



# INSTITUTO NACIONAL DE PESQUISA ESPACIAL – INPE

## PROVA OBJETIVA

# TG18

## DESENVOLVIMENTO DE SOFTWARE PARA OPERAÇÃO DE SATÉLITES



### SUA PROVA

- Além deste caderno contendo **45 (quarenta e cinco)** questões objetivas, você receberá do fiscal de prova o cartão de respostas;
- As questões objetivas têm **5 (cinco)** opções de resposta (A, B, C, D e E) e somente uma delas está correta.



### TEMPO

- Você dispõe de **4 (quatro) horas** para a realização da prova;
- **2 (duas) horas** após o início da prova, é possível retirar-se da sala, sem levar o caderno de questões;
- A partir dos **30 (trinta) minutos** anteriores ao término da prova é possível retirar-se da sala **levando o caderno de questões**.



### NÃO SERÁ PERMITIDO

- Qualquer tipo de comunicação entre os candidatos durante a aplicação da prova;
- Anotar informações relativas às respostas em qualquer outro meio que não seja no caderno de questões e nas folhas de textos definitivos;
- Levantar da cadeira sem autorização do fiscal de sala;
- Usar o sanitário ao término da prova, após deixar a sala.



### INFORMAÇÕES GERAIS

- Verifique se seu caderno de questões está completo, sem repetição de questões ou falhas e também confira seu cargo. Caso tenha recebido caderno de cargo **diferente** do impresso em seu cartão de respostas, o fiscal deve ser **obrigatoriamente** informado para o devido registro na ata da sala;
- Confira seus dados pessoais, especialmente nome, número de inscrição e documento de identidade e leia atentamente as instruções para preencher o cartão de respostas;
- Para o preenchimento do cartão de respostas, use somente caneta esferográfica, fabricada em material transparente, com tinta preta ou azul;
- Assine seu nome apenas no(s) espaço(s) reservado(s) no cartão de respostas;
- Reserve tempo suficiente para o preenchimento do seu cartão de respostas. O preenchimento é de sua responsabilidade e **não será permitida a troca do cartão de respostas em caso de erro cometido pelo candidato**;
- Para fins de avaliação, serão levadas em consideração apenas as marcações realizadas no cartão de respostas;
- A FGV coletará as impressões digitais dos candidatos na lista de presença;
- Os candidatos serão submetidos ao sistema de detecção de metais quando do ingresso e da saída de sanitários durante a realização das provas.

**Boa Prova!**



## CONHECIMENTOS ESPECÍFICOS

### 1

Considerando o contexto de desenvolvimento de *software*, determinadas práticas são essenciais para o bom andamento de um projeto. Um exemplo de boa prática é o uso de *tags*, empregadas na maioria dos sistemas de versionamento modernos.

Sobre as *tags*, assinale a afirmativa correta.

- (A) São frequentemente atribuídas a *commits* consecutivos, para refletir o avanço da equipe.
- (B) Servem para marcar permanentemente pontos importantes do desenvolvimento de um projeto, como o lançamento de novas versões ou outros pontos de referência.
- (C) Devem ser aplicadas automaticamente a cada *commit*, para garantir que os pontos do desenvolvimento sejam preservados para referência futura, maximizando eficiência.
- (D) Têm o seu uso recomendado para documentar alterações em tempo real em um repositório, para facilitar o rastreamento diário do progresso.
- (E) Servem como uma alternativa aos *commits* para indicar mudanças incrementais no código, auxiliando a manutenção de um histórico do repositório mais limpo.

### 2

*Git* é um sistema de controle de versão gratuito e criado principalmente para o desenvolvimento de *software*.

Supondo que um repositório *Git* local possui apenas o *commit* inicial, um novo *branch* local poderia ser criado por meio do comando

- (A) `git commit new_branch`.
- (B) `git branch new_branch`.
- (C) `git push origin new_branch`.
- (D) `git create_branch new_branch`.
- (E) `git checkout new_branch`.

### 3

Uma linguagem de programação Orientada a Objetos deve prover suporte aos principais fundamentos do desenvolvimento Orientação a Objetos. Entretanto, cada linguagem apresenta as suas especificidades e formas de implementar esses fundamentos.

Sobre a Linguagem C++, analise as afirmativas a seguir.

- I. Um ponteiro *this* é um ponteiro para um objeto da própria classe e tem como valor o endereço do objeto ao qual se refere.
- II. Quando internamente a uma ClasseA, define-se outra ClasseB como Friend, significa que a ClasseA está permitindo que a ClasseB acesse seus membros privados e protegidos.
- III. A linguagem C++ permite herança múltipla.

Está correto o que se afirma em

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

### 4

SQL (*Structured Query Language*) é uma linguagem usada para interagir com bancos de dados relacionais.

Em relação ao SQL, analise as afirmativas a seguir e assinale (V) para a verdadeira e (F) para a falsa.

- ( ) A cláusula WHERE só é utilizada associada ao comando SELECT.
- ( ) O Comando CREATE TABLE cria uma nova tabela no banco de dados e o comando INSERT INTO permite popular essa tabela por meio da inserção de registros.
- ( ) O comando INNER JOIN é utilizado para unir os resultados de duas ou mais consultas SELECT e o seu resultado é um conjunto de linhas provenientes das tabelas envolvidas.
- ( ) Os comandos ALTER TABLE e UPDATE são comandos usados para atualizar registros em tabelas de um banco de dados.
- ( ) SQL suporta o conceito de chaves primárias e chaves estrangeiras para manter a integridade dos dados e as relações entre as tabelas.

As afirmativas são, respectivamente,

- (A) F – F – F – V – F.
- (B) F – V – F – F – V.
- (C) V – V – V – F – F.
- (D) V – F – V – V – V.
- (E) F – F – V – F – F.

### 5

Sistemas de versionamento de *software* são ferramentas amplamente utilizadas no desenvolvimento de aplicações.

Nesse contexto, mais especificadamente sobre o *sistema Git*, assinale a afirmativa correta.

- (A) Cada cópia de trabalho do código é um repositório completo, com histórico e capacidades de versionamento.
- (B) O *Git* é único sistema que permite a utilização de ramificações (*branches*) e etiquetas (*tags*).
- (C) Um projeto possui três seções principais: diretório *Git*, diretório de trabalho (*working directory*) e área de lançamento (*launching area*).
- (D) O *Git* organiza os *commits* em uma estrutura de dados linear, que simplifica o rastreamento de alterações.
- (E) O *Git* grava todas as operações de *commit* diretamente no servidor central, exigindo conexão constante para que as alterações sejam registradas.

### 6

Com relação aos diversos modelos de ciclo de vida de um *software*, considere a seguinte situação hipotética:

*Uma equipe inicia um projeto com uma abordagem linear e sequencial. No entanto, ao longo do projeto substitui esta abordagem por um modelo com ênfase em protótipos no início de cada estágio para validação de conceitos e na análise de riscos.*

Assinale a opção que apresenta o modelo que esta equipe empregou inicialmente e o modelo pelo qual este foi substituído.

- (A) cascata e *scrum*.
- (B) desenvolvimento *lean* e espiral.
- (C) cascata e espiral.
- (D) espiral e RAD.
- (E) cascata e incremental.

**7**

O *Scrum* é um *framework* de gerenciamento que as equipes usam para se auto-organizar e trabalhar em direção a um objetivo em comum.

Sobre o *Scrum*, analise as afirmativas a seguir.

- I. Por ser uma metodologia ágil, um de seus pilares é a eliminação de reuniões.
- II. O *backlog* do produto é uma lista dinâmica de funcionalidades organizada por prioridades.
- III. O *Product Owner* planeja os recursos necessários para cada *Sprint*.

Está correto o que se afirma em

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

**8**

No contexto de Projetos Orientados a Objetos, padrões de projetos são soluções generalizadas para problemas comuns de *design* de *software*.

Considere uma situação em que um desenvolvedor foi incumbido de elaborar um sistema de criação de documentos de diversos formatos, como Texto, Planilha e Apresentação, a serem definidos com base nos comandos do usuário.

Para lidar com esses requisitos, o padrão de *design* de *software* mais adequado seria o

- (A) Singleton.
- (B) Factory Method.
- (C) Heritage.
- (D) Builder.
- (E) Strategy.

**9**

Um sistema distribuído tem como objetivo coletar e distribuir dados meteorológicos para um conjunto de usuários. Para tal, foram empregados pelo desenvolvedor dois padrões de *design*:

**Padrão (1):** Criação de uma classe com apenas uma instância, responsável por controlar o acesso a um determinado sensor operado por equipes técnicas remotas.

**Padrão (2):** Definição de um objeto principal, de forma que todos os usuários dependentes são notificados e atualizados em tempo real.

Assinale a opção que indica os padrões de *design* (1) e (2), respectivamente.

- (A) Singleton e Heritage.
- (B) Strategy e Composite.
- (C) Strategy e Observer.
- (D) Singleton e Observer.
- (E) Builder e Factory.

**10**

O Projeto Orientado a Serviços (SOA) tem como objetivo facilitar a integração entre diferentes serviços.

Sobre o emprego de SOA, assinale a afirmativa correta.

- (A) Permite o compartilhamento direto de bancos de dados entre diferentes serviços para reutilização de funcionalidades.
- (B) Busca elevar o grau de acoplamento entre serviços com o objetivo de garantir compatibilidade.
- (C) A lógica é abstraída através de interfaces de serviço, de modo a permitir a integração entre sistemas heterogêneos.
- (D) A segurança é intrinsecamente garantida pela arquitetura, eliminando a necessidade de protocolos de segurança adicionais.
- (E) Não é compatível com o padrão SOAP, por conta de sua restrição ao uso de XML para formatação de mensagens.

**11**

A integração de módulos e sistemas é uma etapa crucial no desenvolvimento de *software*, em que a abordagem de integração a ser tomada depende do cenário concreto.

Assinale a opção que apresenta o cenário para o qual a abordagem *Big Bang* é a mais adequada.

- (A) Testar gradualmente os diferentes componentes do sistema.
- (B) Resolver os problemas conhecidos de integração entre determinados módulos.
- (C) Identificar os problemas de integração de maneira incremental.
- (D) Realizar uma abordagem cuidadosa para garantir a integração correta entre os módulos.
- (E) Integrar simultaneamente todos os módulos e testá-los em conjunto.

**12**

O desenvolvimento orientado a testes (TDD) engloba um conjunto de práticas que visam garantir a detecção de erros e a qualidade do código gerado.

No escopo do TDD, assinale a opção que apresenta a característica do teste unitário.

- (A) Ser realizado desde o início do projeto.
- (B) Ser desnecessário em pequenos projetos.
- (C) Avaliar mudanças em um sistema como um todo.
- (D) Ser substituível por testes de regressão.
- (E) Verificar o funcionamento de módulos em conjunto.

**13**

Em um projeto de desenvolvimento de *software*, uma equipe está focada em diferentes aspectos do sistema.

Assinale a opção que indica o diagrama UML mais adequado para se entender como os objetos do sistema interagem entre si ao longo do tempo.

- (A) Diagrama de Sequência.
- (B) Diagrama de Classes.
- (C) Diagrama de Objetos.
- (D) Diagrama de Estado.
- (E) Diagrama de Casos de Uso.

**14**

Ao se escolher a arquitetura de *software* para um projeto, é fundamental considerar os requisitos a serem priorizados.

A arquitetura de microsserviços, que uma evolução do estilo de arquitetura orientada a serviços (SOA), é mais indicada do que a arquitetura monolítica, quando a prioridade é otimizar

- (A) o tempo de resposta.
- (B) a simplicidade de desenvolvimento.
- (C) a testabilidade.
- (D) a disponibilidade.
- (E) a comunicação entre componentes.

**15**

Acerca de *metodologias ágeis*, assinale a afirmativa correta.

- (A) Satisfazer o cliente através da entrega antecipada e contínua de *software* valioso não é uma das prioridades do Manifesto Ágil.
- (B) É responsabilidade do *Scrum Master* criar e comunicar de maneira clara os itens do *Product Backlog*.
- (C) Integração Contínua é o processo no qual os desenvolvedores recriam o código continuamente assim que melhorias são identificadas, o que contribui para a simplicidade e facilidade de manutenção do código.
- (D) A *Sprint Review* é uma representação altamente visível e em tempo real do trabalho que os desenvolvedores planejam realizar durante a *Sprint*, cujo propósito é alcançar a Meta da *Sprint*.
- (E) Na metodologia XP, a elicitação de requisitos é conduzida pelos próprios membros da equipe de desenvolvimento e os requisitos são desenvolvidos de maneira incremental, conforme as prioridades do usuário.

**16**

A respeito da gestão de projetos que emprega *metodologias ágeis*, assinale a afirmativa correta.

- (A) *Times Scrum* possuem a capacidade de se auto-organizar, o que significa que decidem a melhor maneira de completar seu trabalho, ao invés de serem dirigidos por indivíduos externos à equipe.
- (B) *Scrum Masters* são líderes genuínos que servem tanto ao *Time Scrum* quanto à organização como um todo. Eles têm a responsabilidade de desenvolver e comunicar de forma explícita a meta do produto, além de criar e comunicar de maneira clara os itens do *Product Backlog*.
- (C) Na metodologia XP, é considerado aceitável o uso de grandes quantidades de horas extras, pois, a médio prazo, não há uma diminuição na qualidade do código e na produtividade.
- (D) O planejamento do projeto é comumente realizado em sua fase inicial, detalhando todo o trabalho a ser desenvolvido nas fases posteriores.
- (E) No papel de servir à organização, o *Product Owner* é responsável por liderar, treinar e orientar a organização no emprego do *Scrum*, além de remover barreiras entre os *stakeholders* e os *Times Scrum*.

**17**

Com relação às linguagens de programação C/C++, analise as afirmativas a seguir.

- I. Seja *x* uma variável do tipo inteiro. Na declaração abaixo, o ponteiro *p* é inicializado com o endereço de *x*. `int *p = &x`.
- II. O comando *break* somente pode ser utilizado em conjunto com o comando *switch*.
- III. O comando *return* encerra a execução de uma função.

Está correto o que se afirma em

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

**18**

Com relação à linguagem de programação C++ e o paradigma da orientação a objeto, analise as afirmativas a seguir.

- I. Uma classe define o comportamento dos objetos que são instâncias da classe.
- II. Em C++ é permitido criar classes derivadas, seguindo o conceito de herança de classes.
- III. O polimorfismo permite que objetos de classes diferentes respondam de forma diferente à mesma função.

Está correto o que se afirma em

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

**19**

Com relação à linguagem de programação Python, analise os itens a seguir:

- I. Seja sequência de operações a seguir:

```
>>> x=[1, 2, 3]
>>> y=x
>>> y[0]=0
>>> print(x)
```

O valor de *x* impresso na tela é: [1, 2, 3]

- II. Seja sequência de operações a seguir:

```
>>> x=[1,4,1,2,3,2]
>>> y=set(x)
>>> print(y)
```

O valor de *y* impresso na tela é: {1, 4, 1, 2, 3, 2}

- III. A expressão abaixo cria um dicionário vazio.

```
x = {}
```

Está correto o que se afirma em:

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

**20**

Em relação à linguagem de programação Python, analise o código abaixo.

```
y = range(2,8)
for n in y:
    print( pow(n,1) & 0x3 )
```

Sabendo-se que o código acima corresponde a um programa, que é executado com sucesso, ele terá como resultado os seguintes números impressos:

- (A) 3 – 0 – 1 – 2 – 3.
- (B) 2 – 3 – 0 – 1 – 2 – 3.
- (C) 2 – 3 – 0 – 1 – 2 – 3 – 0.
- (D) 3 – 4 – 0 – 1 – 3 – 7 – 0.
- (E) 3 – 0 – 1 – 2 – 3 – 0.

**21**

Em relação a Bancos de Dados Relacionais, analise as afirmativas a seguir.

- I. As tabelas são os componentes fundamentais de um banco de dados relacional e são usadas para organizar e armazenar dados de forma estruturada.
- II. A chave primária é uma coluna (ou conjunto de colunas) com valor único para cada linha de uma tabela e é usada para identificar registros de forma única.
- III. Os relacionamentos em um banco de dados relacionais são conexões lógicas entre diferentes tabelas que permitem combinar dados e estão restritos a relacionamentos um-para-um entre tabelas.
- IV. A sigla ACID está relacionada à manipulação de dados em bancos de dados relacionais onde o “I” da sigla refere-se à inserção (Insert) de registros em uma tabela e o “D” à deleção (Delete) de registros em uma tabela.
- V. As tabelas podem estar relacionadas entre si por meio de chaves estrangeiras.

Está correto o que se afirma em

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

**22**

Para que o sistema de banco de dados seja funcional, ele precisa recuperar dados de maneira eficiente. A necessidade de eficiência tem levado os projetistas a usarem estruturas de dados complexas para representar dados no banco de dados. Como muitos usuários de sistema de banco de dados não são treinados em computação, os desenvolvedores ocultam a complexidade dos usuários sob vários níveis de abstração de dados, para simplificar as interações do usuário com o sistema. Com relação à abstração de dados, analise as afirmativas a seguir e assinale (V) para a verdadeira e (F) para a falsa.

- ( ) O nível lógico descreve em detalhes estruturas de dados complexas de baixo nível.
- ( ) O nível de visão fornece um mecanismo de segurança de modo a evitar que os usuários acessem certas partes do banco de dados.
- ( ) O nível físico descreve quais dados estão armazenados no banco de dados e que relações existem entre eles.
- ( ) O nível lógico descreve o banco de dados inteiro em termos de um pequeno número de estruturas relativamente simples.

As afirmativas são, respectivamente,

- (A) F – V – F – V.
- (B) V – V – F – V.
- (C) V – F – F – V.
- (D) V – V – V – F.
- (E) F – V – V – F.

**23**

SQL (*Structured Query Language*) é uma linguagem declarativa padrão usada para gerenciar e manipular bancos de dados relacionais. Ela fornece um conjunto de comandos que permitem aos usuários realizarem diversas operações, como consultar, inserir, atualizar e excluir dados de bancos de dados relacionais.

Relacione as linguagens declarativas às suas respectivas propriedades.

1. DDL – Data Definition Language.
  2. DML – Data Manipulation Language.
  3. DTL – Data Transaction Language.
  4. DCL – Data Control Language.
- ( ) Contém o comando SELECT.
  - ( ) Contém os comandos COMMIT e ROLLBACK.
  - ( ) Utilizada para dar acesso aos usuários.
  - ( ) Contém os comandos CREATE, ALTER e DROP.
  - ( ) Contém os comandos INSERT, UPDATE e DELETE.

Assinale a opção que indica a relação correta, segundo a ordem apresentada.

- (A) 1 – 4 – 4 – 1 – 2.
- (B) 2 – 3 – 4 – 1 – 2.
- (C) 2 – 3 – 3 – 1 – 2.
- (D) 3 – 4 – 3 – 2 – 1.
- (E) 2 – 3 – 4 – 3 – 1.

**24**

Os comandos SQL do tipo DML (*Data Manipulation Language*) são utilizados para manipular os dados no banco de dados.

Esses comandos permitem aos usuários interagirem com o banco de dados para executar operações.

Assinale a opção que apresenta apenas exemplos de comandos do tipo DML.

- (A) SELECT, INSERT, DROP e UPDATE.
- (B) INSERT, DROP e UPDATE.
- (C) SELECT, INSERT, DELETE e UPDATE.
- (D) CREATE, ALTER e DELETE.
- (E) SELECT, CREATE e TRUNCATE.

**25**

Um processo é considerado um conceito chave em qualquer Sistema Operacional, pois representa uma abstração de um programa em execução. Um processo pode passar por alguns estados durante o seu ciclo de vida. Sendo assim, um processo pode estar nos estados: (a) bloqueado, (b) pronto ou (c) em execução.

Nesse contexto, analise os itens a seguir.

- I. Para implementar o modelo de processos, o sistema operacional mantém uma tabela de processos, com uma entrada para cada um deles. Cada entrada, também conhecida como bloco de controle de processo (PCB – *Process Control Block*), contém informações relevantes sobre o processo. Um exemplo de informação contida no PCB é o estado do processo.
- II. Considerando um diagrama de estado do processo, não há transição a partir do estado “em execução”, para o estado “bloqueado”. Quando o sistema operacional descobre que o processo não pode continuar em execução, ele vai para o estado de “pronto”.
- III. O processo no estado “bloqueado”, não faz uso da CPU (Unidade Central de Processamento), mas está apto para ser executado (isto é, pronto para transitar ao estado “em execução”), aguardando para isso que o escalonador selecione-o para execução.

Está correto o que se afirma em:

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

**26**

*Multithreading* é um conceito fundamental para o melhor aproveitamento de recursos por um sistema operacional.

Para o emprego eficaz desse conceito, é fundamental a existência de mecanismos de sincronização eficientes.

Nesse contexto, analise as afirmativas a seguir.

- I. Por definição, semáforos possuem um contador, cujos valores podem ser 0, 1 ou 2.
- II. *Mutexes* são projetados para garantir que apenas uma *thread* possa acessar um recurso compartilhado por vez.
- III. *Mutexes* podem ser considerados uma generalização de semáforos, por conta da maior flexibilidade do contador de um *mutex*.

Está correto o que se afirma em

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

**27**

Em caso de perda ou roubo de um *notebook*, deseja-se que o respectivo sistema operacional ofereça proteção contra o acesso não autorizado aos seus dados.

Neste contexto, considerando um *notebook* com sistema operacional Windows 11 Pro, assinale a opção que indica a ferramenta nativa mais adequada para prover tal proteção.

- (A) User Account Control (UAC).
- (B) BitLocker.
- (C) Windows Security Essentials.
- (D) LUKS.
- (E) Gerenciador de Credenciais.

**28**

Sobre os recursos presentes em sistemas operacionais Windows, analise as afirmativas a seguir.

- I. No Windows 11, cada processo ou *thread* em execução tem associado a si um *token* de segurança, que carrega a informação sobre identificação e os privilégios do processo ou *thread*.
- II. O *Windows Sandbox* é recomendado para a execução de programas que o usuário considera suspeitos ou não confiáveis, pois oferece um ambiente isolado que impede que tais programas afetem o sistema operacional principal.
- III. O *Virtualization-Based Security (VBS)* é uma tecnologia que visa a executar uma parte do sistema operacional em ambiente virtualizado seguro, e tem como função oferecer acesso remoto ao sistema.

Está correto o que se afirma em

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

**29**

Uma das tecnologias mais utilizadas no Windows Server é o chamado *pool* de armazenamento (*storage pool*).

Essa tecnologia, baseada nos chamados *storage spaces*, consiste em

- (A) áreas de armazenamento designadas para armazenar os dados mais frequentemente acessados, de modo a aumentar a velocidade de acesso.
- (B) grupos de discos usados para consolidar *backups* de diversas máquinas virtuais, evitando perda de dados.
- (C) uma abstração dos discos físicos existentes em discos virtuais, permitindo uma gestão de armazenamento mais flexível.
- (D) uma coleção de serviços de armazenamento em nuvem, com o objetivo de melhorar a velocidade de recuperação de dados.
- (E) conjuntos de discos rígidos externos configurados para duplicar dados, de modo a garantir a integridade desses dados.

**30**

Relacione os módulos essenciais do Qt às suas respectivas definições ou funcionalidades.

1. Qt Core
2. Qt Quick
3. Qt QML
4. Qt D-Bus

- ( ) É uma linguagem de marcação declarativa e multiparadigma usada para criar interfaces de usuário de forma rápida e eficiente.
- ( ) Permite a troca de mensagens entre processos a nível de sistema e processos comuns dos usuários.
- ( ) Adiciona recursos ao C++ como comunicação contínua de objetos e árvores hierárquicas que organizam as propriedades de objetos de maneira natural com ponteiros protegidos.
- ( ) Disponibiliza tipos para receber os inputs dos usuários, construir componentes visuais, criar modelos de dados e instanciação atrasada de objetos.

Assinale a opção que indica a relação correta, segundo a ordem apresentada.

- (A) 4 – 3 – 2 – 1.
- (B) 1 – 4 – 3 – 2.
- (C) 3 – 4 – 1 – 2.
- (D) 2 – 1 – 4 – 3.
- (E) 3 – 2 – 1 – 4.

**31**

Um desenvolvedor de sistemas para operação de satélites necessita de conhecimentos de ferramentas e *frameworks* das mais variadas áreas, como gerenciamento de projetos, arquitetura de *software*, banco de dados, versionamento de código-fonte etc. Nesse contexto, uma ferramenta importante se destaca por interligar especificações de requisitos de alto nível com modelos de análise, projeto, implementação e testes de sistemas usando UML, SysML, BPMN e outros padrões abertos utilizados no mercado.

Assinale a opção que corresponde à ferramenta descrita no texto

- (A) Enterprise Architect.
- (B) VS Code.
- (C) Jenkins.
- (D) Kubernetes.
- (E) Apache Cassandra.

**32**

O *Visual Studio* é um ambiente integrado de desenvolvimento (IDE) desenvolvido pela Microsoft que suporta diversas linguagens de programação, tais como C#, Visual Basic, C++, F#, JavaScript, TypeScript, Python e outras. Um de seus aspectos mais relevantes é o robusto suporte para a construção, execução e depuração de testes unitários. Considere as seguintes afirmações sobre alguns recursos para testes unitários presentes no Visual Studio:

- I. *IntelliTest* permite gerar valores de entrada e saída para suas funções como pequenos blocos de testes, expandindo-os automaticamente quando seu código for incrementado, o que reduz o esforço para criação de testes unitários.
- II. *Live Unit Test* produz comentários sobre como as alterações de código afetam os testes existentes e se o novo código gerado é coberto por testes já criados.
- III. *CodeLens* permite a inspeção, execução, depuração e navegação direta para os testes unitários a partir do próprio editor de código.

Está correto o que se afirma em

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

**33**

Relacione as terminologias referentes ao Git, listadas a seguir, às suas respectivas definições.

1. Fetch
2. Branch
3. Issue
4. Commit

- ( ) Permite adicionar alterações do repositório remoto à sua ramificação de trabalho local, sem confirmá-las.
- ( ) Está contido no repositório, mas é uma versão paralela do mesmo, não afetando a ramificação principal e permitindo trabalhar livremente sem interromper a versão "live".
- ( ) É uma alteração individual em um ou mais arquivos. Quando usado para salvar um trabalho, o *Git* gera um ID único que permite registrar as alterações confirmadas assim como quem as fez e quando.
- ( ) É uma sugestão de melhoria, tarefa ou dúvida relacionada ao repositório. Pode ser criado por qualquer pessoa (em repositórios públicos), bem como ser categorizado com rótulos e atribuído a colaboradores.

Assinale a opção que indica a relação correta, segundo a ordem apresentada.

- (A) 2 – 1 – 4 – 3.
- (B) 1 – 4 – 2 – 3.
- (C) 2 – 1 – 3 – 4.
- (D) 1 – 2 – 4 – 3.
- (E) 3 – 2 – 4 – 1.



**34**

O protocolo IP (Internet Protocol) é um dos principais protocolos de comunicação em redes de computadores.

Com relação ao endereçamento IPv4 (IP versão 4), analise as afirmativas a seguir.

- I. Considerando a máscara de sub-rede 255.255.128.0, os endereços IPv4 192.168.91.0 e 192.168.48.10 estão na mesma sub-rede.
- II. A faixa de endereços IPv4 referente ao prefixo 124.201.0.0/18 vai de 124.201.0.0 a 124.201.63.255.
- III. Todas as interfaces de rede de um mesmo dispositivo possuem o mesmo endereço IPv4.

Está correto o que se afirma em

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

**35**

O protocolo TCP (*Transmission Control Protocol*) é responsável pelo controle da transmissão de um fluxo de dados em redes de computadores interligadas.

Com relação ao TCP, analise as afirmativas a seguir.

- I. O funcionamento correto do TCP requer o estabelecimento de uma conexão entre a origem e o destino final.
- II. O controle de fluxo é implementado por meio de um protocolo do tipo janela deslizante.
- III. Todas as conexões TCP são do tipo *full-duplex* e ponto a ponto.

Está correto o que se afirma em

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

**36**

As variáveis são elementos essenciais para o desenvolvimento de programas.

A respeito dos diferentes tipos de variáveis que podem ser usadas em um programa escrito em linguagem C, é correto afirmar que

- (A) *overflow* não pode ocorrer ao se realizarem operações aritméticas com variáveis do tipo real com representação em ponto flutuante.
- (B) os vetores possuem uma estrutura que permite armazenar uma quantidade pré-definida de variáveis de tipos distintos entre si.
- (C) todos os caracteres representados por uma variável do tipo *char*, utilizando codificação ASCII, podem ser impressos na tela.
- (D) o maior número que um inteiro sem sinal de 8 bits pode representar é 256.
- (E) as matrizes são armazenadas de forma contígua na memória.

**37**

O correto entendimento sobre os conceitos relacionados a operadores é essencial para o desenvolvimento de programas. Considere as seguintes instruções de parte de um programa desenvolvido em linguagem de programação C:

```
int x, y;  
x = 25;  
++x;  
y = x++;  
y+= x+13-11*2;
```

Os valores das variáveis x e y, após a execução dessas instruções acima, serão, respectivamente:

- (A) 27 e 704
- (B) 27 e 44
- (C) 26 e 43
- (D) 27 e 45
- (E) 28 e 703

**38**

A Notação Polonesa Reversa (RPN, do inglês *Reverse Polish Notation*) foi desenvolvida como uma forma de escrever expressões lógicas e aritméticas sem usar parênteses. Essa notação ganhou popularidade ao ser implementada em calculadoras científicas, onde permite reduzir a quantidade de acionamento de teclas no cálculo de expressões.

Quando uma calculadora opera no modo RPN, os operandos são inseridos previamente em uma estrutura de dados e, ao utilizar-se um operador (soma, subtração, ...), a quantidade de operandos necessários são retirados da estrutura na ordem inversa da inserção e, após o cálculo da operação, o resultado é inserido na estrutura de dados. Assim, por exemplo, caso se deseje calcular a expressão  $A + (B - C) * D$  em uma calculadora operando no modo RPN, pode-se seguir o seguinte procedimento:

- Insere A
- Insere B
- Insere C
- Realiza a operação de subtração
- Insere D
- Realiza a operação de multiplicação
- Realiza a operação de soma

De acordo com a descrição acima, assinale a opção que indica a estrutura de dados que melhor caracteriza a utilizada pelo modo RPN para armazenar os operandos e resultados.

- (A) Lista duplamente encadeada.
- (B) Lista encadeada circular.
- (C) Pilha.
- (D) Fila.
- (E) Árvore.

**39**

As estruturas de dados utilizadas em programação determinam como as informações serão armazenadas, organizadas e acessadas, sendo uma parte importante no projeto de *software*, com impacto no seu desempenho e eficiência.

Sobre estruturas de dados lineares, analise as afirmativas a seguir.

- I. Para realizar uma busca por um elemento em uma lista simplesmente encadeada pode-se começar a busca pelo início ou fim da lista.
- II. Listas duplamente encadeadas não permitem a exclusão de elementos que não sejam o último ou o primeiro elemento da lista.
- III. Uma lista circular pode ser simplesmente encadeada ou duplamente encadeada.

Está correto o que se afirma em

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

**40**

Considere um computador com processador de 32 *bits* cuja memória está organizada de acordo com o endereçamento de *byte*, com o método *big-endian*.

Suponha que as palavras, com tamanho de 4 *bytes*, 25 38 94 67H (identificada por P1) e 18 31 72 46H (identificada por P2) estão armazenadas a partir dos endereços 0010H e 002CH, respectivamente.

Nesse contexto, analise as afirmativas a seguir.

- I. A leitura do endereço 0014H retorna o *byte* 38H da palavra P1.
- II. A leitura do endereço 002FH retorna o *byte* 46H da palavra P2.
- III. Entre as palavras P1 e P2 podem ser armazenadas doze palavras.

Está correto o que se afirma em

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

**41**

Considerando o endereçamento de *byte* em arquitetura de computadores, e, os métodos *big-endian* e *little-endian* de ligação do processador à memória do computador, analise as afirmativas a seguir.

- I. No método *big-endian*, o endereçamento inicia pelo *byte* menos significativo.
- II. O acesso aos *bytes* de palavras armazenadas na memória independe do método de ligação (*big-endian* ou *little-endian*) para processadores de menos de 32 *bits*.
- III. Em ambos os métodos de ligação (*big-endian* e *little-endian*), os *bits* menos significativos do barramento de dados estão ligados aos *bits* menos significativos da palavra armazenada na memória.

Está correto o que se afirma em

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

**42**

Para que um programa de computador possa ser executado, diversas etapas compõem seu desenvolvimento. Com relação ao ciclo de desenvolvimento de um programa, analise as afirmativas a seguir.

- I. A etapa de compilação é responsável por gerar o arquivo executável do programa.
- II. Na etapa de link-edição, módulos pré-compilados são ligados entre si.
- III. Um programa compilado não causa erros de execução.

Está correto o que se afirma em

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

**43**

Com relação ao uso de um ambiente de programação integrado (IDE - *Integrated Development Environment*) no desenvolvimento de programas de computador, analise as afirmativas a seguir.

- I. As etapas de edição e de compilação de um programa de computador podem ser integradas em um IDE.
- II. O processo de link-edição de um programa não pode ser integrado em um IDE.
- III. Um programa desenvolvido em um IDE não requer o processo de depuração.

Está correto o que se afirma em

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

**44**

Com relação ao processo de depuração de um programa de computador, analise as afirmativas a seguir.

- I. Por meio da depuração, é possível investigar a ocorrência de erros no programa.
- II. A depuração é realizada antes da compilação.
- III. A depuração pode ser executada por meio de pontos de parada (*breakpoints*).

Está correto o que se afirma em

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

**45**

Os testes de regressão verificam se há efeitos indesejados nas alterações no software. Uma estratégia consiste em retestar tudo, isto é, reaplicar todos os casos de teste projetados para versões anteriores. Pode-se também utilizar técnicas seletivas, de forma que somente são reexecutados os casos de testes que tenham relação com as alterações realizadas. A seleção pode ser baseada no código, que escolhe casos de teste que exercitam uma porção do código modificado, ou baseada na especificação, que escolhe casos de teste exercitam a parte da especificação que foi alterada. Neste contexto, qual das considerações abaixo é verdadeira:

- (A) A técnica retesta tudo é mais simples de ser realizada e tem o mesmo custo de execução das técnicas seletivas.
- (B) As técnicas seletivas são mais custosas, pois podem reexecutar casos de teste que não podem mais ser executados na nova versão.
- (C) As técnicas seletivas com base na especificação somente executam testes desnecessários, pois as modificações sempre ocorrem no código.
- (D) As técnicas seletivas baseadas no código são mais fáceis de automatizar, pois ferramentas de análise de cobertura ajudam a determinar quais casos de teste devem ser selecionados.
- (E) As técnicas seletivas são intrinsecamente manuais, enquanto a técnica de retestar tudo é fácil de automatizar, pois basta reexecutar todos os testes.

Realização

