1ª Questão (Ref.: 202307627859) "Modelagem de sistema de software consiste na utilização de notações gráficas e textuais com o objetivo de construir modelos que representam as partes essenciais de um sistema, considerando-se várias perspectivas diferentes e complementares." BEZERRA, E. Princípios de análise e projeto de sistemas com UML. 3. ed. Rio de Janeiro: Elsevier, 2015. Ao modelar sistemas orientados a objetos, nos preocupamos em identificar os objetos que se relacionam com o domínio do problema do sistema em estudo. Nesse sentido, a UML oferece um conjunto de diagramas que permite modelar sistemas orientados a objeto. Sobre a UML, analise as assertivas: I. A UML é uma metodologia de desenvolvimento, que pode ser aplicada nas diferentes fases de um processo de desenvolvimento de software. II. A UML disponibiliza diagramas que permitem o entendimento do sistema sob diferes visões. III. Uma das visões, a de casos de uso, tem uma perspectiva externa, tal qual uma maquete de um empreendimento imobiliário. IV. A UML é dependente de tecnologia, podendo ser aplicada em contexto computacional específico. É correto apenas o que se afirma em: □ Le II. □ II e IV. ☐ III e IV. □ I. II e III. XII e III. 2ª Questão (Ref.: 202307606892) O desenvolvimento de sistemas envolve um conjunto de recursos que precisam ser geridos (planejados, executados e controlados). Por isso, é um processo complexo e que deve ser organizado em atividades ou, ainda, em fases. Independentemente do processo que se use, existem as fases em que são realizadas as ações numeradas de I a IV, a seguir. I. Análise dos requisitos para compreender o que o sistema deve fazer. II. Identificação dos requisitos. III. Implementação do sistema numa linguagem de programação. IV. Projeto do sistema, sua estrutura, arquitetura, comportamento e escolha das tecnologias (linguagem, SGBD

Assinale a alternativa que contém a ordem correta em que são executadas as ações no processo de desenvolvimento de

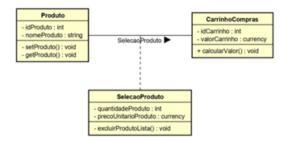
um sistema computacional:

☐ I, II, III, IV.
☐ II, I, III e IV.
☐ IV, III, II, I.
☑ II, I, IV e III.
☐ I, III, IV e II.

3ª Questão (Ref.: 202307735846) CESPE / CEBRASPE Órgão: MPC-PA Prova: CESPE - 2019 - MPC-PA - Analista Ministerial - Tecnologia da Informação
No uso da linguagem UML, o nível de visibilidade de um atributo que só pode ser acessado por classes declaradas dentro do mesmo pacote da classe a que pertence o atributo é denominado.
☐ privado. ☐ protegido. ☑ pacote. ☐ domínio. ☐ público.

4ª Questão (Ref.: 202307735845)

Fonte: Banca: INSTITUTO AOCP, Órgão: EMPREL, Prova: INSTITUTO AOCP - 2019 - EMPREL - Analista de Sistemas Considerando a seguinte figura, analise as assertivas e assinale a alternativa que aponta a(s) correta(s).

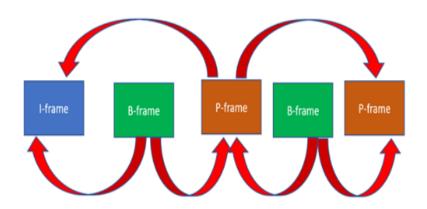


- I. A figura apresenta uma herança entre duas classes.
- II. A figura apresenta os atributos e métodos das classes envolvidas.
- III. O método calcular Valor () só pode ser acessado pela sua classe.
- ☐ Apenas I e II.
- ☐ Apenas II e III.
- ☐ Apenas I.
- X Apenas II.
- ☐ Apenas III.

5 ^a Questão (Ref.: 202308782097)			
O que é a visibilidade de um atributo no diagrama de classes?			
 ☒ A visibilidade de um atributo indica o seu nível de acesso. ☐ A visibilidade de um atributo indica a quantidade de acessos. ☐ A visibilidade de um atributo indica como o acesso acontece. ☐ A visibilidade de um atributo indica a forma de acesso. ☐ A visibilidade de um atributo indica o tipo de acesso. 			
6ª Questão (Ref.: 202308782049) Em que fases do desenvolvimento do software é usualmente construído o diagrama de atividades?			
☐ Análise e prototipação			
☐ Projeto e testes			
☐ Levantamento de requisitos e implementação			
Análise e projeto			
Levantamento de requisitos e testes			

7ª Questão (Ref.: 202307768876)

(INSTITUTO AOCP - EBSERH - 2017) Um recém-contratado analista é integrado a uma equipe de desenvolvimento de um projeto em andamento e lhe são apresentados os diagramas de modelagem. Entre todos esses diagramas, o diagrama a seguir é passado para esse analista reformular. Qual é o diagrama que ele terá que reformular?



- 🗷 Diagrama de Comunicação
- ☐ Diagrama de Pacote
- ☐ Diagrama de Estado
- ☐ Diagrama de Sequência

8ª Questão (Ref.: 202307816842)
(FAURGS –TJ-RS – 2018) Requisitos não funcionais - como o nome sugere - são requisitos que não estão diretamente relacionados com os serviços específicos oferecidos pelo sistema a seus usuários. Podem ser provenientes das características requeridas para o software, da organização que desenvolve o software ou de fontes externas. Os requisitos não funcionais que especificam ou restringem o comportamento do software - por exemplo, o seu desempenho, seus requisitos de proteção, seus requisitos de usabilidade e a taxa aceitável de falhas - são denominados requisitos:
ambientais
□ externos
■ de produto
□ éticos
□ organizacionais
9 ^a Questão (Ref.: 202308782282)
Suponha que o desenho de um gráfico de barras e de um gráfico de linhas possa ser implementado pelo mesmo conjunto de passos, sendo alguns comuns (escrever o título do gráfico e a legenda, por exemplo) e alguns específicos (desenhar os eixos e plotar os dados). Assinale a opção com o padrão de projeto que poderia ser aplicado nesse contexto:
☐ Factory Method
Strategy
Adapter

10ª Questão (Ref.: 202308782248)

☐ Facade

▼ Template Method

O trecho de código a seguir apresenta uma classe que calcula áreas de várias figuras geométricas. Para inserir uma nova figura, devemos modificá-la adicionando uma operação de cálculo de área correspondente à nova figura.

```
\cdot \texttt{public} \cdot \texttt{class} \cdot \texttt{CalculadoraGeometrica} \cdot \{\, \P
····public double obterArea (FiguraGeometrica figura) - { ¶
\cdots\cdots \text{if}\cdot (\texttt{figura}\cdot \texttt{instanceof}\cdot \texttt{Quadrado})\cdot \{\P
······return ·obterAreaQuadrado((Quadrado) ·figura);¶
·····} ·else ·if · (figura ·instanceof ·Triangulo) ·{¶
·····return ·obterAreaTriangulo ((Triangulo) ·figura);¶
·····} ·else ·{¶
·······return·0.0;¶
P { · · · · · · · ·
P {····
....¶
\cdots \cdot \texttt{private} \cdot \texttt{double} \cdot \texttt{obterAreaQuadrado} \, (\texttt{Quadrado} \cdot \texttt{quadrado}) \, \cdot \{\, \P
·····return ··quadrado.getLado() ·* ·quadrado.getLado();¶
· · · · } ¶
\cdots-private ·double ·obterAreaTriangulo (Triangulo ·triangulo) ·{¶
·····return ··triangulo.getBase() ·* ·triangulo.getAltura() ·/·2;¶
P { · · · ·
₽{
```

Assinale a alternativa que contenha o princípio SOLID que está sendo violado por essa construção de projeto:

S