



HOSPITAL DE
CLÍNICAS
PORTO ALEGRE RS

MISSÃO INSTITUCIONAL

Prestar assistência de excelência e referência com responsabilidade social, formar recursos humanos e gerar conhecimentos, atuando decisivamente na transformação de realidades e no desenvolvimento pleno da cidadania.

CADERNO DE QUESTÕES

EDITAL 06/2010 DE PROCESSOS SELETIVOS

PS 51 - ANALISTA DE SISTEMAS I

Fábrica de Aplicativos

Nome do Candidato: _____

Inscrição nº: _____ - _____



Fundação de Apoio da Universidade Federal do Rio Grande do Sul



HOSPITAL DE CLÍNICAS DE PORTO ALEGRE

EDITAL Nº 06/2010 DE PROCESSOS SELETIVOS

GABARITO APÓS RECURSOS

PROCESSO SELETIVO 51

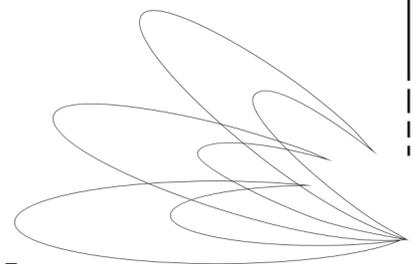
ANALISTA DE SISTEMAS I – Fábrica de Aplicativos

01.	A	11.	E	21.	C
02.	C	12.	C	22.	B
03.	B	13.	A	23.	E
04.	A	14.	E	24.	E
05.	D	15.	C	25.	B
06.	B	16.	C		
07.	A	17.	B		
08.	D	18.	B		
09.	C	19.	D		
10.	A	20.	B		

INSTRUÇÕES

- ❶ Verifique se este CADERNO DE QUESTÕES corresponde ao Processo Seletivo para o qual você está inscrito. Caso não corresponda, solicite ao Fiscal da sala que o substitua.
- ❷ Esta PROVA consta de **25** (vinte e cinco) questões objetivas.
- ❸ Caso o CADERNO DE QUESTÕES esteja incompleto ou apresente qualquer defeito, solicite ao Fiscal da sala que o substitua.
- ❹ Para cada questão objetiva, existe apenas **uma** (1) alternativa correta, a qual deverá ser assinalada com caneta esferográfica, de tinta azul, na FOLHA DE RESPOSTAS.
- ❺ Preencha com cuidado a FOLHA DE RESPOSTAS, evitando rasuras. Eventuais marcas feitas nessa FOLHA, a partir do número 26, serão desconsideradas.
- ❻ Durante a prova, não será permitida ao candidato qualquer espécie de consulta a livros, códigos, revistas, folhetos ou anotações, nem será permitido o uso de telefone celular, transmissor/receptor de mensagem ou similares e calculadora.
- ❼ Ao terminar a prova, entregue a FOLHA DE RESPOSTAS ao Fiscal da sala.
- ❽ A duração da prova é de **duas (2) horas e 30 (trinta) minutos**, já incluído o tempo destinado ao preenchimento da FOLHA DE RESPOSTAS. Ao final desse prazo, a FOLHA DE RESPOSTAS será **imediatamente** recolhida.
- ❾ O candidato somente poderá retirar-se do recinto da prova após transcorrida uma (1) hora do seu início.
- ❿ A desobediência a qualquer uma das recomendações constantes nas presentes instruções poderá implicar a anulação da prova do candidato

Boa Prova!



01. Sabendo-se que é um fator-chave para o processo de testes que se tenha conhecimento básico de seus tipos e de como devemos aplicá-los, considere as afirmações abaixo.

- I - O objetivo principal dos testes de caixa preta é garantir que todas as linhas de código e condições tenham sido executadas pelo menos uma vez, e que estejam corretas.
- II - Os testes de caixa branca são realizados para garantir que todos os requerimentos ou comportamentos da aplicação ou de um componente estejam corretos, concentrando-se assim nos requisitos funcionais do software.
- III- Os testes de caixa preta costumam ser aplicados durante as últimas etapas da atividade de teste, e os testes de caixa branca são aplicados diretamente pelo pessoal de desenvolvimento, visando garantir a qualidade do código enviado para testes.

Quais estão corretas?

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

02. Qualidade de software é algo que todos querem. Os gerentes sabem que eles precisam ter alta qualidade em seus trabalhos; desenvolvedores desejam produzir um produto de alta qualidade; os usuários, por sua vez, esperam que o trabalho através do uso do software seja confiável e consistente. Considere, a esse respeito, as seguintes afirmações.

- I - Qualidade demanda especificação de requerimentos e suficiente detalhamento deles. Deve-se permitir que os requerimentos sejam mensurados, de modo a medir o esforço a ser gasto e o nível de detalhamento real exigido.
- II - Softwares desenvolvidos com alto grau de qualidade não apresentam defeito, não sendo, portanto, necessário definir níveis de aceitação do produto.
- III- A verificação prova que o produto vai ao encontro dos requerimentos especificados nas atividades executadas durante o desenvolvimento do produto, enquanto a validação checa se o sistema vai ao encontro dos requerimentos do consumidor. É uma prática comum combinar verificação com validação no processo de testes.

Quais estão corretas?

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

03. Considere as afirmações abaixo a respeito do desenvolvimento de softwares.

- I - Planejamento e ferramentas de apoio são mais importantes que indivíduos e interações.
- II - O objetivo é chegar ao final do desenvolvimento com uma documentação abrangente do software.
- III- É mais importante responder a mudanças do que seguir um plano.
- IV- Colaboração com o cliente é mais importante do que negociação de contratos.

Quais delas se relacionam com o Manifesto para Desenvolvimento Ágil de Software?

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

04. Considere as afirmativas abaixo a respeito do Manifesto Ágil de Software.

- I - Pessoas de negócio e desenvolvedores devem trabalhar diariamente em conjunto por todo o projeto.
- II - Após o início da construção, devem ser recusadas mudanças nos requisitos, visando manter o escopo e ter uma visão mais exata possível da entrega.
- III- O método mais eficiente e eficaz de transmitir informações para e entre uma equipe de desenvolvimento é através de documentação clara, para que todas as decisões fiquem registradas e possam posteriormente ser consultadas.

Quais delas são princípios desse Manifesto?

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

05. Na UML, qual é o diagrama que tem como foco a ordem temporal das mensagens?

- (A) Diagrama de Caso de Uso.
- (B) Diagrama de Tempo.
- (C) Diagrama de Eventos.
- (D) Diagrama de Sequência.
- (E) Diagrama de Gráficos de Estados.

06. Na tecnologia de objetos, um padrão é uma descrição nomeada de um problema e sua solução, que pode ser utilizada em novos contextos, com conselhos sobre como utilizá-las em novas situações. Portanto, o padrão é um veículo para apresentar e lembrar princípios úteis de engenharia de software. Dentro dos padrões de projeto de software existentes e conhecidos, o que descrevem os padrões GRASP?

- (A) Princípios de materialização sob demanda em uma hierarquia de composição.
- (B) Princípios fundamentais de atribuição de responsabilidades a objetos.
- (C) Como utilizar um conjunto de classes extensíveis que forneçam serviços para objetos persistentes.
- (D) Como melhorar o desempenho e suporte a operações de gerenciamento de transações.
- (E) Princípios fundamentais de atribuição de segurança a objetos.

07. Assinale a alternativa que preenche, correta e respectivamente, as lacunas do texto abaixo.

Uma classe com _____ não é dependente de muitas outras classes. Uma classe com responsabilidades altamente relacionadas e que não executa um formidável volume de trabalho tem _____.

- (A) acoplamento baixo – coesão alta
- (B) coesão baixa – acoplamento alto
- (C) acoplamento alto – coesão alta
- (D) coesão baixa – acoplamento fraco
- (E) acoplamento baixo – acoplamento alto

08. Considerando que as afirmações abaixo referem-se a motivos que justificam a utilização de uma arquitetura multicamadas para sistemas de informação, assinale-os com **V** (para verdadeiro) ou **F** (para falso).

- () Aumentar o reaproveitamento entre classes DAO e a camada de interface no mapeamento objeto-relacional.
- () Isolar a lógica da aplicação em componentes separados que podem ser reutilizados em outros sistemas.
- () Favorecer o acesso da camada de interface ao banco de dados sem ser necessário verificações em outras camadas.
- () Alocar desenvolvedores para a construção de camadas específicas, tal como ter uma equipe trabalhando exclusivamente para a camada de apresentação.
- () Possibilitar distribuição de camadas em diferentes nós físicos de processamento e/ou diferentes processos.

A sequência correta de preenchimento dos parênteses, de cima para baixo, é

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

09. Como se chama um conjunto de classes reutilizáveis – e usualmente extensíveis – que fornece serviços para objetos que devem ser armazenados em um banco de dados?

- (A) Interface.
- (B) SOA.
- (C) Framework de Persistência.
- (D) Mapeamento Objeto-relacional.
- (E) Camada de Negócio.

10. Assinale a alternativa que preenche, correta e respectivamente, as lacunas do texto abaixo.

Java permite que vários métodos com o mesmo nome sejam definidos, contanto que esses métodos tenham conjuntos diferentes de parâmetros. Esse recurso é chamado de _____. O compilador Java seleciona o método adequado examinando a _____ na chamada. Esta funcionalidade da linguagem é realizada através da combinação _____.

- (A) sobrecarga de método – assinatura – do nome do método, dos tipos e dos parâmetros
- (B) assinatura do método – sobrecarga de método – do nome do método e dos parâmetros
- (C) polimorfismo – superclasse do objeto utilizado – de API's específicas
- (D) reutilização de código – visibilidade do método – do nome do método, dos tipos e dos parâmetros
- (E) polimorfismo – assinatura – de API's específicas

11. A exceção é uma indicação de que ocorreu um problema durante a execução do programa, o qual precisa ser tratado. Esse tratamento deve ser utilizado:

- I - em projetos grandes, para tratar exceções de uma maneira uniforme em todo o projeto;
- II - para processar situações excepcionais em que um método é incapaz de completar sua tarefa por razões que não se podem controlar;
- III- para processar exceções de componentes do programa que não são projetados para tratar essas exceções diretamente.

Quais estão corretas?

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

12. Assinale as afirmações abaixo com **V** (verdadeiro) ou **F** (falso) no que se refere a classes e atributos.

- () Uma classe é uma descrição de um conjunto de objetos que compartilham os mesmos atributos, operações, relacionamentos e semântica.
- () Um atributo é uma propriedade nomeada de uma classe que descreve um intervalo de valores que as instâncias da propriedade podem apresentar. Uma classe pode ter qualquer número de atributos, porém deve ter ao menos um atributo.
- () Quando utilizamos herança para criar uma nova classe a partir de uma classe existente, a nova classe herda os atributos e comportamentos dessa classe existente. Dessa forma, é possível adicionar novos atributos e comportamentos para a nova classe, mas não é permitido sobrescrever comportamentos da superclasse.
- () Uma interface é uma coleção de operações utilizadas para especificar um serviço de uma classe ou componente. Assim como as classes, uma interface poderá ter qualquer número de operações.

A sequência correta de preenchimento dos parênteses, de cima para baixo, é

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

13. Considerando-se que os modificadores de acesso a membro *public* e *private* controlam o acesso às variáveis de instância e aos métodos de uma classe, assinale a afirmação correta.

- (A) A ação de restringir o acesso aos membros de uma classe através da palavra-chave *private* é chamada de encapsulamento.
- (B) Uma classe *private* não poderá ser utilizada nem mesmo pelas classes pertencentes ao mesmo pacote.
- (C) A visibilidade de um método é um detalhe irrelevante, visto que estas informações não interferem na utilização do mesmo em qualquer parte do sistema.
- (D) Ao especificar a visibilidade das características de um classificador, geralmente deseja-se exibir todos seus detalhes de implementação e exibir as características necessárias para a execução.
- (E) Tornar *private* as variáveis de instância de uma classe e os seus métodos facilita a depuração, uma vez que os problemas com manipulações de dados estão localizados nos métodos da classe.

14. No que se refere a aspectos do banco de dados, considere as afirmações abaixo.

- I - A cláusula WHERE seleciona as linhas de entrada antes dos grupos e agregações a serem computados, enquanto HAVING seleciona linhas de grupo após os grupos e agregações serem computados.
- II - O comando ANALYZE coleta estatísticas sobre o banco de dados.
- III- OPEN, FETCH e CLOSE são instruções que operam sobre cursores.

Quais estão corretas?

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

15. Segundo a documentação do Postgresql 8.2.0, em qual das situações abaixo é correto utilizar a palavra-chave ONLY?

- (A) Para identificar linhas duplicadas em uma tabela.
- (B) Para identificar qual coluna tem dados únicos.
- (C) Para efetuar a consulta apenas na tabela identificada no SELECT, sem incluir as tabelas abaixo desta na hierarquia de herança.
- (D) Para eliminar linhas duplicadas em um resultado de um SELECT.
- (E) Para contar linhas resultantes de uma *query*.

16. Uma das limitações do modelo E-R é sua incapacidade de expressar relacionamentos entre relacionamentos. A solução dessa limitação encontra-se no uso de

- (A) generalização.
- (B) especialização.
- (C) agregação.
- (D) combinação.
- (E) tabelas redundantes.

17. Considere as afirmações abaixo no que se refere ao framework JBoss Seam.

- I - Implementa a especificação JEE.
- II - Simplifica e estende JEE.
- III - É uma alternativa à plataforma JEE.
- IV - É um servidor de aplicação.

Quais estão corretas?

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

18. O framework Seam adiciona um novo conceito às aplicações web baseadas em Servlets, JSP e/ou JSF, chamado **conversação**. Assinale a afirmação correta a respeito desse novo conceito.

- (A) Uma conversação só é finalizada quando expira o seu tempo de Timeout.
- (B) Uma conversação aninhada (*nested*) tem apenas acesso de leitura aos dados de contexto armazenados em sua conversação pai.
- (C) Uma conversação explícita (*long-running*) só pode ser iniciada através da anotação @Begin em um método *action listener*.
- (D) Uma conversação implícita (*temporary*) pode ser iniciada através da anotação @Begin em um método *action listener*.
- (E) Não existem conversações temporárias; todas as requisições serão executadas dentro de uma conversação de longa duração que finalizará apenas quando for solicitado explicitamente.

19. Sabendo-se que, na estrutura de coleções Java (Java Collections Framework), algumas classes de coleção permitem elementos duplicados e outras não, considere as afirmações abaixo.

- I - HashSet implementa a interface Set e não permite elementos duplicados.
- II - ArrayList implementa a interface List e permite elementos duplicados.
- III- TreeSet não implementa a interface Set e permite elementos duplicados.

Quais estão corretas?

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

20. Quanto ao controle de acesso a classes ou a seus membros, assinale a alternativa **INCORRETA**.

- (A) Métodos com o modificador de acesso protected só podem ser acessados por classes que herdarem esses métodos ou por classes que residem no mesmo pacote.
- (B) Métodos com nível de acesso default são visíveis por qualquer classe de qualquer pacote.
- (C) Classes com nível de acesso default são visíveis apenas dentro de seu próprio pacote.
- (D) Uma subclasse não herda os atributos ou métodos privados da superclasse.
- (E) Uma classe só pode ser privada se for uma classe interna.



21. Qual a tag HTML correta para forçar uma quebra de linha?

- (A) <lb />
- (B) <linefeed />
- (C)

- (D) <break />
- (E) <tr />

22. Em JPA, qual anotação deve ser utilizada para que uma classe seja uma entidade persistente?

- (A) javax.persistence.Embeddable
- (B) javax.persistence.Entity
- (C) javax.persistence.Table
- (D) javax.persistence.Persistent
- (E) javax.persistence.Anotation

23. Assinale as afirmativas abaixo com **V** (verdadeiro) ou **F** (falso), no que se refere ao modelo de componentes do JBoss Seam.

- () A anotação @In é usada para injetar um componente Seam em outro componente Seam.
- () Todos os componentes Seam são instanciados na inicialização da aplicação.
- () Componentes Seam podem ser configurados utilizando-se uma anotação de código @Name ou uma entrada de componente no arquivo components.xml.
- () Seam não fornece serviços de gestão do ciclo de vida de seus componentes.

A sequência correta de preenchimento dos parênteses, de cima para baixo, é

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

24. Considere as declarações abaixo.

I - @In
private SomeBean bean;

II - Supondo que exista um componente Seam chamado "product" com uma propriedade chamada details, é permitido usar expressões como em @In(value="#{product.details}")
private SomeBean bean;

III- @In(create=true)
public SomeBean getBean(){ return bean; }

Quais dessas declarações são válidas na injeção de componentes Seam?

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

25. Considere o código abaixo.

```
public class Equals {  
  
    public static void main(String[] args) {  
        String s1 = "abc";  
        String s2 = new String("abc");  
  
        if (s1 == s2)  
            System.out.print("== true");  
        else  
            System.out.print("== false");  
  
        if (s1.equals(s2))  
            System.out.print(" equals true");  
        else  
            System.out.print(" equals false");  
    }  
}
```

Ao compilar e executar o código descrito acima, o que será impresso?

- (A) == true equals true
- (B) == false equals true
- (C) == true equals false
- (D) == false equals false
- (E) Nada será impresso, pois o código não compila.