `duplicated()` identifica as duplicatas e, com o parâmetro `subset=['cpf', 'nome_cliente']`, avalia as duplicatas com base nas colunas `cpf` e `nome_cliente`.\*/

||C|| - Incorreta. Esse código verifica duplicatas com base apenas na coluna `nome_cliente`, sem levar em consideração o campo `cpf`. Assim, podem ser apresentados falsos positivos, como situações de dois clientes diferentes com o mesmo nome.\*/

||D|| - Incorreta. Esse código remove duplicatas com base nas colunas `cpf` e `nome_cliente` e mantém apenas a primeira ocorrência, o que pode ser útil para corrigir os dados. Porém, a questão pede o trecho de código que identifica os registros duplicados, e não o trecho de código que os remove.\*/

||E|| - Incorreta. Este código remove duplicatas considerando todas as colunas do `DataFrame`, mas não especifica as colunas relevantes. Ele não atende ao objetivo da questão, que é identificar duplicatas com base em `cpf` e `nome_cliente`.\*/

### ❏ Questão 59

A respeito do aprendizado supervisionado, assinale a opção correta.

- A Os algoritmos de regressão logística garantem que o modelo seja treinado e testado exclusivamente com o mesmo conjunto de dados para minimizar a variância do modelo.
- B O algoritmo de regressão linear é mais adequado para prever variáveis categóricas do que o Naive Bayes.
- C Na validação cruzada, o conjunto de dados é particionado em conjunto de treino, validação e teste, para que se avalie o desempenho do modelo de maneira confiável.
- D A máquina de suporte de vetores é um algoritmo probabilístico que sempre fornece a probabilidade de uma classe específica.
- E O algoritmo *k-nearest neighbors* utiliza a média ponderada dos coeficientes para estimar as relações lineares entre as variáveis.

#### ||JUSTIFICATIVAS||

||A|| - Incorreta. O treinamento supervisionado não pode utilizar o mesmo conjunto de dados para treinamento e validação. A validação cruzada, por exemplo, divide os dados em subconjuntos, de forma que um subconjunto é usado para teste enquanto os outros são usados para treino. Isso permite a

avaliação da capacidade de generalização do modelo e reduz a variância da avaliação, sem violar a separação entre treino e teste.\*/

||B|| - Incorreta. O algoritmo de regressão linear é mais adequado para prever variáveis contínuas, enquanto o Naive Bayes é mais apropriado para variáveis categóricas (classificação).\*/

||C|| - Correta. A validação cruzada é uma técnica que divide os dados em conjuntos de treino, validação e teste para garantir que o desempenho do modelo seja avaliado de maneira robusta. Essa prática é amplamente usada para prevenir problemas — como *overfitting* — e para fornecer uma avaliação mais confiável do modelo em outro conjunto de dados.\*/

||D|| - Incorreta. A máquina de suporte de vetores não é um algoritmo probabilístico. Ela busca encontrar o melhor hiperplano que separa as classes e não fornece diretamente a probabilidade de uma classe específica.\*/

||E|| - Incorreta. O algoritmo *k-nearest neighbors* não utiliza coeficientes para estimar relações entre variáveis. Ele classifica ou prevê valores com base na proximidade dos exemplos mais próximos. A média ponderada não é parte do funcionamento desse algoritmo.\*/

### ❏ Questão 60

No que se refere ao aprendizado não supervisionado, julgue os próximos itens.

- I A análise de componentes principais é uma técnica de redução de dimensionalidade, cujo objetivo é a preservação da maior parte da variância dos dados por meio da redução de redundâncias e da simplificação da análise, sem que haja perda de informações importantes.
- II O algoritmo *k-means* utiliza dendrogramas para formar grupos de dados em que os pontos são atribuídos ao *cluster* mais próximo, determinado por centros calculados iterativamente, de modo a minimizar a soma das distâncias quadráticas.
- III Scikit-learn é uma biblioteca de aprendizado não supervisionado que permite o agrupamento de dados semelhantes, com técnicas como *k-means* e clusterização hierárquica, modelagem de dados complexos com distribuições gaussianas e redução da dimensionalidade.

Assinale a opção correta.

- A Nenhum item está certo.
- B Apenas o item II está certo.
- C Apenas o item III está certo.
- D Apenas os itens I e II estão certos.
- E Apenas os itens I e III estão certos.

#### ||JUSTIFICATIVAS||

||A|| - Incorreta. Os itens I e III estão certos.\*/

||B|| - Incorreta. O item II é o único item errado.\*/

||C|| - Incorreta. O item I também está certo.\*/

||D|| - Incorreta. O item II está errado e o item III está certo.\*/

||E|| - Correta. O item I está certo, pois a PCA é uma técnica de redução de dimensionalidade que se dá por meio da criação de componentes principais, que são combinações lineares das variáveis originais. A maior parte da variância nos dados é mantida nos primeiros componentes, enquanto as variáveis correlacionadas são combinadas e transformadas em componentes ortogonais (não correlacionados). Isso simplifica a análise e a visualização dos dados sem perda significativa de informação.

O item II está errado, pois o *k-means* não utiliza dendrogramas. Ele tem como base a definição de um número fixo de *clusters*, de modo que lhes atribui pontos de acordo com a distância até os centros (ou centroides). Além disso, dendrogramas são utilizados no agrupamento hierárquico, não no *k-means*.

O item III está certo, pois o Scikit-learn oferece suporte abrangente para técnicas de aprendizado não-supervisionado, como clusterização, modelagem probabilística de dados com distribuições gaussianas e redução de dimensionalidade com



PCA no módulo `sklearn.decomposition`. Essas funcionalidades são muito utilizadas na descoberta de padrões em dados não-rotulados.\*/\*

### Questão 61

Acerca das regras de associação e do aprendizado não supervisionado por meio da biblioteca Scikit-learn do Python, assinale a opção correta.

- A No algoritmo *Apriori*, parâmetros como suporte mínimo e confiança são fundamentais para filtrar e determinar a relevância das regras geradas.
- B Regras de associação são aplicáveis nos modelos de aprendizado supervisionado e somente em um modelo de aprendizado não supervisionado.
- C O aumento dos valores de suporte mínimo e confiança no algoritmo *Apriori* tende a aumentar o número de regras geradas.
- D O Scikit-learn possui implementação nativa para a extração de regras de associação com base no algoritmo *Apriori*.
- E A métrica *lift* é utilizada apenas para calcular a frequência dos itens no conjunto de dados, de modo que não é útil para avaliar a relevância das regras geradas.

#### ||JUSTIFICATIVAS||

||A|| - Correta. Descreve com precisão a importância dos parâmetros no *Apriori*.\*/

||B|| - Incorreta. As regras de associação pertencem ao aprendizado não supervisionado.\*/\*

||C|| - Incorreta. O aumento dos valores de suporte mínimo e confiança reduz o número de regras geradas, o que torna menor o número de regras geradas.\*/\*

||D|| - Incorreta. O Scikit-learn não possui implementação nativa para o *Apriori* ou regras de associação.\*/\*

||E|| - Incorreta. A métrica *lift* não é utilizada apenas para calcular frequências, mas também para avaliar a força das associações, de modo que se faz essencial para determinar a relevância das regras geradas.\*/\*

### Questão 62

A respeito das redes neurais artificiais, assinale a opção correta.

- A Redes neurais artificiais são compostas por uma única camada de neurônios e são incapazes de modelar relações complexas entre os dados.
- B O algoritmo *Backpropagation* ajusta os pesos da rede neural aleatoriamente em cada iteração, sem considerar gradientes ou uma função de custo específica.
- C Redes neurais convolucionais não possuem camadas de ativação e dependem das operações de convolução para processar os dados.
- D Redes neurais convolucionais podem ser usadas tanto para a classificação, em que a rede atribui rótulos a imagens inteiras, quanto para a segmentação, em que a rede identifica regiões específicas da imagem e fornece um mapa *pixel a pixel*.

- E Em Python, bibliotecas como Keras e PyTorch não permitem o uso de GPUs para acelerar o treinamento de redes neurais, limitando-se ao processamento em CPU.

#### ||JUSTIFICATIVAS||

||A|| - Incorreta. Redes neurais artificiais não são limitadas a uma única camada de neurônios. Elas podem ser compostas por várias camadas (aprendizado profundo), o que permite a modelagem de relações complexas entre os dados.\*/\*

||B|| - Incorreta. O algoritmo *Backpropagation* ajusta os pesos a partir do gradiente do erro em relação aos pesos, calculado com base na regra de cadeia. Ele não ajusta pesos aleatoriamente, mas de maneira sistemática para minimizar a função de custo.\*/\*

||C|| - Incorreta. Redes neurais convolucionais utilizam camadas de ativação juntamente com operações de convolução para introduzir não linearidades no modelo e melhorar a capacidade de aprendizado.\*/\*

||D|| - Correta. Redes neurais convolucionais podem ser aplicadas em tarefas como classificação e segmentação. Na classificação, o objetivo é atribuir um único rótulo à imagem inteira. Na segmentação, o foco é identificar regiões específicas da imagem, o que gera um mapa que indica a classe de cada *pixel*.\*/

||E|| - Incorreta. Em Python, bibliotecas como Keras e PyTorch permitem o uso de GPUs para acelerar o treinamento de redes neurais. Essa é uma das principais vantagens dessas bibliotecas, especialmente ao lidar com grandes volumes de dados e redes complexas.\*/\*

### Questão 63

A respeito do planejamento e da gestão estratégicos de TIC, assinale a opção correta.

- A Os planos diretores de TIC dos órgãos do SISP devem estar alinhados de forma integral aos objetivos estratégicos e metas propostos na estratégia de governança digital.
- B A estratégia de governança digital e o plano diretor de TIC devem ser integrados ao plano estratégico institucional e direcionados pelo plano estratégico de TIC.
- C O plano diretor de TIC e o plano estratégico de TIC, situados no nível estratégico, são documentos que complementam o plano estratégico institucional.
- D Prescinde-se no plano diretor de TIC da definição de indicadores, uma vez que estes devem ser elencados, inicialmente, no plano estratégico de TIC, em conformidade com os objetivos estratégicos.
- E Os planos de ação, no nível tático, propiciam a execução das ações e o alcance das metas descritas no plano diretor de TIC, de forma que é essencial que esses planos estejam alinhados às soluções de TIC e às metas da organização.

#### ||JUSTIFICATIVAS||

||A|| - Correta. Os planos diretores de TIC de órgãos e entidades do SISP devem estar alinhados de forma integral aos objetivos estratégicos e metas propostos na estratégia de governança digital. O Decreto n.º 8.638/2016, artigo 5.º, estabelece que o Ministro de Estado do Planejamento, Orçamento e Gestão editará a estratégia de governança digital da administração pública federal, documento que definirá os objetivos estratégicos, as metas, os indicadores e as iniciativas da Política de Governança Digital e norteará programas, projetos, serviços, sistemas e atividades a ela relacionados.\*/\*

||B|| - Incorreta. O plano estratégico de TIC e/ou o plano diretor de TIC devem ser integrados ao plano estratégico institucional e ser direcionados pela estratégia de governança digital. O plano estratégico institucional é um documento que se situa no nível estratégico, que determina as políticas e as estratégias que governarão a organização e define os principais objetivos a serem alcançados. Também resolve o uso e a disponibilização

dos recursos para a realização desses objetivos, sendo composto por premissas, planejamento, implementação e revisão. Ele é elaborado, implementado e avaliado a partir de uma metodologia adequada à realidade de cada organização.\*

||C|| - Incorreta. No nível tático, o instrumento mais comumente usado para representar o planejamento de TIC é o plano diretor de TIC. O plano estratégico de TIC, situado no nível estratégico, é um documento que complementa o plano estratégico institucional por meio do planejamento dos recursos de TIC, de modo que possibilita a definição de objetivos específicos para a área.\*

||D|| - Incorreta. O plano diretor de TIC deve definir indicadores, em conformidade com os objetivos estratégicos da TIC, e conter o planejamento de investimentos necessários, o quantitativo e a capacitação de pessoas, bem como a identificação e o tratamento de riscos relacionados à TIC.\*

||E|| - Incorreta. É fundamental que o plano diretor de TIC proporcione o alinhamento das soluções de TIC às metas do negócio e às necessidades da organização. Dessa maneira, o planejamento de TIC complementa o planejamento estratégico da organização, bem como permite o alinhamento das expectativas entre as diversas áreas da organização a respeito dos recursos de TIC. Como consequência, garante-se que as soluções de TIC estejam em conformidade com os objetivos e as iniciativas da área, de modo a possibilitar a definição dos planos de ação. Por fim, os planos de ação, no nível operacional, auxiliam a execução das ações e o alcance das metas, alinhados ao plano diretor de TIC.\*

#### Questão 64

Suponha que, em uma organização, seja necessário implementar ações com o objetivo de possibilitar mudanças em todos os diferentes aspectos da organização — como canais e processos a dados —, garantir que o nível desejado de digitalização seja parte integrante da direção futura de I&T, avaliar a atual maturidade digital da organização e repensar as operações internas e as atividades voltadas para o cliente.

Com base no COBIT 2019, assinale a opção que corresponde ao processo diretamente afeto às ações da situação hipotética precedente.

- A inovação gerenciada
- B entrega de benefícios assegurada
- C engajamento com as partes assegurada
- D programas gerenciados
- E estratégia gerenciada

||JUSTIFICATIVAS||

||A|| - Incorreta.\*

||B|| - Incorreta.\*

||C|| - Incorreta.\*

||D|| - Incorreta.\*

||E|| - Correta. São ações da estratégia gerenciada:

- apoiar a estratégia de transformação digital da organização e entregar o valor desejado por meio de um roteiro de mudanças incrementais;
- utilizar uma abordagem holística de I&T, de modo a assegurar que cada iniciativa esteja claramente ligada a uma estratégia abrangente;
- possibilitar mudanças em todos os diferentes aspectos da organização, como canais e processos a dados, cultura, competências, modelo operacional e incentivos;
- fornecer uma visão holística do ambiente atual de negócios e de I&T, da direção futura e das iniciativas necessárias para garantir um bom futuro ao meio ambiente;
- garantir que o nível desejado de digitalização seja parte integrante da direção futura e da estratégia de I&T;
- avaliar a atual maturidade digital da organização e desenvolver um roteiro para fechar as lacunas;
- repensar as operações internas e as atividades voltadas para o cliente;
- garantir o foco na jornada de transformação em toda a organização;

- aproveitar os blocos de construção da arquitetura empresarial, os componentes de governança e o ecossistema da organização — como os serviços fornecidos externamente e os recursos relacionados — para obter uma resposta confiável, ágil e eficiente aos objetivos estratégicos.\*

#### Questão 65

No contexto do COBIT 2019, a continuidade e a disponibilidade de serviços de negócios são consideradas como

- A uma métrica do processo de continuidade gerenciada que mensura a quantidade de tempo necessária para reparação e recuperação do ativo após um incidente.
- B um objetivo de alinhamento associado aos objetivos empresariais, correlacionado, especificamente, aos processos continuidade gerenciada e disponibilidade gerenciada.
- C um processo do COBIT específico do domínio DSS de entrega, serviço e suporte.
- D um objetivo empresarial associado aos objetivos de alinhamento, correlacionado, especificamente, ao processo de continuidade gerenciada.
- E uma métrica do processo de disponibilidade e capacidade gerenciadas que possui KPI igual a 0, uma vez que mensura o período médio de indisponibilidade do ativo/sistema de TI.

||JUSTIFICATIVAS||

||A|| - Incorreta. São um objetivo empresarial.\*

||B|| - Incorreta. São um objetivo empresarial.\*

||C|| - Incorreta. São um objetivo empresarial.\*

||D|| - Correta. Os objetivos empresariais incluem continuidade e disponibilidade de serviços de negócios.\*

||E|| - Incorreta. São um objetivo empresarial.\*

#### Questão 66

Assinale a opção que descreve corretamente características da *sprint* no Scrum.

- A Em que se pese manter objetivo e duração fixos, a *sprint* não pode ser considerada um projeto curto.
- B A *sprint* pode ter duração fixa de 10 dias, e a *daily scrums*, a *sprint review* e a *sprint retrospective* acontecem dentro da *sprint*.
- C Uma *sprint* somente pode ser cancelada pelo *scrum master* se a meta da *sprint* tornar-se obsoleta.
- D Durante a *sprint*, nenhuma mudança é permitida — nem mesmo o refinamento do *product backlog* —, uma vez que mudanças põem em risco a meta da *sprint*.
- E A *sprint* encerra-se após a conclusão das tarefas elencadas na *sprint backlog*, e imediatamente após a conclusão devem ser realizadas, fora da *sprint*, a *sprint review* e a *sprint retrospective*, com vistas a inspecionar o resultado obtido.

||JUSTIFICATIVAS||

||A|| - Incorreta. Cada *sprint* pode ser considerada um projeto curto. Durante a *Sprint*, não é feita nenhuma mudança que coloque em risco a meta estabelecida, não há queda de qualidade e o *product backlog* é refinado.\*

||B|| - Correta. Trata-se de eventos de duração fixa de um mês ou menos para se criar consistência. Uma nova *sprint* começa imediatamente após a conclusão da *sprint* anterior. Todo o trabalho necessário para atingir a meta do produto — incluindo-se *sprint planning*, *daily scrums*, *Sprint review* e *sprint retrospective* — acontece dentro de *sprints*.\*

||C|| - Incorreta. Uma *Sprint* pode ser cancelada se a meta da *sprint* tornar-se obsoleta. Apenas o *product owner* tem autoridade para cancelar a *sprint*.\*

||D|| - Incorreta. Cada *sprint* pode ser considerada um projeto curto. Durante a *sprint*, não é feita nenhuma mudança que coloque em risco a meta estabelecida, não há queda de qualidade e o *product backlog* é refinado.\*

||E|| - Incorreta. Trata-se de eventos de duração fixa de um mês ou menos para se criar consistência. Uma nova *sprint* começa imediatamente após a conclusão da *sprint* anterior. Todo o trabalho necessário para atingir a meta do produto — incluindo-se *sprint planning*, *daily scrums*, *sprint review* e *sprint retrospective* — acontece dentro de *sprints*.\*/

### Questão 67

No PMBOK 7.<sup>a</sup> edição, o domínio no qual o resultado desejado é um ciclo de vida de projetos que consiste em fases que facilitam a abordagem de entrega e desenvolvimento e que são necessárias para produzir produtos entregues pelo projeto é denominado domínio

- A da equipe.
- B das partes interessadas.
- C da abordagem de desenvolvimento e ciclo de vida.
- D do trabalho do projeto.
- E do planejamento.

#### ||JUSTIFICATIVAS||

||A|| - Incorreta. Domínio de desempenho da abordagem de desenvolvimento e ciclo de vida.\*/

||B|| - Incorreta. Domínio de desempenho da abordagem de desenvolvimento e ciclo de vida.\*/

||C|| - Correta. Um domínio de desempenho de projetos é um grupo de atividades cruciais para a entrega eficaz dos resultados do projeto. As áreas de foco são interativas, interrelacionadas e interdependentes e trabalham juntas para alcançar os resultados desejados do projeto.

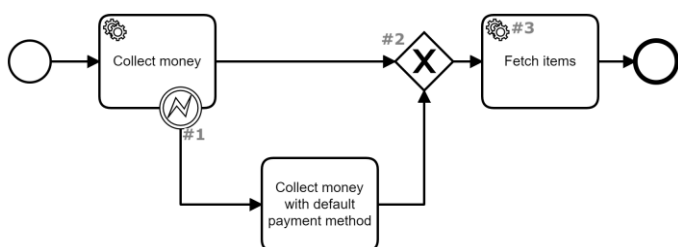
O domínio de desempenho da abordagem de desenvolvimento e do ciclo de vida trata das atividades e funções associadas às fases de abordagem de desenvolvimento, cadência e ciclo de vida do projeto. A execução efetiva deste domínio de desempenho entrega os seguintes resultados desejados:

- abordagens de desenvolvimento que são consistentes com os resultados do projeto;
- um ciclo de vida de projetos consiste em fases que conectam a entrega de valor de negócios e partes interessadas do início ao fim do projeto;
- um ciclo de vida de projetos consiste em fases que facilitam a abordagem de entrega e desenvolvimento necessárias para produzir os produtos entregues pelo projeto.\*/

||D|| - Incorreta. Domínio de desempenho da abordagem de desenvolvimento e ciclo de vida.\*/

||E|| - Incorreta. Domínio de desempenho da abordagem de desenvolvimento e ciclo de vida.\*/

### Questão 68



À luz da notação BPMN, é correto afirmar que, na figura precedente,

- A #1 indica um *error start event* que só é permitido para acionar um subprocesso de evento em linha.
- B #2 indica um *parallel gateway* que cria caminhos paralelos sem verificar nenhuma condição.
- C #2 indica um *complex gateway* que pode ser usado para modelar o comportamento de sincronização complexa quando há mais de um fluxo de entrada.

D #3 indica uma *business rule task* que descreve uma tarefa em que é fornecido um mecanismo para que o processo, a partir de uma entrada, receba uma saída com base nos cálculos da regra de negócio.

E #3 indica uma *manual task* que descreve uma tarefa que se espera ser executada sem o auxílio de qualquer mecanismo de execução de processos de negócios ou qualquer aplicativo.

#### ||JUSTIFICATIVAS||

||A|| - Correta. O *error start event* só é permitido para acionar um subprocesso de evento em linha. Se houver apenas um *EventDefinition* associado ao evento de início e este for da subclasse *ErrorEventDefinition*, o evento será de início de erro e usará um marcador de relâmpago. Dada a natureza dos erros, um subprocesso de evento com um gatilho de erro sempre interromperá o processo que o contém.\*/

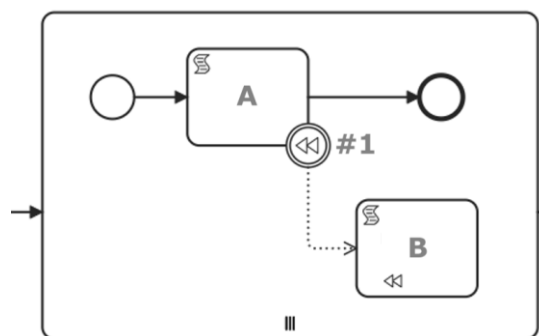
||B|| - Incorreto. O evento é *exclusive gateway*.\*/

||C|| - Incorreto. O evento é *exclusive gateway*.\*/

||D|| - Incorreto. #3 é um *Service Task*. Uma tarefa de serviço pode ser um serviço *web* ou um aplicativo automatizado. Uma tarefa de envio é uma tarefa simples projetada para enviar uma mensagem a um participante externo (em relação ao processo). Uma vez que a mensagem foi enviada, a tarefa é concluída.\*/

||E|| - Incorreto. #3 é um *Service Task*. Uma tarefa de serviço pode ser um serviço *web* ou um aplicativo automatizado. Uma tarefa de envio é uma tarefa simples projetada para enviar uma mensagem a um participante externo (em relação ao processo). Uma vez que a mensagem foi enviada, a tarefa é concluída.\*/

### Questão 69



Com base na notação BPMN, é correto afirmar que, na figura precedente, #1 indica

- A um evento que descreve que a tarefa referenciada deverá ser cancelada e no qual será acionado um evento intermediário anexado à tarefa.
- B um subprocesso de eventos de escalonamento que implementa medidas para agilizar a conclusão de uma atividade de negócio, caso não satisfaça uma restrição especificada quanto à execução — um prazo, por exemplo.
- C um subprocesso que envia um sinal que interrompe a tarefa referenciada e indica o limite de acionamento do evento.
- D um evento múltiplo em que há várias maneiras de acionar a tarefa referenciada, de modo que apenas um deles é necessário para realmente iniciar um evento.
- E um evento que descreve uma compensação que se preocupa em desfazer etapas que já foram concluídas com sucesso, porque os resultados e os possíveis efeitos colaterais não são mais desejados e precisam ser revertidos.

#### ||JUSTIFICATIVAS||

||A|| - Incorreta. A opção corresponde a *cancel*.\*/



||B|| - Incorreta. A opção corresponde a *escalation*.\*/

||C|| - Incorreta. A opção corresponde a *signal*.\*/

||D|| - Incorreta. A opção corresponde a *multiple*.\*/

||E|| - Correta. Uma compensação que se preocupa em desfazer etapas que já foram concluídas com sucesso, porque os resultados e os possíveis efeitos colaterais não são mais desejados e precisam ser revertidos.\*/



### Questão 70

Assinale a opção em que é citada a ciência do desenvolvimento de algoritmos e modelos estatísticos utilizados por sistemas de computador para a realização de tarefas sem instruções explícitas, baseando-se em padrões e inferências, de modo que um sistema aprenda e melhore automaticamente e com base na experiência.

A *internet of things*

B *generative AI*

C *big data*

D *machine learning*

E *cloud computing*

#### ||JUSTIFICATIVAS||

||A|| - Incorreta.\*/

||B|| - Incorreta.\*/

||C|| - Incorreta.\*/

||D|| - Correta. O *machine learning* é um subconjunto da inteligência artificial que permite que uma máquina ou um sistema aprenda e melhore com base na exposição de dados, de forma automática. Em vez de uma programação explícita, o *machine learning* usa algoritmos para analisar grandes volumes de dados, aprender com os *insights* e tomar decisões embasadas.\*/

||E|| - Incorreta.\*/