HOSPITAL DE CLÍNICAS
PORTO ALEGRE RS

# HOSPITAL DE CLÍNICAS DE PORTO ALEGRE

# EDITAL N.º 01/2016 DE PROCESSOS SELETIVOS

# GABARITO APÓS RECURSOS

## PROCESSO SELETIVO 05

## ANALISTA DE TI I
## (Desenvolvimento)

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 01. | **A** | 11. | **A** | 21. | **E** | 31. | **D** |
| 02. | **D** | 12. | **B** | 22. | **C** | 32. | **E** |
| 03. | **C** | 13. | **A** | 23. | **D** | 33. | **A** |
| 04. | **B** | 14. | **B** | 24. | **B** | 34. | **D** |
| 05. | **E** | 15. | **C** | 25. | **C** | 35. | **E** |
| 06. | **E** | 16. | **D** | 26. | **C** | 36. | **A** |
| 07. | **A** | 17. | **E** | 27. | **E** | 37. | **B** |
| 08. | **C** | 18. | **A** | 28. | **E** | 38. | **E** |
| 09. | **E** | 19. | **D** | 29. | **E** | 39. | **B** |
| 10. | **D** | 20. | **D** | 30. | **B** | 40. | **E** |

CADERNO DE QUESTÕES

HOSPITAL DE CLÍNICAS
PORTO ALEGRE RS

# EDITAL Nº 01/2016
## DE PROCESSOS SELETIVOS (PS)

***MISSÃO***

*Ser um referencial público em saúde, prestando assistência de excelência, gerando conhecimento, formando e agregando pessoas de alta qualificação.*

# PS 05 - ANALISTA DE TI I
# (Desenvolvimento)

| MATÉRIA | QUESTÕES | PONTUAÇÃO |
|---|---|---|
| Conhecimentos Específicos | 01 a 40 | 0,25 cada |

FAURGS
Fundação de Apoio da Universidade Federal do Rio Grande do Sul



**Nome do Candidato:** _____

**Inscrição nº:** ___ ___ ___ ___ ___ ___ – ___

FAURGS
Fundação de Apoio da Universidade Federal do Rio Grande do Sul

# INSTRUÇÕES

1. 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.

2. Esta PROVA consta de **40** (quarenta) questões objetivas.

3. Caso o CADERNO DE QUESTÕES esteja incompleto ou apresente qualquer defeito, solicite ao Fiscal da sala que o substitua.

4. Para cada questão objetiva, existe apenas **uma** (1) alternativa correta, a qual deverá ser assinalada na FOLHA DE RESPOSTAS.

5. Os candidatos que comparecerem para realizar a prova **não deverão portar** armas, malas, livros, máquinas calculadoras, fones de ouvido, gravadores, *pagers, notebooks*, **telefones celulares**, *pen drives* ou quaisquer aparelhos eletrônicos similares, nem utilizar véus, bonés, chapéus, gorros, mantas, lenços, aparelhos auriculares, prótese auditiva, óculos escuros, ou qualquer outro adereço que lhes cubra a cabeça, o pescoço, os olhos, os ouvidos ou parte do rosto. **Os relógios de pulso serão permitidos, desde que permaneçam sobre a mesa, à vista dos fiscais, até a conclusão da prova.** (conforme subitem 7.10 do Edital de Abertura)

6. **É de inteira responsabilidade do candidato comparecer ao local de prova munido de caneta esferográfica preferencialmente de tinta azul, de escrita grossa, para a adequada realização de sua Prova Escrita. Não será permitido o uso de lápis, marca textos, lapiseira/grafite e/ou borracha durante a realização da prova.** (conforme subitem 7.16.2 do Edital de Abertura)

7. Não serão permitidos: nenhuma espécie de consulta em livros, códigos, revistas, folhetos ou anotações, nem o uso de instrumentos de cálculo ou outros instrumentos eletrônicos, exceto nos casos estabelecidos no item 13 do Edital. (conforme subitem 7.16.3 do Edital de Abertura)

8. Preencha com cuidado a FOLHA DE RESPOSTAS, evitando rasuras. Eventuais marcas feitas nessa FOLHA a partir do número **41** serão desconsideradas.

9. Ao terminar a prova, entregue a FOLHA DE RESPOSTAS ao Fiscal da sala.

10. A duração da prova é de **três horas e trinta minutos (3h30min)**, já incluído o tempo destinado ao preenchimento da FOLHA DE RESPOSTAS. Ao final desse prazo, a FOLHA DE RESPOSTAS será **imediatamente** recolhida.

11. **O candidato somente poderá se retirar da sala de prova uma hora (1h) após o seu início. Se quiser levar o Caderno de Questões da Prova Escrita Objetiva, o candidato somente poderá se retirar da sala de prova uma hora e meia (1h30min) após o início. O candidato não poderá anotar/copiar o gabarito de suas respostas de prova.**

12. **Após concluir a prova e se retirar da sala de prova, o candidato somente poderá se utilizar de sanitários nas dependências do local de prova, se for autorizado pela Coordenação do Prédio e estiver acompanhado de um fiscal.** (conforme subitem 7.16.6 do Edital de Abertura)

13. Ao concluir a Prova Escrita, o candidato deverá devolver ao fiscal da sala a Folha de Respostas (Folha Óptica). Se assim não proceder, será excluído do Concurso. (Conforme subitem 7.16.8 do Edital de Abertura)

14. A desobediência a qualquer uma das recomendações constantes nas presentes instruções poderá implicar a anulação da prova do candidato.

HOSPITAL DE CLÍNICAS
PORTO ALEGRE RS

**Boa prova!**

**01.** O padrão de projeto *strategy* pode ser utilizado quando se necessita

(A) de variantes de um algoritmo.
(B) deixar visível aos usuários estruturas de dados complexas.
(C) isolar a aplicação da implementação da classe concreta.
(D) reutilizar classes com interfaces inicialmente incompatíveis.
(E) usar uma classe existente que não corresponde à interface requerida.

**02.** Considere as afirmações abaixo sobre as estratégias de herança no Hibernate.

I - A herança simples pode ser mapeada para uma tabela única por hierarquia de classe, sendo que uma única tabela armazena todas as instâncias de uma hierarquia de classes.
II - A herança múltipla pode ser mapeada para tabelas distintas tanto para a superclasse quanto para as subclasses.
III- A herança simples pode ser mapeada para uma tabela por classe concreta. Cada tabela armazena as propriedades da classe e de suas superclasses, isto é, o estado da entidade é armazenado na tabela inteiramente dedicada para a sua classe.

Quais estão corretas?

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

**03.** Considere o trecho de código abaixo, escrito em Hibernate.

```
@Entity
public class Troop {
   @OneToMany(mappedBy="troop")
   public Set<Soldier> getSoldiers() {
   ...
}

@Entity
public class Soldier {
   @ManyToOne
   @JoinColumn(name="troop_fk")
   public Troop getTroop() {
   ...
}
```

Com base nesse código, é correto afirmar que

(A) *Troop* tem uma associação bidirecional de muitos para muitos com *Soldier* através da propriedade *troop* .
(B) a associação bidirecional muitos para muitos é definida logicamente usando a anotação *@ManyToOne.*
(C) *Troop* tem uma associação bidirecional de um para muitos com *Soldier* através da propriedade *troop*.
(D) o nome da associação bidirecional é definido pela anotação *@JoinColumn*.
(E) a obrigatoriedade do armazenamento lógico é definida pela anotação *mappedBy.*

**04.** Qual é a anotação no Hibernate que permite marcar uma propriedade como identificador?

(A) @Entity
(B) @Id
(C) @Identifier
(D) @Person
(E) @Key

**05.** Qual é o método de PL/SQL que retorna o tamanho máximo de uma coleção?

(A) MAXIMUM
(B) UTMOST
(C) ALL
(D) COUNT
(E) LIMIT

**06.** Considere as afirmações abaixo sobre a estrutura de um bloco PL/SQL.

I - A seção declarativa é uma seção opcional e deve ser utilizada somente quando variáveis, cursores, exceções e/ou tipos construídos forem requeridos na resolução do problema.

II - A seção executável é obrigatória na construção de um bloco e compreende o conjunto de instruções para a resolução do problema do bloco que será construído.

III- A seção de tratamento de exceções é opcional e deve ser utilizada somente quando houver necessidade de tratamento de erros que podem ocorrer durante a execução do bloco.

Quais estão corretas?

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

**07.** Qual é a variável de PL/SQL que armazena blocos grandes de dados com caracteres de um único *byte* no banco de dados?

(A) CLOB
(B) BLOB
(C) LOB
(D) BFILE
(E) DLOB

**08.** O resultado de duas consultas pode ser combinado na linguagem PostgreSQL através da seguinte sintaxe:

*comando1* UNION [ALL] *comando2*
*comando1* INTERSECT [ALL] *comando2*
*comando1* EXCEPT [ALL] *comando2*

Com base nessa sintaxe, é correto afirmar que

(A) UNION anexa o resultado do *comando2* no resultado do *comando1*, na ordem em que as linhas são retornadas pela consulta.
(B) UNION ALL anexa o resultado do *comando2* no resultado do *comando1*, eliminando as linhas duplicadas.
(C) INTERSECT retorna todas as linhas presentes tanto no resultado do *comando1* quanto no resultado do *comando2*, eliminando as linhas duplicadas.
(D) INTERSECT ALL retorna todas as linhas presentes no resultado do *comando1* que não estão presentes no resultado do *comando2*.
(E) EXCEPT ALL retorna as linhas que estão no resultado do *comando2*, mas não estão no resultado do *comando1*, mantendo as linhas duplicadas.

**09.** Analise a sintaxe para gatilhos do PostgreSQL.

```
CREATE TRIGGER nome { BEFORE | AFTER } { evento [ OR ... ] }
    ON tabela [ FOR [ EACH ] { ROW | STATEMENT } ]
    EXECUTE PROCEDURE nome_da_função ( argumentos )
```

A respeito dessa sintaxe, assinale a alternativa que apresenta a afirmação correta.

(A) O comando CREATE TRIGGER cria um gatilho que fica associado a todas as tabelas do banco de dados e executa a função especificada *nome_da_função* quando ocorre uma nova inserção no banco de dados.
(B) Um gatilho que está marcado com FOR EACH STATEMENT é chamado uma vez para cada linha que a operação modifica.
(C) Um gatilho que está marcado com FOR EACH ROW é chamado uma única vez para uma determinada operação, não importando quantas linhas sejam modificadas.
(D) Se existirem vários gatilhos do mesmo tipo, definidos para o mesmo evento, estes serão disparados em ordem crescente de especificação no banco de dados.
(E) Se BEFORE for utilizado, o gatilho pode fazer com que a operação não seja realizada para a linha corrente ou pode modificar a linha que está sendo inserida (para as operações de INSERT e UPDATE somente).

**10.** Considere a tabela relacional abaixo.

EMPREGADO_PROJETO
(num_empregado, num_projeto, horas_trabalhadas, nome_empregado, nome_projeto, localização_projeto)

Nessa tabela, não se pode inserir um projeto, a menos que um empregado esteja associado. Por outro lado, não se pode inserir um empregado, a menos que esteja associado a um projeto. Esse é um exemplo de

(A) *deadlock*.
(B) anomalia de exclusão.
(C) atualização fantasma.
(D) anomalia de inserção.
(E) atualização postergada.

**11.** Uma relação está na _____ se todos os seus atributos são monovalorados e atômicos.

Assinale a alternativa que completa, corretamente, a lacuna da afirmação acima.

(A) Primeira Forma Normal
(B) Segunda Forma Normal
(C) Terceira Forma Normal
(D) Quarta Forma Normal
(E) Quinta Forma Normal

**12.** Assinale a alternativa que apresenta afirmação correta sobre o uso de um atributo NULL.

(A) Para uma entidade *Pessoa*, o valor do atributo *idade* pode ser determinado pela data atual (hoje) e o valor do atributo *data_nascimento* dessa Pessoa.
(B) O atributo *numero_apartamento* de um endereço só se aplica a endereços que estão em prédios de apartamento e não em outros tipos de residências, como casas.
(C) O atributo *cores* para um entidade *Carros* pode ter um valor único quando o carro tem uma cor ou mais de um valor quando o carro tem várias cores.
(D) O atributo *numero_funcionarios* de uma entidade *Departamento* pode ser derivado contando-se o número de funcionários trabalhando para este departamento.
(E) O atributo *formação_academica* para uma entidade *Pessoa* pode conter um valor único ou dois ou mais valores quando a pessoa tem mais de uma formação.

**13.** Um algoritmo de ordenação é executado através dos seguintes passos: (I) escolha de um elemento da lista, denominado pivô; (II) rearranjo da lista, de forma que todos os elementos anteriores ao pivô sejam menores do que ele e que todos os elementos posteriores ao pivô sejam maiores do que ele; e, também, de modo que o pivô, ao fim do processo, esteja em sua posição final, havendo duas sublistas não ordenadas; (III) ordenação recursiva das sublistas dos elementos menores e dos elementos maiores. Que algoritmo é esse?

(A) Quick Sort
(B) Merge Sort
(C) Bubble Sort
(D) Insertion Sort
(E) Selection Sort

**14.** Editores de Texto geralmente oferecem um mecanismo de reversão de operações (*undo*) que cancela operações recentes e reverte um documento a estados anteriores. A operação de reversão é implementada mantendo as alterações na estrutura de dados

(A) fila.
(B) pilha.
(C) *heap*.
(D) *hash*.
(E) árvore.

**15.** Analise o trecho de código abaixo, escrito em HTML5.

```
interface HTMLIFrameElement : HTMLElement {
          attribute DOMString src;
          attribute DOMString srcdoc;
          attribute DOMString name;
 [PutForwards=value] readonly attribute DOMSet-
tableTokenList sandbox;
          attribute DOMString width;
          attribute DOMString height;
readonly attribute Document? contentDocument;
readonly attribute WindowProxy? contentWindow;
};
```

Com base nesse código, é correto afirmar que

(A) *src* é o documento para renderizar o *iframe*.
(B) *iframe* é a regra de segurança para o conteúdo aninhado.
(C) *name* é o nome do contexto de navegação aninhada.
(D) *srcdoc* é o endereço do recurso.
(E) *height* é o alinhamento horizontal.

**16.** Qual elemento HTML5 fornece um ponto de integração para uma aplicação externa ou conteúdo interativo (ambos tipicamente não HTML)?

(A) *source*
(B) *track*
(C) *captcha*
(D) *embed*
(E) *param*

**17.** Considere as seguintes afirmações sobre Scrum.

I - Scrum não prescreve o uso de práticas de programação, como programação em pares e desenvolvimento *test-first*. Portanto, pode ser usado com abordagens ágeis mais técnicas, como XP, para fornecer um *framework* de gerenciamento de projeto.

II - Um *sprint* do Scrum é uma unidade de planejamento na qual o trabalho a ser feito é avaliado, os recursos para o desenvolvimento são selecionados e o *software* é implementado. No fim de um *sprint*, a funcionalidade completa é entregue aos *stakeholders*.

III- Toda a equipe participa das reuniões diárias; às vezes, essas são feitas com os participantes em pé (*stand up*), de forma muito rápida, para a manutenção do foco da equipe. Durante a reunião, todos os membros da equipe compartilham informações e descrevem seu progresso desde a última reunião, debatendo os problemas que surgiram desde então e o que está planejado para o dia seguinte.

Quais estão corretas?

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

**18.** No Scrum, o ponto de partida para o planejamento é _____, que é a lista do trabalho a ser feito no projeto. Durante a fase de avaliação do *sprint*, essa lista é revista, e as prioridades e o riscos são identificados. O cliente está intimamente envolvido nesse processo e, no início de cada *sprint*, pode introduzir novos requisitos ou tarefas.

Assinale a alternativa que completa, corretamente, a lacuna do trecho acima.

(A) *Backlog* do produto (*Product Backlog*)
(B) Estrutura de Subdivisão do Trabalho (*Work Breakdown Structure* – WBS)
(C) Registro parcial de trabalho (*Sprint Backlog*)
(D) Técnica de avaliação e revisão de Programa (*Program Evaluation and Review Technique* – PERT)
(E) Rede do valor Agregado (*Earned Value Network* – EVN)

**19.** Sobre teste de unidade, considere as afirmações abaixo.

I - Sempre que possível, deve-se automatizar os testes de unidade. No entanto, se deseja-se realizar testes manuais, pode-se usar um *framework* de teste (como *JUnit*) para escrever e executar testes do programa.

II - Um teste automatizado tem três partes. Há uma parte de configuração, em que se inicia o sistema com o caso de teste, ou seja, as entradas e saídas esperadas; há uma parte de chamada, em que se chama o objeto ou método a ser testado; há uma parte de afirmação, em que se compara o resultado da chamada com o resultado esperado. Se a afirmação avaliada for verdadeira, o teste foi bem sucedido; se for falsa, houve falha no teste.

III- É muito importante a escolha de casos de teste efetivos. Deve-se, portanto, escrever dois tipos de casos de teste. O primeiro deve refletir o funcionamento normal de um programa e deve mostrar que o componente funciona. Por exemplo, se está sendo testado um componente que cria e inicia um novo registro de paciente, o caso de teste deve mostrar que o registro existe no banco de dados e que os campos foram criados como especificados. Outro tipo de caso de teste deve ser baseado em testes de experiência, nos quais surgem os problemas mais comuns. Devem-se usar entradas anormais para verificar que essas sejam devidamente processados e que não façam o componente falhar.

Quais estão corretas?

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

**20.** Como se chama o processo que testa individualmente os componentes de programa, como métodos ou classes de objetos?

(A) Teste de sistema.
(B) Desenvolvimento dirigido a testes (TDD).
(C) Teste de *release*.
(D) Teste unitário.
(E) Teste de aceitação.

**21.** Sobre JSF 2.0, considere as afirmações abaixo.

I - JSF é projetado para aliviar significativamente o ônus da escrita e manutenção de aplicativos que são executados em um servidor de aplicativos Java e também para tornar as suas interfaces com usuário mais focadas no cliente.

II - JSF está incluído na plataforma Java EE; portanto, podem-se criar aplicativos que usem JSF, sem acrescentar quaisquer bibliotecas extras ao seu projeto. JSF funciona igualmente bem como um *framework web* autônomo (*standalone*), capaz de ser usado com *Spring*.

III- JSF 2.0 fornece uma API comum *JavaScript* que pode ser usada pelos componentes de interface do usuário para ajudar a promover a interoperabilidade.

Quais estão corretas?

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

**22.** Assinale a alternativa que apresenta o *framework* padrão orientado a componente de interface de usuário para a plataforma Java EE – ou seja, um *framework web* baseado em Java.

(A) *JBoss AS*.
(B) *Oracle's WebLogic*.
(C) *JavaServer Faces* (JSF).
(D) *GlassFish Open Source Edition*.
(E) *Apache Maven*.

**23.** JSF 2 possui um mecanismo denominado _____ que realiza a checagem da consistência dos dados de entrada fornecidos para cada componente *EditableValueHolder* na árvore de componentes. JSF 2 define uma suite padronizada de implementações que realizam um variedade de checagens comumente requeridas.

Assinale a alternativa que completa, corretamente, a lacuna do texto acima.

(A) resposta *(response)*
(B) requisição *(request)*
(C) evento *(event)*
(D) validador *(validator)*
(E) visão *(view)*

**24.** Sobre Maven, considere as afirmações abaixo.

I - *Maven* fornece uma abundância de informações úteis sobre o projeto, que são, em parte, retiradas do seu *Project Object Model* (POM) e, em parte, geradas a partir dos fontes do projeto. O *Maven* pode fornecer, por exemplo, *log* de alterações de documentos.

II - *Maven* visa tornar o processo de *build* mais fácil e prover um sistema de *build* uniforme, adotando o *Project Object Model* (POM) e um conjunto de *plugins*.

III- Apesar de ser muito útil para suporte a *builds, Maven* não provê apoio à especificação e à execução de testes de unidade como parte normal de um ciclo de *build*.

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.

**25.** Sobre Versionamento e Gerenciamento de Versões, considere as afirmações abaixo.

I - O gerenciamento de versões é o processo de acompanhamento de diferentes versões de componentes de *software* ou itens de configuração e dos sistemas em que esses componentes são usados. Refere-se também à garantia de que as mudanças feitas por diferentes desenvolvedores para essas versões não interfiram umas nas outras.

II - Em um sistema de gerenciamento de versões com uma variedade de recursos, desenvolvedores diferentes não podem trabalhar, ao mesmo tempo, no mesmo componente, pois se o fizessem, as mudanças feitas por diferentes desenvolvedores poderiam interferir umas nas outras.

III- Os sistemas de gerenciamento de versões fornecem, em geral, recursos de gerenciamento de armazenamento. Sua função é reduzir o espaço de armazenamento requerido pelas várias versões de componentes, que diferem apenas ligeiramente umas das outras. Em vez de manter uma cópia completa de cada versão, o sistema armazena uma lista de diferenças (deltas) entre uma versão e outra.

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.

**26.** Considere os itens abaixo.

I - Depuração simplificada: quando um teste falha, a localização do problema deve ser óbvia.
II - Diminuição da necessidade de cobertura de código: nem todo segmento de código que é escrito deve ter um teste associado.
III- Documentação do sistema: os testes agem como uma forma de documentação, descrevendo o que o código deve estar fazendo.

Quais são benefícios do uso do Desenvolvimento Dirigido por Testes (TDD)?

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

**27.** A uma coleção de versões de componentes que compõem um sistema, dá-se o nome de _____. Há um controle sobre essa coleção, o que significa que as versões dos componentes que constituem o sistema não podem ser alteradas e, portanto, deveria ser sempre possível recriar a coleção a partir de seus componentes.

Assinale a alternativa que preenche, corretamente, a lacuna do texto acima.

(A) *codeline*
(B) ramificação *(branching)*
(C) *mainline*
(D) *release*
(E) *baseline*

**28.** Sobre *PrimeFaces*, considere as afirmações abaixo.

I - *HtmlEditor*, *Dialog*, *AutoComplete* e *Charts* são exemplos de componentes disponibilizados por *PrimeFaces*.
II - *Dialog Framework* (DF) é usado para abrir uma página xhtml externa em um diálogo que é gerado dinamicamente a tempo de execução.
III- *PrimeFaces Mobile* (PFM) é um *kit* de interface com usuário (UI) para criar aplicações *JavaServer Faces* (JSF) otimizadas para dispositivos móveis.

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.

**29.** Sobre Desenvolvimento Dirigido por Testes (TDD), considere os itens abaixo.

I - Teste Isolado *(Isolated Test)*
II - Teste Primeiro *(Test First)*
III- Teste de Explicação *(Explanation Test)*

Quais são padrões para TDD?

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

**30.** Métricas de produto são usadas para medir atributos de qualidade de um sistema de *software*. Por definição, _____ é a medida do número de métodos que poderiam ser executados em resposta a uma mensagem recebida por um objeto dessa classe. Quanto maior o seu valor, mais complexa é a classe e, portanto, mais provável que inclua erros.

Assinale a alternativa que preenche, corretamente, a lacuna do texto acima.

(A) complexidade ciclomática
(B) resposta para uma classe
(C) índice *fog*
(D) método ponderado por classe
(E) profundidade de aninhamento condicional

**31.** *JavaScript* oferece algumas funções globais. A função global que recebe como argumento um *string* representando o código *JavaScript* e o executa denomina-se

(A) *ParseInt*.
(B) *escape.*
(C) *unescape.*
(D) *eval.*
(E) *ParseFloat*.

**32.** Considere as afirmações abaixo em relação à herança, um dos principais conceitos de programação orientada a objetos.

I - A herança permite a criação de hierarquias de classes, e a raiz dessa árvore deve ser uma classe abstrata.

II - As classes derivadas de uma determinada classe são chamadas de subclasses.

III- Superclasses são classes a partir das quais foram definidas subclasses.

Quais estão corretas?

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

**33.** No que se refere a classes abstratas e polimorfismo, assinale com **V** (verdadeiro) ou **F** (falso) as afirmações abaixo.

( ) Classes abstratas não podem ser instanciadas.
( ) Todos os métodos de uma classe abstrata devem estar sobrescritos na classe concreta que a estende.
( ) Uma classe concreta corresponde a uma generalização de uma classe abstrata.
( ) Diferentes classes concretas de uma mesma classe abstrata podem sobrescrever o mesmo método.

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

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

**34.** Numere a segunda coluna de acordo com a primeira, associando os tipos de direito de acesso a métodos de uma classe às respectivas características.

(1) *Private*
(2) *Protected*
(3) *Public*

( ) Métodos que podem ser acessados em classes externas ao respectivo *package*.
( ) Métodos que só podem ser chamados dentro da classe em que estão definidos.
( ) Métodos que podem ser chamados dentro das classes derivadas daquela em que estão definidos.

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

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

**35.** A respeito de variáveis na linguagem Java, assinale a alternativa que apresenta a afirmação correta.

(A) Variáveis declaradas como *static* não podem ser também *final*.
(B) Os nomes "Salario" e "salario", quando usados em declarações diferentes dentro da mesma classe, representam a mesma variável.
(C) Variáveis locais são declaradas da mesma forma que variáveis de instância, mas somente são visíveis dentro do método no qual estão declaradas.
(D) Variáveis de instância têm cópia única para todos os objetos da classe quando recebem a qualificação *final*.
(E) Parâmetros são variáveis, sendo que seus valores podem ser alterados dentro dos métodos nos quais estão definidos.

**Instrução:** Para responder as questões de nº **36** e nº **37**, considere o código abaixo.

```
1  List<Integer> minhaLista = new LinkedList<Integer>( );
2  minhaLista.add(new Integer(0));
3  Integer posic = minhaLista.iterator( ).next( );
```

**36.** A respeito do código acima, considere as seguintes afirmações.

I - A declaração na linha fonte 1 faz uso do conceito de *Generics* em Java.

II - A linha fonte 2 contém uma chamada de método para adicionar um objeto do tipo primitivo inteiro na estrutura minhaLista.

III- Uma operação de *cast* na linha 3 é necessária porque minhaLista pode conter objetos de qualquer tipo.

Quais estão corretas?

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

**37.** A respeito da linha 3, é correto afirmar que

(A) iterator( ) é um método da classe Integer.
(B) next( ) é um método que devolve um elemento de minhaLista.
(C) posic é um novo objeto que recebe o valor inteiro retirado da lista.
(D) um *iterator* contém os objetos devolvidos numa certa ordem.
(E) um *iterator* é atualizado implicitamente por um método da interface List a cada chamada de next( ).

**Instrução:** Para responder as questões de nº **38** a nº **40**, considere o trecho de código abaixo.

```
1      public class Veiculo {

2      private int HP;
3      private int consumoMedio;
4      private int velocidadeMaxima;
5      private int ID;

6      private static int contador = 0;

7      public Veiculo( ) {ID=++contador;}
8      public int retornaID( ) {return ID;}
9      public void imprimeCampos( ) { ... }

       ...
10   }

11   public class Utilitario extends Veiculo {
12     public String tipo;
13     public Utilitario (String tipoU) {
14          super( );
15          tipo = tipoU;
16     public void imprimeCampos ( ) { ... }
       ...
17      }
18   }
```

**38.** Utilizando esse código, são executadas as chamadas abaixo:

```
Veiculo v1 = new Veiculo( );
Veiculo v2 = new Veiculo( );
Utilitario U1 = new Utilitario ("4x4");
Veiculo v3 = new Veiculo( );
```

Qual o valor da variável contador após a última chamada?

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

**39.** Os atributos de instância da classe Utilitario são:

(A) HP, consumoMedio, velocidadeMaxima, ID
(B) HP, consumoMedio, velocidadeMaxima, ID, tipo
(C) HP, consumoMedio, velocidadeMaxima, ID, contador
(D) HP, consumoMedio, velocidadeMaxima, ID, tipo, contador
(E) tipo, tipoU, contador

**40.** Em relação a uma chamada Utilitario U2 = new Utilitario ("4x4"); considere as afirmações abaixo.

I - A classe Utilitario é uma especialização da classe Veiculo.

II - Uma chamada U2.imprimeCampos(); causa a execução do código na linha 16, seguida da execução do código da linha 9.

III- O código da linha 16 corresponde à sobrescrita do método definido na linha 9.

Quais estão corretas?

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