



PROGRAMA DE RESIDÊNCIA EM TECNOLOGIA DA INFORMAÇÃO
JUSTIÇA FEDERAL NO RIO GRANDE DO NORTE

EDITAL 004/2019 – PROVA DE CONHECIMENTOS ESPECÍFICOS
ÁREA DE CONCENTRAÇÃO 1: DESENVOLVIMENTO DE SOFTWARE
24 / 02 / 2019

Identificação do Candidato	
Nome completo:	
CPF:	Assinatura:

Leia com atenção as seguintes instruções:

1. Aguarde a autorização do(s) fiscal(is) para poder iniciar a Prova.
2. Não esqueça de colocar seu **nome completo** (preferencialmente em letras maiúsculas) e de assinar o campo acima.
3. Este Caderno de Prova, com páginas numeradas de 1 a 11, é constituído de **30 (trinta) questões** de múltipla escolha, cada uma com quatro alternativas. Verifique se o Caderno de Prova está completo, sem falhas de impressão ou problemas que comprometam sua leitura. Caso necessário, solicite imediatamente ao(s) fiscal(is) a substituição do Caderno de Prova completo.
4. Confira se este Caderno de Prova corresponde à área de concentração para a qual foi inscrito. Caso haja alguma divergência, notifique imediatamente o(s) fiscal(is).
5. Leia com atenção o enunciado das questões antes de responde-las.
6. Cada questão possui **apenas uma** alternativa correta. Você deverá marcar a resposta que julgar correta usando caneta esferográfica de tinta na cor azul ou preta no local correspondente à respectiva questão. A interpretação das questões faz parte da Prova, de modo que não será permitida qualquer tipo de pergunta ou explicação ao(s) fiscal(is).
7. Não serão computadas questões sem marcação de resposta ou que contenham mais de uma marcação, marcação rasurada ou emendada.
8. Tenha cuidado ao manusear este Caderno de Prova, evitando rasuras, pois ele **não será substituído** por esse motivo. Também não é permitido destacar quaisquer das folhas que compõem este Caderno de Prova.
9. O tempo máximo para resolução desta Prova é de **2 (duas) horas**, para o qual **não haverá prorrogação**. Transcorrido esse tempo, o Caderno de Prova será recolhido pelo(s) fiscal(is).
10. Terminada a realização da Prova, este Caderno de Prova deverá ser **obrigatoriamente** entregue ao(s) fiscal(is) antes de se retirar da sala de realização.



QUESTÕES

1. Sobre os paradigmas de Programação Estruturada e Programação Orientada a Objetos, é incorreto afirmar que:
 - a) No paradigma estruturado, qualquer problema pode ser dividido em problemas menores, chamados de funções.
 - b) A linguagem de programação C é um exemplo de linguagem de programação estruturada, compilada e procedural.
 - c) O paradigma orientado a objetos entende o problema como um conjunto de objetos interagindo por meio de troca de mensagens.
 - d) A linguagem de programação C++ é um exemplo de linguagem orientada a objetos que não permite a utilização do paradigma estruturado na solução de um problema.

 2. A respeito das tecnologias Java/Java EE, assinale a alternativa correta.
 - a) EJB é uma arquitetura de componentes multiplataforma para o desenvolvimento de aplicações distribuídas, escaláveis e orientadas a objetos.
 - b) JTA é uma API da linguagem de programação Java que permite a componentes baseados em Java/Java EE criar, enviar, receber e ler mensagens.
 - c) JPA é uma especificação feita pela Sun que visa padronizar o uso de transações distribuídas feitas por aplicativos Java.
 - d) JMS é um *framework* para desenvolver aplicações Web de forma ágil, permitindo que o desenvolvedor crie interfaces de usuário através de um conjunto de componentes predefinidos.

 3. Marque a alternativa correta a respeito de padrões de projeto.
 - a) Cada padrão de projeto oferece código-fonte que mostra como utilizar bibliotecas de classes diretamente na aplicação.
 - b) Cada padrão de projeto busca resolver um problema recorrente de projeto de *software*, oferecendo uma solução concreta dentro de um contexto específico.
 - c) Todo padrão de projeto permite flexibilizar um determinado aspecto da aplicação.
 - d) Padrões de projeto promovem o reuso de código de *frameworks* existentes.

 4. Em grandes organizações, pessoas com diferentes papéis dentro da organização podem estar envolvidas com um banco de dados. Cada uma dessas pessoas, dependendo de seu papel, pode exercer atividades diferentes sobre o banco de dados. Nesse contexto, é correto afirmar que:
 - a) o projetista do banco de dados é responsável por identificar os dados a serem armazenados e por escolher estruturas apropriadas para representar e armazenar esses dados.
 - b) os usuários finais do banco de dados têm a capacidade de interagir diretamente com o banco de dados e de alterar sua estrutura interna, podendo assim, melhorar o seu acesso aos dados.
 - c) a responsabilidade de projetar e implementar ferramentas que facilitem a modelagem, projeto, criação e melhora de performance de um banco de dados é, principalmente, do administrador do banco de dados.
 - d) os projetistas de banco de dados são os principais responsáveis pelo monitoramento do banco de dados.
-



-
5. Acerca de Arquitetura Orientada a Serviços (SOA), é incorreto afirmar:
- a) SOA é um estilo arquitetural que preconiza que aplicações sejam fundamentalmente construídas através de serviços.
 - b) Em SOA, existem basicamente dois tipos de agentes de *software* trocando mensagens entre si, os provedores de serviços e os clientes que fazem uso de tais serviços.
 - c) SOA não é suficiente para promover interoperabilidade entre serviços, principalmente quando se tem aplicações desenvolvidas com diferentes tecnologias e plataformas e elas precisam comunicar-se umas com as outras.
 - d) SOA permite combinar serviços para fornecer funcionalidades agregadas, de mais alto nível.
6. Considerando o padrão de projeto *Singleton*, é correto afirmar que:
- a) O padrão *Singleton* pode ser usado para determinar uma quantidade fixa de instâncias de uma classe que podem ser criadas em uma aplicação.
 - b) A classe que implementa o padrão *Singleton* deve manter público seus construtores.
 - c) A implementação comum do padrão *Singleton* funciona em ambientes *multithreading*.
 - d) O padrão *Singleton* é implementado na linguagem Java, através de atributos estáticos e métodos não estáticos na classe onde é aplicado.
7. Qual das opções a seguir não apresenta um tipo de modelo de dados?
- a) Modelo XML
 - b) Modelo Relacional
 - c) Modelo Entidade-Relacionamento
 - d) Modelo Hierárquico
8. O projeto do *framework* Spring cujo objetivo é facilitar a criação de aplicações *stand-alone* é:
- a) *Spring SA*.
 - b) *Spring Data*.
 - c) *Spring Boot*.
 - d) *Spring Security*.
9. No que se refere à linguagem HTML:
- I. O elemento <head> é usado para a descrição do título do documento e pode ser subdividido em <h1>, <h2>, <h3>, <h4>, <h5> e <h6>.
 - II. O elemento <head> é usado para a descrição de informações complementares sobre o documento, como, por exemplo, código de caracteres do documento e língua na qual ele está escrito.
 - III. O elemento <head> é o local para colocar o cabeçalho de uma página no *layout* em HTML 5.
- Dentre as afirmações anteriores:
- a) apenas II está correta
 - b) apenas III está correta
 - c) I e II estão corretas
 - d) I e III estão corretas
-



10. São métodos do ciclo de vida de um componente do tipo *Service* da plataforma Android:

- a) onCreate, onStartCommand, onBind, onDestroy.
- b) onCreate, onBind, onResume, onPause, onStop.
- c) onCreate, onStartCommand, onSaveInstanceState, onService, onStop.
- d) onCreate, onStartCommand, onSaveInstanceState, onService.

11. Importante conceito do paradigma de Programação Orientada a Objetos, o encapsulamento de dados tem por objetivo ocultar detalhes de implementação de um determinado módulo. Em linguagens de programação que seguem esse paradigma, tais como Java, esse ocultamento é obtido fazendo com que todos os membros (atributos e métodos) em uma classe tenham um nível particular de visibilidade com relação às suas subclasses e às classes que acessam esses membros. No que se refere aos níveis de visibilidade, assinale a alternativa correta:

- a) Um membro público é visível a qualquer classe que acessa esse membro bem como à subclasse da classe à qual ele pertence.
- b) Um membro protegido é visível somente à classe à qual ele pertence, mas não às suas subclasses ou às classes que o acessam.
- c) Um membro privado é visível somente às subclasses da classe à qual ele pertence.
- d) Um método público pode acessar somente atributos públicos declarados na classe à qual ele pertence.

12. Ajax é:

- a) uma linguagem que facilita a construção de sistemas de conteúdos dinâmicos.
- b) um conjunto de objetos e métodos JavaScript que permitem acesso a recursos em servidores Web pelo protocolo HTTP.
- c) uma API JavaScript que permite o acesso ao DOM (modelo de objetos do documento).
- d) uma API Javascript para manipulação de objetos JSON.

13. Considere o seguinte trecho de código desenvolvido na linguagem de programação Java para a plataforma Android:

```
transaction.replace(R.id.fragment_container, newFragment);  
transaction.addToBackStack(null);  
transaction.commit();
```

Sobre esse código, é correto afirmar que:

- a) ocorrerá uma exceção e o aplicativo será fechado.
- b) o que estiver no fragment_container tornar-se-á nulo.
- c) o que estiver no fragment_container será colocado para *backstack*.
- d) o que estiver no fragment_container será substituído pelo newFragment.

Em um arquivo CSS, a referência aos elementos HTML pode ser feita por elementos, classes e identificadores. Para responder as questões 14 e 15, observe e analise o código CSS a seguir:



```
h2.right {  
    text-align: right;  
    color: blue;  
}  
#p1 {  
    text-align: left;  
    color: red;  
}  
p {  
    color:black  
}  
div {  
    color:black;  
}
```

14. Qual alternativa corresponde a um emprego correto de CSS em um trecho de HTML?

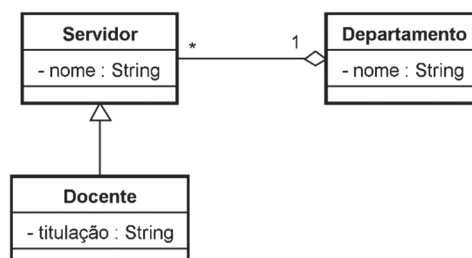
- a) <h2 id="right"> Minha página </h2>
- b) <h2 id="p1"> Minha página </h2>
- c) <h2 class="p1"> Minha página </h2>
- d) <p class="right"> Minha página </p>

15. Considere o seguinte trecho em HTML e indique as respectivas cores de *Cabeçalho*, *Texto1*, *Texto 2* após a aplicação do estilo em CSS:

```
<h2>Cabeçalho</h2>  
<div><p>Texto 1</p>  
<p id="#p1">Texto 2</p>  
</div>
```

- a) Vermelho, preto, azul
- b) Preto, preto, preto
- c) Azul, azul no fundo preto, vermelho no fundo preto
- d) Azul, preto, vermelho

16. Considere o seguinte Diagrama UML de Classes:



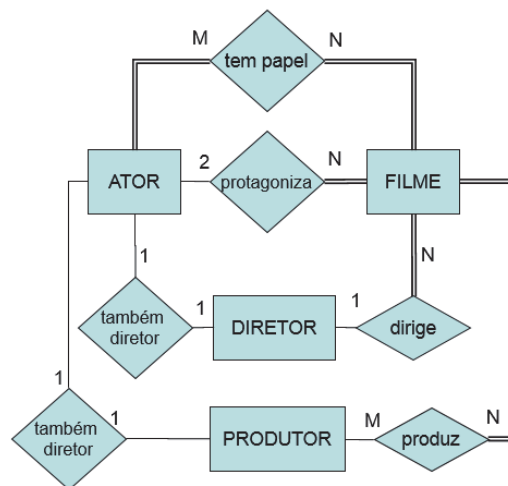
Com base no que está representado nesse diagrama, assinale a alternativa incorreta:

- a) Todo docente está associado a um departamento.
- b) Todo departamento tem ao menos um servidor.
- c) Um departamento pode ter nenhum servidor associado.



d) Um departamento pode ter nenhum docente associado.

17. Considere o seguinte Modelo Entidade-Relacionamento para um banco de dados de filmes:



Com relação a esse modelo, analise a seguinte lista de afirmações:

1. Nesse banco de dados, não existem atores/atrizes que não estiveram em nenhum filme.
2. Existem alguns atores/atrizes que atuaram em mais de dez filmes.
3. Alguns atores/atrizes foram protagonistas de vários filmes.
4. Todos os filmes têm dois protagonistas.
5. Todo(a) diretor(a) é ator/atriz de pelo menos um filme.
6. Nenhum(a) produtor(a) é um ator/atriz.
7. Um(a) produtor(a) não pode ser ator/atriz em um outro filme.
8. Alguns produtores foram também diretores.
9. Todos os filmes têm um diretor.
10. Nenhum filme pode ter um diretor(a) que também atuou neste mesmo filme.

Assuma ainda que foram inseridas nesse banco de dados informações quaisquer que respeitem esse modelo. Qual das seguintes opções apresenta todas as afirmações dessa lista as quais serão sempre verdade?

- a) 1, 4, 9
- b) 1, 2, 3
- c) 4, 9, 6
- d) 2, 3, 5

18. *JavaServer Faces* (JSF) é um *framework* baseado em componentes voltados à camada de apresentação de aplicações corporativas para a Web sobre a plataforma Java EE. Assinale a alternativa correta.

- a) O *framework* JSF implementa o padrão MVC (*Model-View-Controller*).
- b) As bibliotecas de *tags* (*taglibs*) são utilizadas para adicionar componentes EJB a páginas Web.
- c) Para preservar informações na navegação entre as páginas Web de uma aplicação, é possível



utilizar tanto um *managed bean* com escopo de visão (*view scoped*) quanto de sessão (*session scoped*), pois, em ambos os casos, os objetos gerenciados têm necessariamente o mesmo tempo de vida na aplicação.

- d) Apesar de o JSF disponibilizar validadores em sua biblioteca padrão, o *framework* também permite que o desenvolvedor utilize seus próprios validadores para tratar a entrada fornecida pelo usuário. No entanto, essa flexibilidade não se aplica aos conversores, de modo que o desenvolvedor se limita a utilizar unicamente os providos pelo JSF.

19. No contexto da UML (*Unified Modeling Language*), um relacionamento é uma ligação entre itens, podendo ser representado graficamente através de diferentes tipos de linhas. Associe os tipos de relacionamentos existentes na UML a suas respectivas descrições:

- | | |
|---|-------------------|
| I. É um relacionamento de utilização, determinando que um item usa as informações e/ou serviços de outro item, mas não necessariamente o inverso. | (a) Associação |
| II. É um relacionamento entre itens gerais e tipos mais específicos desses itens. | (b) Dependência |
| III. É um relacionamento estrutural que especifica objetos de um item conectados a objetos de outro item. A partir desse relacionamento, é possível navegar de um objeto de uma classe para um objeto de outra classe e vice-versa. | (c) Generalização |

Assinale a alternativa que contém a associação correta:

- a) I-a, II-b, III-c
- b) I-b, II-a, III-c
- c) I-b, II-c, III-a
- d) I-c, II-b, III-a

20. Uma empresa de auditoria foi contratada para analisar o banco de dados com o cadastro dos advogados registrados na Ordem dos Advogados do Rio Grande do Norte (OAB/RN). A primeira tarefa é encontrar os pares de advogados cadastrados que possuem o mesmo nome e números diferentes de inscrição na OAB para verificar possíveis fraudes. Considere que a tabela que armazena o cadastro dos advogados possui o número de inscrição como chave primária e as colunas referentes a nome, endereço, número de telefone e data de ingresso na Ordem. Assinale a alternativa que apresenta a consulta em SQL que busca corretamente os resultados, ordenados pelo nome dos advogados:

- a)

```
SELECT a1.nome, a1.inscricao, a2.inscricao
FROM advogados AS a1, advogados a2
WHERE a1.nome = a2.nome AND inscricao1 > inscricao2
GROUP BY nome
```
- b)

```
SELECT a1.nome, a1.inscricao, a2.inscricao
FROM advogados a1 JOIN advogados a2 ON a1.nome = a2.nome
WHERE a1.inscricao > a2.inscricao
ORDER BY a1.nome
```
- c)

```
SELECT a1.nome, a1.inscricao, a2.inscricao
FROM advogados a1
```

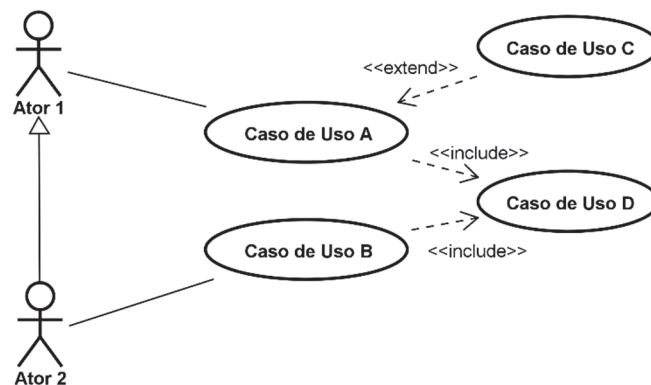



```
WHERE nome IN  
(SELECT nome FROM advogados a2 WHERE a1.nome = nome AND a1.inscricao > inscricao)  
ORDER BY nome
```

d)

```
SELECT * FROM advogados as a1, advogados AS a2  
WHERE a1.nome LIKE a2.nome AND inscricao1 > inscricao2  
GROUP BY a1.nome
```

21. Considere o seguinte Diagrama UML de Casos de Uso:



Com base no que está representando nesse diagrama, assinale a alternativa incorreta:

- a) O Ator 1 pode participar do Caso de Uso B.
- b) O Ator 1 pode participar do Caso de Uso D.
- c) O Ator 2 pode participar do Caso de Uso A.
- d) O Ator 2 pode participar do Caso de Uso C.

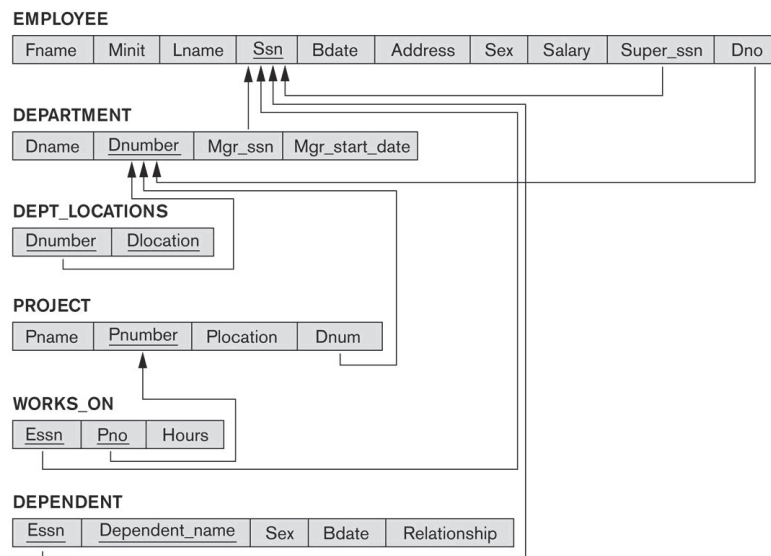
22. Analise as seguintes afirmativas acerca da estrutura de aplicações na plataforma Android:

- I. Uma *Task* corresponde a uma unidade de interação do usuário ou uma unidade de execução.
- II. Um *Intent* representa uma descrição abstrata de uma operação que uma atividade requer que outra desempenhe.
- III. Uma *Activity* representa uma cadeia de atividades que podem se estender por mais de um aplicativo.

Está(ão) correta(s) a(s) afirmativa(s):

- a) II, apenas.
- b) I e II, apenas.
- c) I e III, apenas.
- d) I, II e III.

23. Considere o seguinte modelo relacional:



Qual das opções abaixo apresenta a consulta que lista os primeiros nomes de todos os gerentes que não possuem dependentes?

- a) SELECT fname FROM EMPLOYEE WHERE
(ssn IN (SELECT mgr_ssn FROM DEPARTMENT)) AND
(ssn NOT IN (SELECT essn FROM DEPENDENT));
- b) SELECT fname FROM EMPLOYEE WHERE
(ssn IN (SELECT mgr_ssn FROM DEPARTMENT)) OR
(ssn NOT IN (SELECT essn FROM DEPENDENT));
- c) SELECT fname FROM EMPLOYEE WHERE
(ssn IN (SELECT mgr_ssn FROM DEPARTMENT)) OR
(ssn IN (NOT SELECT essn FROM DEPENDENT));
- d) SELECT fname FROM EMPLOYEE WHERE
(ssn NOT IN (SELECT mgr_ssn FROM DEPARTMENT)) AND
(ssn NOT IN (SELECT essn FROM DEPENDENT));

24. Uma empresa foi contratada para desenvolver aplicativos de uma instituição financeira que possam ser executados em diferentes plataformas, como PC, *smartphones* e *tablets*. Para evitar a necessidade de criar diferentes soluções para cada plataforma, a empresa decidiu adotar um padrão de projeto que defina uma família de componentes para cada plataforma e uma implementação que os instancie de acordo com a plataforma-alvo na qual a aplicação estará sendo executada. O padrão de projeto que seria adequado para essa implementação é o:

- a) *Adapter*
- b) *Template Method*
- c) *Abstract Factory*
- d) *Strategy*

25. Com o passar dos anos, as aplicações corporativas evoluíram em sua arquitetura, saindo de um modelo monolítico executado em computadores de grande porte para um modelo em duas camadas (*two-tier*) cliente-servidor e então para um modelo contendo no mínimo três camadas (*three-tier*). Essas camadas são:



- a) visualização, lógica e negócio
- b) apresentação, negócio e acesso a dados
- c) domínio, negócio e acesso a dados
- d) sistemas, processos e bancos de dados

26. Observe na figura a seguir a representação gráfica de uma classe através de um Diagrama UML de Classes:

Exemplo
- atributo1 : int # atributo2 : boolean
+ operação1() : float ~ operação2() : void

Na classe Exemplo, a visibilidade do atributo atributo2 é:

- a) protegida
- b) pública
- c) privada
- d) de pacote

27. Quanto aos tipos de relacionamentos do paradigma de Programação Orientada a Objetos, é correto afirmar que:

- a) a dependência determina que um item utiliza as informações e os serviços de outro item e vice-versa.
- b) a associação especifica que objetos de um item estão conectados a objetos de outro item.
- c) a especialização define uma relação entre itens gerais, chamados classes-filha (ou classes derivadas), e itens específicos, chamados classes-mãe (ou classes base), dos itens gerais.
- d) a generalização permite que os objetos da classe-filha (ou classe derivada) possam ser utilizados em qualquer local em que a classe-mãe (classe base) ocorra e vice-versa.

28. Dentre os Padrões GoF (*Gang of Four*), existem os padrões que descrevem maneiras comuns de organizar classes e objetos em um sistema e há aqueles que fornecem estratégias testadas para modelar a maneira como os objetos colaboram entre si em um sistema. Esses tipos de padrões são, respectivamente:

- a) estruturais e de concorrência.
- b) organizacionais e comportamentais.
- c) de criação e organizacionais.
- d) estruturais e comportamentais.

29. A respeito do modelo de componentes EJB, assinale a alternativa incorreta.

- a) No modelo de componentes EJB, existem três tipos básicos de *beans* de sessão, a saber, sem estado (*stateless*), com estado (*stateful*) e de instância única (*singleton*).
- b) Aplicações que fazem uso de componentes EJB precisam necessariamente ser executadas sobre um *container* que dê suporte às especificações da plataforma Java EE.
- c) Nas especificações atuais da plataforma Java EE, a implementação de um *bean* de sessão consiste



de uma classe implementada na linguagem de programação Java anotada como esse tipo de *bean* e que codifica métodos que realizam a lógica de negócio da aplicação.

d) A especificação JPA pode ser utilizada para o envio de mensagens utilizando *beans* orientados a mensagens.

30. No SGBD PostgreSQL, uma forma de melhorar o desempenho (*tuning*) quando da inserção de grande quantidade de registros em uma tabela é:

- a) criar uma nova tabela com o mesmo nome da tabela original e inserir os registros nessa nova tabela
- b) desabilitar o *autocommit* e executar apenas um *commit* ao final das inserções de registros
- c) realizar, obrigatoriamente, um *commit* a cada novo registro inserido
- d) realizar, simultaneamente a cada inserção de novo registro, o *backup* da tabela