



Mais que um banco. O Nosso Banco.

CONCURSO PÚBLICO

2. PROVA OBJETIVA

Língua Portuguesa e Conhecimentos Específicos

ANALISTA DE INFORMÁTICA JÚNIOR

INSTRUÇÕES

- ♦ VOCÊ RECEBEU SUA FOLHA DEFINITIVA DE RESPOSTAS E ESTE CADERNO CONTENDO 80 QUESTÕES OBJETIVAS.
- ♦ PREENCHA, COM SEU NOME E NÚMERO DE INSCRIÇÃO, OS ESPAÇOS RESERVADOS NA CAPA DESTE CADERNO.
- ♦ LEIA CUIDADOSAMENTE AS QUESTÕES OBJETIVAS E ESCOLHA A RESPOSTA QUE VOCÊ CONSIDERA CORRETA.
- ♦ ASSINALE, NA TIRA DE RESPOSTAS DA CAPA DESTE CADERNO, A ALTERNATIVA DE SUA OPÇÃO.
- ♦ RESPONDA A TODAS AS QUESTÕES.
- ♦ TRANSCREVA PARA A FOLHA DEFINITIVA DE RESPOSTAS, COM CANETA DE TINTA AZUL OU PRETA, A ALTERNATIVA QUE JULGAR CERTA.
- ♦ A DURAÇÃO DA PROVA É DE 4 HORAS E 30 MINUTOS.
- ♦ A SAÍDA DO CANDIDATO DO PRÉDIO SERÁ PERMITIDA APÓS TRANSCORRIDAS 2 HORAS DO INÍCIO DA PROVA E SÓ SERÁ PERMITIDO LEVAR O CADERNO DE QUESTÕES DEPOIS DE TRANSCORRIDAS 3 HORAS DO INÍCIO DA PROVA.
- ♦ AO TERMINAR A PROVA, VOCÊ ENTREGARÁ AO FISCAL A FOLHA DEFINITIVA DE RESPOSTAS.

AGUARDE A ORDEM DO FISCAL PARA ABRIR ESTE CADERNO DE QUESTÕES.

Nome do candidato _____

Número de inscrição _____

RESPOSTAS

01	<input type="checkbox"/>	41	<input type="checkbox"/>
02	<input type="checkbox"/>	42	<input type="checkbox"/>
03	<input type="checkbox"/>	43	<input type="checkbox"/>
04	<input type="checkbox"/>	44	<input type="checkbox"/>
05	<input type="checkbox"/>	45	<input type="checkbox"/>
06	<input type="checkbox"/>	46	<input type="checkbox"/>
07	<input type="checkbox"/>	47	<input type="checkbox"/>
08	<input type="checkbox"/>	48	<input type="checkbox"/>
09	<input type="checkbox"/>	49	<input type="checkbox"/>
10	<input type="checkbox"/>	50	<input type="checkbox"/>
11	<input type="checkbox"/>	51	<input type="checkbox"/>
12	<input type="checkbox"/>	52	<input type="checkbox"/>
13	<input type="checkbox"/>	53	<input type="checkbox"/>
14	<input type="checkbox"/>	54	<input type="checkbox"/>
15	<input type="checkbox"/>	55	<input type="checkbox"/>
16	<input type="checkbox"/>	56	<input type="checkbox"/>
17	<input type="checkbox"/>	57	<input type="checkbox"/>
18	<input type="checkbox"/>	58	<input type="checkbox"/>
19	<input type="checkbox"/>	59	<input type="checkbox"/>
20	<input type="checkbox"/>	60	<input type="checkbox"/>
21	<input type="checkbox"/>	61	<input type="checkbox"/>
22	<input type="checkbox"/>	62	<input type="checkbox"/>
23	<input type="checkbox"/>	63	<input type="checkbox"/>
24	<input type="checkbox"/>	64	<input type="checkbox"/>
25	<input type="checkbox"/>	65	<input type="checkbox"/>
26	<input type="checkbox"/>	66	<input type="checkbox"/>
27	<input type="checkbox"/>	67	<input type="checkbox"/>
28	<input type="checkbox"/>	68	<input type="checkbox"/>
29	<input type="checkbox"/>	69	<input type="checkbox"/>
30	<input type="checkbox"/>	70	<input type="checkbox"/>
31	<input type="checkbox"/>	71	<input type="checkbox"/>
32	<input type="checkbox"/>	72	<input type="checkbox"/>
33	<input type="checkbox"/>	73	<input type="checkbox"/>
34	<input type="checkbox"/>	74	<input type="checkbox"/>
35	<input type="checkbox"/>	75	<input type="checkbox"/>
36	<input type="checkbox"/>	76	<input type="checkbox"/>
37	<input type="checkbox"/>	77	<input type="checkbox"/>
38	<input type="checkbox"/>	78	<input type="checkbox"/>
39	<input type="checkbox"/>	79	<input type="checkbox"/>
40	<input type="checkbox"/>	80	<input type="checkbox"/>

LÍNGUA PORTUGUESA

Leia o texto a seguir para responder às questões de números **01** a **10**.

De um lado estão os prejuízos e a restrição de direitos causados pelos protestos que param as ruas de São Paulo. De outro está o direito à livre manifestação, assegurado pela Carta de 1988. Como não há fórmula perfeita de arbitrar esse choque entre garantias democráticas fundamentais, cabe lançar mão de medidas pontuais – e sobretudo de bom senso.

A Companhia de Engenharia de Tráfego (CET) estima em R\$ 3 milhões o custo para a população dos protestos ocorridos nos últimos três anos na capital paulista. O cálculo leva em conta o combustível consumido e as horas perdidas de trabalho durante os engarrafamentos causados por protestos. Os carros enfileirados por conta de manifestações nesses três anos praticamente cobririam os 231 km que separam São Paulo de São Carlos.

A Justiça é o meio mais promissor, em longo prazo, para desestimular os protestos abusivos que param o trânsito nos horários mais inconvenientes e acarretam variados transtornos a milhões de pessoas. É adequada a atitude da CET de enviar sistematicamente ao Ministério Público relatórios com os prejuízos causados em cada manifestação feita fora de horários e locais sugeridos pela agência ou sem comunicação prévia.

Com base num documento da CET, por exemplo, a Procuradoria acionou um líder de sindicato, o qual foi condenado em primeira instância a pagar R\$ 3,3 milhões aos cofres públicos, a título de reparação. O direito à livre manifestação está previsto na Constituição. No entanto, tal direito não anula a responsabilização civil e criminal em caso de danos provocados pelos protestos.

O poder público deveria definir, de preferência em negociação com as categorias que costumam realizar protestos na capital, horários e locais vedados às passeatas. Práticas corriqueiras, como a paralisação de avenidas essenciais para o tráfego na capital nos horários de maior fluxo, deveriam ser abolidas.

(Folha de S.Paulo, 29.09.07. Adaptado)

01. De acordo com o texto, é correto afirmar que

- (A) a Companhia de Engenharia de Tráfego não sabe mensurar o custo dos protestos ocorridos nos últimos anos.
- (B) os prejuízos da ordem de R\$ 3 milhões em razão dos engarrafamentos já foram pagos pelos manifestantes.
- (C) os protestos de rua fazem parte de uma sociedade democrática e são permitidos pela Carta de 1988.
- (D) após a multa, os líderes de sindicato resolveram organizar protestos de rua em horários e locais predeterminados.
- (E) o Ministério Público envia com frequência estudos sobre os custos das manifestações feitas de forma abusiva.

02. No primeiro parágrafo, afirma-se que não há fórmula perfeita para solucionar o conflito entre manifestantes e os prejuízos causados ao restante da população. A saída estaria principalmente na

- (A) sensatez.
- (B) Carta de 1998.
- (C) Justiça.
- (D) Companhia de Engenharia de Tráfego.
- (E) na adoção de medidas amplas e profundas.

03. De acordo com o segundo parágrafo do texto, os protestos que param as ruas de São Paulo representam um custo para a população da cidade. O cálculo desses custos é feito a partir

- (A) das multas aplicadas pela Companhia de Engenharia de Tráfego (CET).
- (B) dos gastos de combustível e das horas de trabalho desperdiçadas em engarrafamentos.
- (C) da distância a ser percorrida entre as cidades de São Paulo e São Carlos.
- (D) da quantidade de carros existentes entre a capital de São Paulo e São Carlos.
- (E) do número de usuários de automóveis particulares da cidade de São Paulo.

04. A quantidade de carros parados nos engarrafamentos, em razão das manifestações na cidade de São Paulo nos últimos três anos, é equiparada, no texto,

- (A) a R\$ 3,3 milhões.
- (B) ao total de usuários da cidade de São Carlos.
- (C) ao total de usuários da cidade de São Paulo.
- (D) ao total de combustível economizado.
- (E) a uma distância de 231 km.

- 05.** No terceiro parágrafo, a respeito do poder da Justiça em coibir os protestos abusivos, o texto assume um posicionamento de
- (A) indiferença, porque diz que a decisão não cabe à Justiça.
 - (B) entusiasmo, porque acredita que o órgão já tem poder para impedir protestos abusivos.
 - (C) decepção, porque não vê nenhum exemplo concreto do órgão para impedir protestos em horários de pico.
 - (D) confiança, porque acredita que, no futuro, será uma forma bem-sucedida de desestimular protestos abusivos.
 - (E) satisfação, porque cita casos em que a Justiça já teve êxito em impedir protestos em horários inconvenientes e em avenidas movimentadas.
- 06.** De acordo com o texto, a atitude da Companhia de Engenharia de Tráfego de enviar periodicamente relatórios sobre os prejuízos causados em cada manifestação é
- (A) pertinente.
 - (B) indiferente.
 - (C) irrelevante.
 - (D) onerosa.
 - (E) inofensiva.
- 07.** No quarto parágrafo, o fato de a Procuradoria condenar um líder sindical
- (A) é ilegal e fere os preceitos da Carta de 1998.
 - (B) deve ser comemorada, ainda que viole a Constituição.
 - (C) é legal, porque o direito à livre manifestação não isenta o manifestante da responsabilidade pelos danos causados.
 - (D) é nula, porque, segundo o direito à livre manifestação, o acusado poderá entrar com recurso.
 - (E) é inédita, porque, pela primeira vez, apesar dos direitos assegurados, um manifestante será punido.
- 08.** Dentre as soluções apontadas, no último parágrafo, para resolver o conflito, destaca-se
- (A) multa a líderes sindicais.
 - (B) fiscalização mais rígida por parte da Companhia de Engenharia de Tráfego.
 - (C) o fim dos protestos em qualquer via pública.
 - (D) fixar horários e locais proibidos para os protestos de rua.
 - (E) negociar com diferentes categorias para que não façam mais manifestações.
- 09.** No trecho – *É adequada a atitude da CET de enviar relatórios* –, substituindo-se o termo *atitude* por *comportamentos*, obtém-se, de acordo com as regras gramaticais, a seguinte frase:
- (A) É adequada comportamentos da CET de enviar relatórios.
 - (B) É adequado comportamentos da CET de enviar relatórios.
 - (C) São adequado os comportamentos da CET de enviar relatórios.
 - (D) São adequadas os comportamentos da CET de enviar relatórios.
 - (E) São adequados os comportamentos da CET de enviar relatórios.
- 10.** No trecho – *No entanto, tal direito não anula a responsabilização civil e criminal em caso de danos provocados pelos protestos* –, a locução conjuntiva *no entanto* indica uma relação de
- (A) causa e efeito.
 - (B) oposição.
 - (C) comparação.
 - (D) condição.
 - (E) explicação.

CONHECIMENTOS ESPECÍFICOS

11. Uma das etapas para se desenvolver um programa de computador envolve o desenvolvimento de um Modelo _____, destinado para se obter uma _____ do sistema. Já no Modelo _____ são especificados os _____ e os diagramas lógicos para a solução.

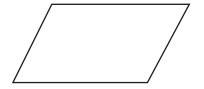
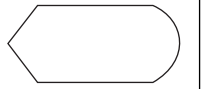

Assinale a alternativa que completa, correta e respectivamente, as lacunas do texto.

- (A) Conceitual ... especificação funcional ... Físico ... algoritmos
 (B) Conceitual ... especificação funcional ... Lógico ... algoritmos
 (C) Conceitual ... validação ... Físico ... dados
 (D) Lógico ... especificação funcional ... Conceitual ... algoritmos
 (E) Lógico ... validação ... Conceitual ... dados

12. A etapa final do processo de criação de sistemas é a representação formal de um modelo em uma linguagem de programação. Sobre essa etapa, pode-se afirmar que ela

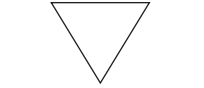
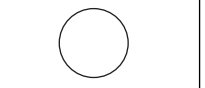

- (A) corresponde ao passo seguinte da etapa de geração do Modelo Lógico.
 (B) é denominada Realização.
 (C) é denominada Implementação.
 (D) é precedida pela etapa de Modelagem Abstrata e sucedida pela etapa de Modelagem Real.
 (E) é sucedida pela etapa de Modelagem.

13. O Fluxograma, também conhecido com Diagrama de Blocos, utiliza diversos blocos, sendo alguns representados a seguir.

1	2	3
		

As denominações desses blocos são:

- (A) 1-Decisão; 2-Display; 3-Impressora.
 (B) 1-Impressora; 2-Terminal; 3-Decisão.
 (C) 1-Leitura/Gravação; 2-Disco; 3-Teclado.
 (D) 1-Leitura/Gravação; 2-Display; 3-Teclado.
 (E) 1-Leitura/Gravação; 2-Terminal; 3-Teclado.
14. Os Fluxogramas podem requerer representações extensas. Analise as afirmações sobre os símbolos que podem ser utilizados nessas situações, considerando a tabela a seguir.

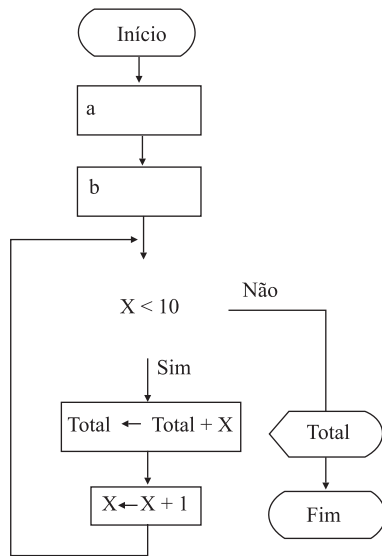
1	2	3
		

- I. O símbolo 1 é denominado Conector, e se destina a fazer conexão de páginas, caso o fluxograma ocupe mais de uma página.
 II. O símbolo 2 é utilizado quando é preciso seccionar o fluxograma, ligando uma parte a outra. Dentro dele deve ser colocado um número seqüencial.
 III. O símbolo 3 se destina a fazer conexão de subrotinas, caso estas necessitem ser seccionadas.

Sobre as afirmações, pode-se dizer que está correto o contido em

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

Considere o fluxograma a seguir para responder às questões de números 15 e 16. O fluxograma deverá representar o algoritmo que soma os 10 primeiros números inteiros positivos (iniciando por 0).



15. O bloco que deverá ser colocado na condição $X < 10$ é:

- (A)
- (B)
- (C)
- (D)
- (E)

16. Os comandos que deverão ser colocados nos blocos a e b são, respectivamente,

- (A) $X \leftarrow 0$; $Total \leftarrow 0$.
- (B) $X \leftarrow 0$; $Total \leftarrow 1$.
- (C) $X \leftarrow -1$; $Total \leftarrow 0$.
- (D) $X \leftarrow -1$; $Total \leftarrow 1$.
- (E) $X \leftarrow -1$; $Total \leftarrow X$.

Considere o enunciado a seguir para responder às questões de números 17 e 18.

Na etapa de elaboração do Modelo Lógico de um programa é necessário empregar operadores. A tabela a seguir apresenta alguns nomes e símbolos de operadores e suas classificações.

Operadores:					
1	Unário	4	Resto da Divisão		
2	Exponenciação	5	Diferente de		
3	Negação	6	Divisão real		
Símbolos dos Operadores:					
a	Rest	e	Mod	i	\neq
b	neg	f	.não.	j	1
c	^	g	-	k	/
d	e	h	\diamond	l	Div
Classificação:					
I	Operador Aritmético				
II	Operador Relacional				
III	Operador Lógico				

17. Uma correspondência correta entre Operadores, seus respectivos Símbolos e Classificação é:

- (A) 1-g-I; 2-c-I; 3-f-II
- (B) 1-g-I; 3-f-III; 5-h-II
- (C) 1-j-II; 3-b-III; 5-i-II
- (D) 2-d-I; 4-e-I; 6-l-I
- (E) 4-a-I; 5-h-II; 6-k-I

18. Em relação à prioridade dos operadores presentes na tabela, é correto afirmar que

- (A) 1 é mais prioritário do que 6.
- (B) 1 é menos prioritário do que 2.
- (C) 1 é menos prioritário do que 4.
- (D) 2 é menos prioritário do que 4.
- (E) 4 é mais prioritário do que 6.

Considere o programa de nome Ordenação, escrito na forma de um pseudocódigo, para responder às questões de números 19 e 20.

```

programa Ordenação
  var Aux, j, k: inteiro
    Vetor: array[1..n] de inteiro
  início
    {leitura dos elementos do vetor}
    para j de 1 até n faça
      leia Vetor [j]
    fim_para
    {ordenação do vetor – Método de Bolha}
    para j de 1 até (n – 1) faça
      para k de (j+1) até n faça
        se (Vetor [j] > Vetor [k]) então
          Aux ← Vetor [j]
          Vetor [j] ← Vetor [k]
          Vetor [k] ← Aux
        fim_se
      fim_para.k
    fim_para.j
  fim

```

19. Suponha que o vetor lido seja: 3, 2, 4, 1. A tabela a seguir deve apresentar a evolução desse vetor à medida que o algoritmo vai sendo executado.

Linha 1	3 2 4 1
Linha 2	
Linha 3	
Linha 4	
Linha 5	
Linha 6	
Linha 7	1 2 3 4

Leia as afirmações:

- I. a linha 2 deverá possuir: 2 3 4 1;
- II. a linha 3 deverá possuir: 2 3 4 1;
- III. a linha 4 deverá possuir: 1 3 4 2;
- IV. a linha 5 deverá possuir: 1 3 4 2.

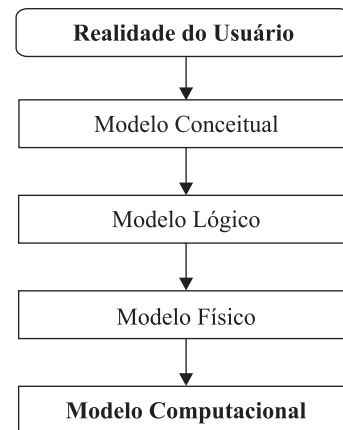
Sobre as afirmações, pode-se dizer que está correto o conteúdo em

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

20. No programa anterior, caso o vetor lido seja 3, 2, 5, 4, 1, o número de linhas que a tabela deverá apresentar é:

- (A) 7.
- (B) 8.
- (C) 9.
- (D) 10.
- (E) 11.

21. A criação de um programa (aplicação) envolve diversas etapas, visando traduzir as necessidades do usuário, dentro de sua realidade, para um sistema de computação. Considere as seguintes etapas para a solução de um problema.



A frase “construído(a) pelo desenvolvedor para obter uma especificação funcional do sistema” refere-se

- (A) à Realidade do Usuário.
- (B) ao Modelo Conceitual.
- (C) ao Modelo Lógico.
- (D) ao Modelo Físico.
- (E) ao Modelo Computacional.

22. Considere alguns dos objetivos da programação estruturada, quando do seu surgimento:

- I. obtenção de programas mais corretos;
- II. domínio da extensão dos programas (melhor manutenção);
- III. disciplina do raciocínio.

Dentre os objetivos listados, pode-se dizer que foi alcançado o contido em

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

23. A sistemática de programação estruturada na qual o método de desenvolvimento de programas parte de uma solução mais geral para soluções particulares é denominada

- (A) *factored*.
- (B) *specialized*.
- (C) *bottom-up*.
- (D) *top-down*.
- (E) *up-down*.

RASCUNHO

24. Considere o programa de nome Programa_1, escrito na forma de um pseudocódigo e apresentado a seguir.

```
programa Programa_1
  var X, N1, N2, N3: Real
início
  leia N1, N2, N3
  se (N1 > N2) então
    X ← N1
  senão
    X ← N2
  fim_se
  se (N3 > X) então
    X ← N3
  fim_se
  imprima X
fim
```

Assinale a alternativa que expressa o algoritmo implementado por esse programa.

- (A) Calcula a média aritmética de três números lidos e a imprime.
- (B) Calcula a média ponderada de três números lidos e a imprime.
- (C) Encontra o maior número dentre três números lidos e o imprime.
- (D) Encontra o menor número dentre três números lidos e o imprime.
- (E) Obtém o número intermediário dentre os três números lidos e o imprime.

RASCUNHO

25. Deseja-se elaborar um algoritmo para o cálculo da média ponderada de um aluno de um curso superior. Para isso, dispõe-se de dois vetores contendo, respectivamente, as notas do aluno em 5 disciplinas do semestre (D) e o número de créditos de cada uma destas disciplinas (NC), conforme mostrado a seguir.

Vetor de Notas nas Disciplinas (D)

D1	D2	D3	D4	D5
----	----	----	----	----

Vetor de Número de Créditos (NC) nas Disciplinas

NC1	NC2	NC3	NC4	NC5
-----	-----	-----	-----	-----

Desenvolveu-se um algoritmo para realizar essa implementação, apresentado a seguir, onde o símbolo ♣ representa duas linhas que foram suprimidas.

programa Média

var MP, SD, SNC: Real

var i: Inteiro

var D **array**[1..5] **de** Real, NC **array**[1..5] **de** Real

início

...

{Lê as notas do aluno e número de créditos das disciplinas e armazena nos vetores SD e SNC, respectivamente}

...

SD \leftarrow 0,0

SNC \leftarrow 0,0

para i **de** 1 **até** 5 **faça**

♣

fim_para

MP \leftarrow SD / SNC

imprima MP

fim

Assinale a alternativa que contém as linhas que completam, corretamente, o algoritmo apresentado.

(A) SD \leftarrow D[i]
SNC \leftarrow NC[i] + 1

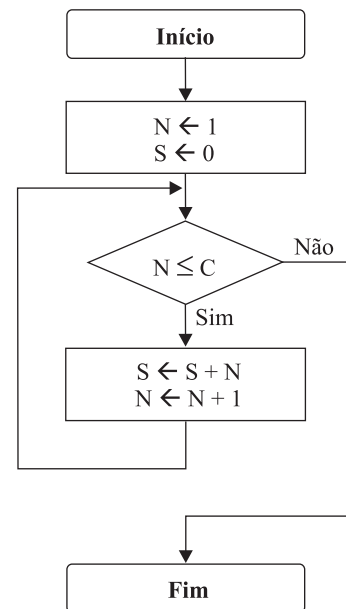
(B) SD \leftarrow NC[i]*D[i]
SNC \leftarrow NC[i] + i

(C) SD \leftarrow SD + D[i]
SNC \leftarrow SNC + 1

(D) SD \leftarrow SD + NC[i]*D[i]
SNC \leftarrow SNC + i

(E) SD \leftarrow SD + NC[i]*D[i]
SNC \leftarrow SNC + NC[i]

26. Considere um algoritmo expresso pelo seguinte fluxograma.



Suponha que C apresente o valor 7. Ao final da execução desse algoritmo, a variável S terá o valor

- (A) 7.
- (B) 8.
- (C) 21.
- (D) 28.
- (E) 36.

Considere o programa de nome Programa_F, escrito na forma de um pseudocódigo, para responder às questões de números 27 e 28.

programa Programa_F

var F, i: Inteiro

início

F \leftarrow 1

i \leftarrow 1

repita

F \leftarrow F*i

i \leftarrow i + 1

até que (i=N)

♥

...

Fim

27. Supondo que a constante N seja igual a 5, quando o programa anterior atingir o ponto ♥, pode-se afirmar que F será igual a

- (A) 1.
- (B) 5.
- (C) 6.
- (D) 24.
- (E) 120.

28. Com base no programa anterior, pode-se dizer que o
- (A) pseudocódigo só funciona para N maior do que 2.
 - (B) pseudocódigo causará um erro de execução para todo valor de N maior do que 5.
 - (C) pseudocódigo apresenta uma estrutura de repetição com teste no final.
 - (D) código presente no bloco **repita ... até que** não será executado se N for igual a 1.
 - (E) código presente no bloco **repita ... até que** não será executado se N for igual a 0.

Considere o programa de nome Programa_P, escrito na forma de um pseudocódigo, para responder às questões de números 29 e 30. O operador **Mod** faz a operação de cálculo do resto da divisão do primeiro operando pelo segundo.

```

programa Programa_P
  var i, j, Soma: Inteiro
início
  Soma ← 0
  j ← 0
  para i de 0 até 10 passo 2 faça
    j = j + 1
    se ( (j Mod 2) = 0 ) então
      Soma ← Soma + (i*j)
    senão
      Soma ← Soma - (i*j)
    fim_se
  fim_para
  imprima Soma
fim

```

29. Analise o Programa_P apresentado e assinale a alternativa que contém o número de vezes que o código dentro da estrutura **para ... fim_para** será repetido.
- (A) 4
 - (B) 6
 - (C) 8
 - (D) 10
 - (E) 12
30. O Programa_P imprimirá como resultado final, obtido a partir da variável Soma, o valor
- (A) -8.
 - (B) -24.
 - (C) 36.
 - (D) 40.
 - (E) 60.

Considere o programa de nome Matriz, escrito na forma de um pseudocódigo, para responder às questões de números 31 e 32.

```

programa Matriz
  var A array[1..3, 1..3] de Inteiro
  var i, j: Inteiro
início
  para i de 1 até 3 faça
    para j de 1 até 3 faça
      se ( i=j )
        A[i,j] ← 1
      senão
        se ( i<j )
          A[i,j] ← i
        senão
          A[i,j] ← j
        fim_se
      fim_se
    fim_para
  fim

```

31. Considere que a primeira dimensão de uma matriz represente suas linhas e que a segunda represente suas colunas. Assinale a alternativa que apresenta uma representação correta da matriz A preenchida pelo programa.

(A)
$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

(B)
$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 2 \\ 1 & 2 & 1 \end{bmatrix}$$

(C)
$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 2 \\ 1 & 3 & 1 \end{bmatrix}$$

(D)
$$\begin{bmatrix} 1 & 2 & 1 \\ 2 & 1 & 3 \\ 1 & 3 & 1 \end{bmatrix}$$

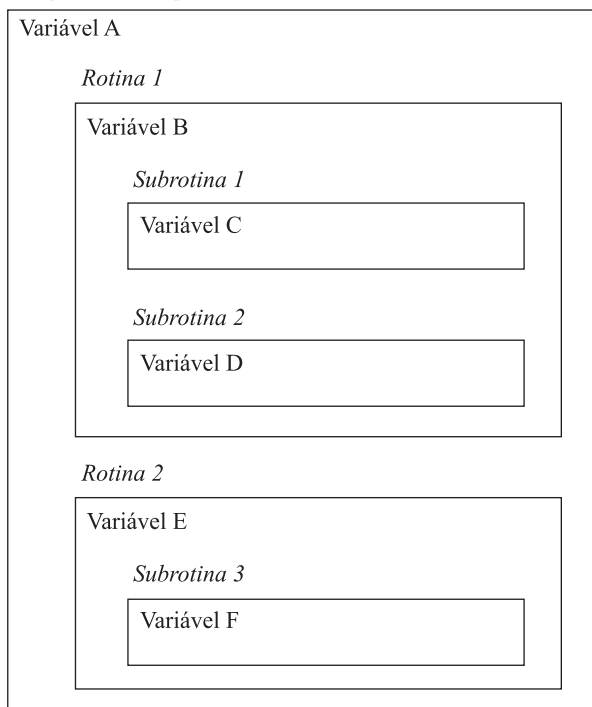
(E)
$$\begin{bmatrix} 1 & 2 & 3 \\ 2 & 2 & 2 \\ 3 & 2 & 3 \end{bmatrix}$$

32. No programa anterior, a linha $A[i,j] \leftarrow i$ é executada

- (A) 1 vez.
- (B) 2 vezes.
- (C) 3 vezes.
- (D) 6 vezes.
- (E) 9 vezes.

33. Considere a estrutura de um programa composto pelo *Programa Principal*, que chama a *Rotina 1* e a *Rotina 2* que, por sua vez, chamam as *Subrotinas 1, 2 e 3*, de acordo com a figura a seguir.

Programa Principal



Considerando a questão da visibilidade de variáveis, pode-se afirmar que

- (A) as variáveis C e D são visíveis para a *Rotina 1*.
- (B) a variável A é global a todo o programa, sendo visível para todas as rotinas e subrotinas.
- (C) a variável E, declarada na *Rotina 2*, não é visível para a *Subrotina 3*.
- (D) as variáveis C, D e F são visíveis para todas as sub-rotinas de mesmo nível (*Subrotina 1*, *Subrotina 2* e *Subrotina 3*).
- (E) o Programa Principal pode ler e alterar todas as variáveis apresentadas (A, B, C, D, E e F).

34. Considere as 4 divisões utilizadas na linguagem de programação COBOL:

- I. *IDENTIFICATION DIVISION*;
- II. *ENVIRONMENT DIVISION*;
- III. *DATA DIVISION*;
- IV. *PROCEDURE DIVISION*.

Embora algumas delas possam ser omitidas, a seqüência em que são especificadas é fixa e deve seguir a ordem:

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

35. Considere as seguintes seções utilizadas na linguagem de programação COBOL:

- I. *CONFIGURATION SECTION*;
- II. *INPUT-OUTPUT SECTION*;
- III. *FILE SECTION*;
- IV. *WORKING-STORAGE SECTION*.

Em *DATA DIVISION* são válidas as seções que estão contidas apenas em

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

36. Considere a seguinte construção na Linguagem COBOL.

DECIMAL-POINT IS COMMA.

Em *ENVIRONMENT DIVISION* esta construção é declarada na seção

- (A) *FILE SECTION*.
- (B) *STORAGE SECTION*.
- (C) *INPUT-OUTPUT SECTION*.
- (D) *CONFIGURATION SECTION*.
- (E) *WORKING-STORAGE SECTION*.

37. Na _____ da linguagem COBOL, a cláusula _____ é empregada para a definição de arquivos informando os parâmetros por meio de: *RECORD*, *LABEL RECORD*, *VALUE OF* e *DATA RECORD*.

As palavras que completam, correta e respectivamente, as lacunas do texto são:

- (A) *FILE SECTION ... FD*
- (B) *CONFIGURATION SECTION ... FILE-CONTROL*
- (C) *CONFIGURATION SECTION ... SPECIAL-NAMES*
- (D) *INPUT-OUTPUT SECTION ... FILE-CONTROL*
- (E) *INPUT-OUTPUT SECTION ... I-O-CONTROL*

38. Considere as seguintes cláusulas da linguagem de programação COBOL:

- I. *PROGRAM-ID*;
- II. *ASSIGN TO PRINTER*;
- III. *FILE-CONTROL*;
- IV. *SPECIAL-NAMES*.

Em *INPUT-OUTPUT SECTION* são válidas as cláusulas que estão contidas apenas em

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

39. Considere a seguinte expressão na linguagem de programação COBOL.

```
COMPUTE result1 ROUNDED, result2=((2*3) + 12)/10
```

Os valores das variáveis *result1* e *result2* são, respectivamente,

- (A) 1,5 e 2.
- (B) 1,8 e 1,8.
- (C) 1,8 e 2.
- (D) 2 e 1,8.
- (E) 2 e 2.

40. Considere as seguintes afirmações sobre a *WORKING-STORAGE SECTION*, utilizada para definir registros e itens de dados:

- I. os números de níveis válidos são os valores: 01 a 49, 66, 77, 78 e 88;
- II. o número de nível especial 77 é utilizado para declarar itens de dados contíguos;
- III. o número de nível especial 66 é utilizado junto com a cláusula *RENAMES*.

Sobre as afirmações, pode-se dizer que está correto o conteúdo em:

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

41. Considere as seguintes formatações e os resultados para o valor -123,45:

- I. Picture: 999,99CR -> Resultado: -123,45CR;
- II. Picture: -\$9999,99 -> Resultado: -\$0123,45;
- III. Picture: 999,99DB -> Resultado: 123,45DB

Dentre as formatações e resultados, são válidos os conteúdos em

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

42. Sobre a linguagem COBOL, pode-se dizer que:

- (A) COBOL é um acrônimo de *COmmon Business Oriented Language*.
- (B) a *American National Standards Institute* (ANSI) vem estabelecendo padrões para a linguagem.
- (C) a linguagem COBOL nasceu graças à iniciativa da Universidade de Austin no Texas – EUA.
- (D) existem versões de compiladores da linguagem para vários ambientes em plataforma baixa, como MS DOS, OS/2, WINDOWS e UNIX.
- (E) CODASYL (*Conference On Data Systems Languages*) é composta pelos fabricantes de computadores, universidades e usuários, estabelecendo as primeiras definições da linguagem COBOL.

43. Considere as afirmações sobre os comandos da linguagem COBOL:

I. há 4 tipos de comandos em COBOL: condicionais, imperativos, delimitados por escopo e diretivas do compilador;

II. um comando imperativo é aquele que indica uma ação incondicional específica;

III. uma diretiva ao compilador instrui a execução de determinadas ações em tempo de execução do programa.

Sobre as afirmações, pode-se dizer que está correto o contido em:

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

44. Considere os seguintes comandos em linguagem COBOL.

```
77 campo1 PIC 9(02) VALUE 10.
77 campo2 PIC 9(02) VALUE 20.
77 campo3 PIC 9(02) VALUE 50.
77 campo4 PIC 9(02) VALUE 15.
PROCEDURE DIVISION.
    ADD campo1 campo2 TO campo3 GIVING campo4.
    ADD campo1 campo2 GIVING campo4.
```

Sobre a execução dos comandos, pode-se dizer que: “O valor da variável `campo4` na primeira adição é _____. O valor da variável `campo4` na segunda adição é _____”.

A alternativa que completa, correta e respectivamente, as lacunas do texto é:

- (A) 30 ... 30
- (B) 60 ... 10
- (C) 70 ... 10
- (D) 80 ... 30
- (E) 95 ... 30

45. Considere os seguintes comandos em linguagem COBOL.

```
77 campo1 PIC 9(02) VALUE 60.
PROCEDURE DIVISION.
    ADD 60 TO campo1.
    MOVE 60 TO campo1.
    ADD 60 TO campo1 ON SIZE ERROR
        DISPLAY "Estouro".
```

Sobre a execução dos comandos, pode-se dizer que: “O valor da primeira adição armazenado em `campo1` é _____. O valor da segunda adição armazenado em `campo1` é _____”.

A alternativa que completa, correta e respectivamente, as lacunas do texto é:

- (A) 12 ... 60 (“Estouro”)
- (B) 20 ... 60 (“Estouro”)
- (C) 120 ... 20 (“Estouro”)
- (D) 120 ... 60 (“Estouro”)
- (E) 120 ... 120

46. Considere os seguintes comandos em linguagem COBOL.

```
SPECIAL-NAMES. DECIMAL POINT IS COMMA.
77 r1 PIC 9(03) VALUE ZEROS.
77 r2 PIC 9(03) VALUE ZEROS.
77 r3 PIC 9(03) VALUE ZEROS.
77 r4 PIC 9(03) VALUE ZEROS.
77 fator PIC 9(01) VALUE 4.
PROCEDURE DIVISION.
    COMPUTE r1 = 2+3*fator.
    COMPUTE r2 = (2+3)*r1.
    COMPUTE r3 = fator**2.
    COMPUTE r4 = fator**0,5.
```

Sobre a execução dos comandos, pode-se afirmar que os valores de `r1`, `r2`, `r3` e `r4` são, respectivamente,

- (A) 14, 70, 16 e 0,5.
- (B) 14, 70, 16 e 2.
- (C) 14, 100, 16 e 1.
- (D) 20, 100, 16 e 1.
- (E) 20, 100, 16 e 2.

47. Considere os seguintes comandos em linguagem COBOL.

```
01 item1.
   05 ca1      PIC 9(02).
   05 ca2      PIC A(02).
   05 ca3      PIC X(02).
PROCEDURE DIVISION.
  INITIALIZE item1 REPLACING NUMERIC BY 1
                  ALPHABETIC BY "A"
                  ALPHANUMERIC BY "B".
```

Sobre a execução dos comandos, pode-se dizer que o valor de item1 é (considere que "b" indica o caractere "branco")

- (A) ca1 = 01 , ca2 = Ab, ca3 = Bb
- (B) ca1 = 11 , ca2 = AA ca3 = BB
- (C) ca1 = 01 , ca2 = AA ca3 = BB
- (D) ca1 = 10 , ca2 = bA ca3 = bB
- (E) ca1 = 10 , ca2 = AA ca3 = BB

48. Considere os seguintes comandos em linguagem COBOL.

```
WORKING-STORAGE SECTION.
77 c1      PIC 9(01) VALUE ZEROS.
77 c2      PIC 9(01) VALUE ZEROS.
77 c3      PIC 9(01) VALUE ZEROS.
77 c4      PIC 9(01) VALUE ZEROS.
77 campo1  PIC X(06) VALUE "BANANA".
PROCEDURE DIVISION.
INSPECT campo1 TALLYING c1 FOR ALL "A".
INSPECT campo1 TALLYING c2 FOR ALL "A" AFTER "A".
INSPECT campo1 TALLYING c3 FOR ALL "A" AFTER "N".
INSPECT campo1 TALLYING c4 FOR ALL "A" BEFORE "N".
```

Sobre a execução dos comandos, pode-se dizer que os valores de c1, c2, c3 e c4 são, respectivamente,

- (A) 2, 1, 1 e 0.
- (B) 2, 1, 1 e 1.
- (C) 3, 1, 2 e 1.
- (D) 3, 2, 1 e 1.
- (E) 3, 2, 2 e 1.

49. Considere os seguintes comandos em linguagem COBOL.

```
SPECIAL-NAMES. DECIMAL-POINT IS COMMA.
WORKING-STORAGE SECTION.
77 c1      PIC 99.
77 c2      PIC 99V99.
77 c3      PIC 99V99.
77 c4      PIC X(02).
PROCEDURE DIVISION.
  MOVE 123          TO c1.
  MOVE 123,456     TO c2.
  MOVE 1,2         TO c3.
  MOVE "ABCD"      TO c4.
```

Considere, ainda, os seguintes possíveis valores obtidos após a execução dos comandos:

- I. c1 = 12 e c2 = 23,45
- II. c3 = 01,20
- III. c4 = "CD"

Dentre os valores apresentados, são válidos os contidos em

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

50. Considere os seguintes comandos em linguagem COBOL.

```
PROCEDURE DIVISION.
  PERFORM rotinal.
  PERFORM rotina2.
  STOP RUN.
rotinal.
  DISPLAY "ROTINA1".
fimrotinal.
  DISPLAY "FIM DA ROTINA1".
rotina2 SECTION.
  DISPLAY "ROTINA2".
fimrotina2.
  DISPLAY "FIM DA ROTINA2".
rotina3 SECTION.
  DISPLAY "ROTINA3".
fimrotina3.
  DISPLAY "FIM DA ROTINA3".
```

Considere, ainda, as seguintes afirmações sobre a execução dos comandos:

- I. a execução de rotinal imprimirá: "ROTINA1" e "FIM DA ROTINA1";
- II. a execução de rotina2 imprimirá: "ROTINA2" e "FIM DA ROTINA2";
- III. após o término da execução do programa serão impresos: "ROTINA1", "FIM DA ROTINA1", "ROTINA2", "FIM DA ROTINA2", "ROTINA3" e "FIM DA ROTINA3".

Sobre as afirmações, pode-se dizer que está correto o contido em:

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

51. Considere os seguintes comandos em linguagem COBOL.

```
WORKING-STORAGE SECTION.  
77 contador PIC 9(01) VALUE 1.  
PROCEDURE DIVISION.  
    PERFORM exhibe TEST BEFORE UNTIL contador > 1.  
    PERFORM exhibe TEST AFTER UNTIL contador > 1.  
    STOP RUN.  
exibe.  
    DISPLAY "EXIBE".
```

Considere também as seguintes afirmações sobre a execução dos comandos:

- I. a opção TEST AFTER faz com que o comando do parágrafo exhibe seja executado pelo menos uma vez;
- II. a opção TEST BEFORE resolve a expressão lógica (contador > 1) antes, impedindo a execução do parágrafo exhibe;
- III. após o término da execução do programa será impresso o texto "EXIBE" uma única vez.

Sobre as afirmações, pode-se dizer que está correto o conteúdo em:

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

52. Considere os seguintes comandos em linguagem COBOL.

```
WORKING-STORAGE SECTION.  
77 numero PIC 9(01) VALUE 1.  
PROCEDURE DIVISION.  
PERFORM exhibe-numeros TEST BEFORE VARYING numero  
    FROM 1 BY 2 UNTIL numero > 7.  
STOP RUN.  
exibe-numeros.  
    DISPLAY numero.
```

Indique a seqüência de números resultante da execução dos comandos.

- (A) 01, 03 e 05.
- (B) 01, 03, 05 e 07.
- (C) 01, 03, 05, 07 e 09.
- (D) 01, 02, 03, 04, 05 e 06.
- (E) 01, 02, 03, 04, 05, 06 e 07.

53. Considere os seguintes comandos em linguagem COBOL.

```
WORKING-STORAGE SECTION.  
77 ws-campo1 PIC X(05) VALUE "COBOL".  
77 ws-saida1 PIC X(05) VALUE SPACES.  
77 ws-saida2 PIC X(05) VALUE SPACES.  
PROCEDURE DIVISION.  
    USTRING ws-campo1 DELIMITED BY "B"  
    INTO ws-saida1 ws-saida2.
```

Considere também as seguintes afirmações sobre a execução dos comandos:

- I. os valores das variáveis são: ws-saida1 = "CO" e ws-saida2 = "BOL";
- II. a função do comando USTRING é extrair caracteres de uma string, armazenando os resultados em campos receptores;
- III. as opções DELIMITER e COUNT do comando USTRING podem ser utilizadas apenas com o DELIMITED BY.

Sobre as afirmações, pode-se dizer que está correto o conteúdo em:

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

54. Em um banco de dados relacional, um atributo que pode ser dividido em outros atributos é chamado de atributo

- (A) associativo.
- (B) composto.
- (C) elementar.
- (D) relacional.
- (E) restritivo.

55. Em um banco de dados relacional, um atributo multivalorado de um conjunto de entidades pode conter, considerando o conjunto de valores associado a esse atributo,

- (A) apenas um valor.
- (B) apenas o valor nulo.
- (C) apenas valores numéricos.
- (D) exatamente dois valores.
- (E) um número qualquer de valores.

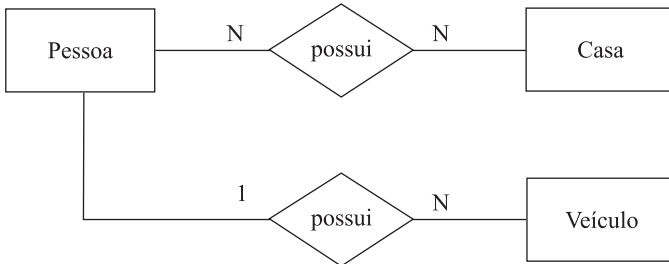
56. Considere as seguintes afirmações a respeito de dois conjuntos de entidades A e B em um banco de dados relacional:

- I. só pode haver um único relacionamento binário entre os dois conjuntos de entidades;
- II. pode haver mais de um relacionamento entre os dois conjuntos de entidades;
- III. qualquer relacionamento entre esses conjuntos de entidades pode conter atributos descritivos.

Dessas afirmações, pode-se dizer que está correto o contido em

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

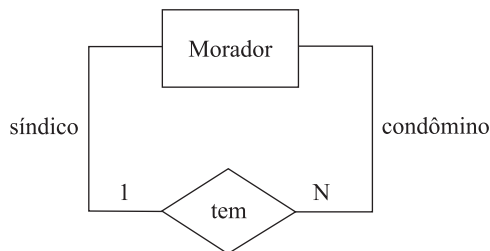
57. Considere o seguinte diagrama entidade-relacionamento de um banco de dados relacional.



A partir desse diagrama, é possível afirmar que

- (A) um veículo pode ser possuído por várias pessoas.
- (B) uma casa só pode ser possuída por uma pessoa.
- (C) uma pessoa só pode possuir duas casas.
- (D) uma pessoa pode possuir vários veículos.
- (E) uma pessoa tem, obrigatoriamente, vários veículos.

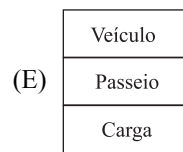
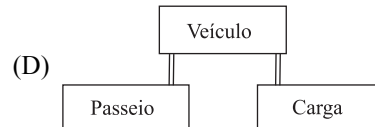
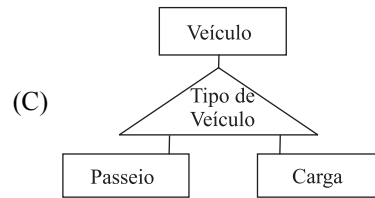
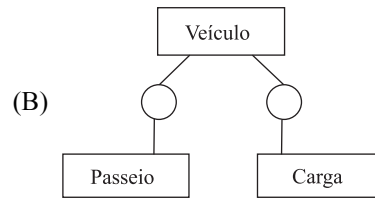
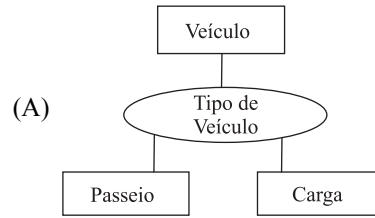
58. Considere o seguinte diagrama entidade-relacionamento de um banco de dados relacional.



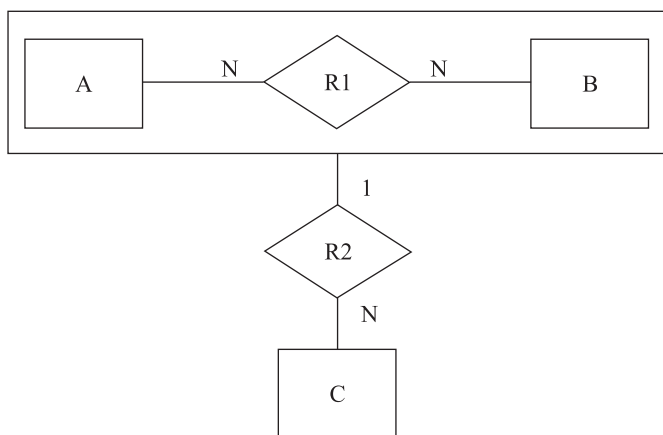
Esse diagrama está representando

- (A) um auto-relacionamento.
- (B) um relacionamento histórico.
- (C) um relacionamento ternário.
- (D) uma agregação.
- (E) uma herança.

59. Suponha que no projeto de um banco de dados relacional exista uma entidade chamada **Veículo**, à qual se deseja representar especializações, tais como **Passeio** e **Carga**. A forma de representar tal característica em um diagrama entidade-relacionamento é:



60. Considere o seguinte diagrama entidade-relacionamento de um banco de dados relacional.



O retângulo que abrange as entidades A e B e o relacionamento R1 está indicando uma

- (A) agregação.
 (B) desnormalização.
 (C) indexação.
 (D) normalização.
 (E) multivaloração.
61. Considerando um banco de dados relacional, na _____ Forma Normal, cada coluna de cada tabela é composta por _____ em cada linha.
- As palavras que completam, correta e respectivamente, as lacunas do texto são:
- (A) Primeira ... um único valor
 (B) Primeira ... Valores Múltiplos
 (C) Segunda ... dois valores
 (D) Segunda ... valores múltiplos
 (E) Terceira ... três valores
62. A teoria das dependências funcionais é utilizada para a definição de algumas formas normais em bancos de dados relacionais. A seguinte notação dessa teoria
- $$A \rightarrow B$$
- significa que, sendo A e B atributos de uma tabela T, o atributo
- (A) A não é chave primária da tabela T.
 (B) A é funcionalmente dependente do atributo B.
 (C) B deve ser indexado.
 (D) B é chave primária da tabela T.
 (E) B é funcionalmente dependente do atributo A.

63. Considere a definição: *uma tabela relacional está em determinada forma normal se e somente se estiver na primeira forma normal e qualquer atributo fora da chave primária for dependente funcional total em relação à chave primária*. Essa definição corresponde à

- (A) Segunda Forma Normal.
 (B) Terceira Forma Normal.
 (C) Quarta Forma Normal.
 (D) Forma Normal de Boyce Codd.
 (E) Forma Normal de Projeção-Junção.

64. Considere a seguinte dependência funcional.

Código, Quantidade \rightarrow Quantidade

Esse é um caso de dependência funcional

- (A) decomposta.
 (B) periódica.
 (C) transitiva.
 (D) trivial.
 (E) secundária.
65. A definição da quarta forma normal baseia-se no conceito de
- (A) classe.
 (B) herança.
 (C) polimorfismo.
 (D) impedância mútua.
 (E) dependências multivaloradas.

O enunciado a seguir deve ser utilizado para responder às questões de números **66** e **67**.

Deseja-se criar uma tabela chamada Cliente, com os atributos Nome, CPF e Cidade de nascimento, sendo CPF a chave primária da tabela.

66. Um comando SQL para criar a tabela Cliente é:

- (A) COMMIT TABLE Cliente
(CPF char(11) not null, PK (CPF),
Nome char(30),
Cidade char(20))
- (B) COPY TABLE Cliente
(CPF char(11) index in,
Nome char(30),
Cidade char(20),
PRIMARY KEY (CPF))
- (C) CREATE TABLE Cliente
(CPF char(11) not null,
Nome char(30),
Cidade char(20),
PRIMARY KEY (CPF))
- (D) PASTE TABLE Cliente
(CPF char(11) index in,
Nome char(30),
Cidade char(20),
PK (CPF))
- (E) ROLLBACK TABLE Cliente
(CPF char(11),
Nome char(30),
Cidade char(20))

67. Com base no enunciado fornecido, um comando para criar um índice chamado Ind-Nome sobre a coluna Nome da tabela Cliente é:

- (A) CREATE INDEX Ind-Nome
ON Cliente (Nome)
- (B) CREATE PREINDEX Ind-Nome
BY Cliente (Nome)
- (C) MAKE TRIGGER Ind-Nome
BY Cliente (Nome)
- (D) PASTE TRIGGER Ind-Nome
IN Cliente (Nome)
- (E) PRINT COMMIT Ind-Nome
IN Cliente (Nome)

A seguinte descrição deve ser utilizada para responder às questões de números **68** a **72**.

Considere uma tabela chamada Produtos, composta pelos campos ID, Nome, Categoria e Peso, sendo ID a chave primária. Considere ainda que ID, Nome e Categoria são do tipo caractere e Peso do tipo numérico.

68. Com base no enunciado fornecido, o comando SQL para obter o Nome dos Produtos com peso inferior a 200 gramas é:

- (A) SELECT Nome
FROM Produtos
HAVING Peso BELOW 200
- (B) SELECT Nome
FROM Produtos
HAVING Peso LIKE 200
- (C) SELECT Nome
FROM Produtos
HAVING Peso UNTIL 200
- (D) SELECT Nome
FROM Produtos
WHERE Peso < 200
- (E) SELECT Nome
FROM Produtos
WHERE Peso INF 200

69. Com base no enunciado fornecido, o comando SQL para obter o número de registros da tabela Produtos é:

- (A) SELECT AVG (&)
FROM Produtos
- (B) SELECT COUNT (*)
FROM Produtos
- (C) SELECT INT (*)
FROM Produtos
- (D) SELECT MAX (&)
FROM Produtos
- (E) SELECT SUM (&)
FROM Produtos

70. Com base no enunciado fornecido, o comando SQL para obter o ID e o Nome dos Produtos cujos nomes comecem por “Pre” é:

- (A) SELECT ID, Nome
FROM Produtos
HAVING Nome COMMON Pre\$
- (B) SELECT ID, Nome
FROM Produtos
ORDER BY Nome LIKE “Pre@”
- (C) SELECT ID, Nome
FROM Produtos
WHERE Nome LIKE “Pre%”
- (D) SELECT ID, Nome
FROM Produtos
GROUP BY Nome = Pre&
- (E) SELECT ID, Nome
FROM Produtos
HAVING Nome = Pre\$

71. Com base no enunciado fornecido, o comando SQL para obter o ID dos Produtos cujos valores de Peso estejam vazios, ou seja, não estejam preenchidos, é:

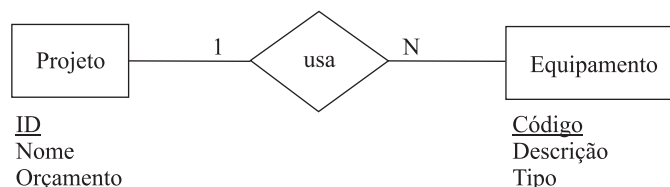
- (A) SELECT ID
FROM Produtos
WHERE Peso = 0
- (B) SELECT ID
FROM Produtos
HAVING Peso < NULL
- (C) SELECT ID
FROM Produtos
WHERE Peso IS NULL
- (D) SELECT ID
FROM Produtos
HAVING Peso LIKE “NULL”
- (E) SELECT ID
FROM Produtos
HAVING Peso LESS THAN NULL

72. Com base no enunciado fornecido, o comando SQL para obter cada Categoria e o peso médio dos produtos de cada Categoria, é

- (A) SELECT Categoria, AVG (Peso)
FROM Produtos
GROUP BY Categoria
- (B) SELECT Categoria, AVG (Peso)
FROM Produtos
HAVING BY Categoria
- (C) SELECT Categoria, AVG (Peso)
FROM Produtos
INCLUDING Categoria
- (D) SELECT Categoria, AVG (Peso)
FROM Produtos
ORDER BY Categoria
- (E) SELECT Categoria, AVG (Peso)
FROM Produtos
WHERE Categoria

A seguinte descrição deve ser utilizada para responder às questões de números 73 a 75.

Considere o seguinte diagrama entidade-relacionamento.



As tabelas que implementam esse diagrama entidade-relacionamento são:

- Projeto (ID, Nome, Orçamento)
- Equipamento (Código, Descrição, Tipo, ID)

73. Com base na descrição fornecida, o comando SQL para obter o Nome dos Projetos e o Código dos Equipamentos usados é:

- (A) SELECT Nome, Código
FROM Projeto, Equipamento
- (B) SELECT Nome, Código
FROM Projeto, Equipamento
JOIN P.ID = E.ID
- (C) SELECT Nome, Código
FROM Projeto, Equipamento
JOIN Projeto.ID = Equipamento.ID
- (D) SELECT Nome, Código
FROM P, E
WHERE P.ID = E.ID
- (E) SELECT Nome, Código
FROM Projeto, Equipamento
WHERE Projeto.ID = Equipamento.ID

74. Com base na descrição fornecida, o comando SQL para obter o valor total dos orçamentos de todos os projetos é:
- (A) `SELECT ALL(Orçamento)`
`FROM Projeto`
 - (B) `SELECT AVG(Orçamento)`
`FROM Projeto`
 - (C) `SELECT COUNT(Orçamento)`
`FROM Projeto`
 - (D) `SELECT MIN&MAX(Orçamento)`
`FROM Projeto`
 - (E) `SELECT SUM(Orçamento)`
`FROM Projeto`
75. Com base na descrição fornecida, o comando SQL para obter o Tipo e a Descrição dos equipamentos utilizados no projeto de nome "Alfa" é:
- (A) `SELECT Tipo, Descrição`
`FROM Projeto, Equipamento`
`WHERE Projeto.ID = Equipamento.ID AND`
`Nome = "Alfa"`
 - (B) `SELECT Tipo, Descrição`
`FROM Projeto, Equipamento`
`COUNTING P.ID = E.ID AND`
`Nome = "Alfa%"`
 - (C) `SELECT Tipo, Descrição`
`FROM Equipamento`
`WHERE Projeto.ID = Equipamento.ID OR`
`Nome LIKE Alfa%`
 - (D) `SELECT Tipo, Descrição`
`FROM Equipamento`
`HAVING P.ID = E.ID OR`
`Nome = "Alfa"`
 - (E) `SELECT Tipo, Descrição`
`FROM Equipamento`
`GROUP BY P.ID = E.ID OR`
`Nome → "Alfa"`
76. Os dicionários de dados presentes nos bancos de dados relacionais, geralmente, contêm
- (A) a especificação completa do banco de dados, incluindo a interface homem-máquina.
 - (B) as figuras componentes do diagrama entidade-relacionamento completo do banco de dados.
 - (C) os indicadores do grau de normalização do banco de dados.
 - (D) os ponteiros para todas as figuras eventualmente armazenadas no banco de dados.
 - (E) dados sobre as tabelas do banco de dados, como por exemplo, o nome dos atributos de cada tabela.

77. Há um tipo de tabela que constitui uma implementação virtual, não sendo implementada fisicamente no banco de dados. Esse tipo de tabela é conhecido como
- (A) visão.
 - (B) agregação.
 - (C) normalização.
 - (D) especialização.
 - (E) materialização.
78. Os valores de _____ possibilitam a distinção de cada entidade de forma individual.
A alternativa que completa corretamente a lacuna do texto é:
- (A) um atributo determinante
 - (B) um atributo não determinante
 - (C) qualquer atributo derivado
 - (D) qualquer campo numérico
 - (E) todos os relacionamentos
79. Considere a seguinte tabela de uma base de dados.
- Países (Nome, Continente, População)
- Foi realizada a seguinte consulta SQL sobre essa tabela:
- ```
SELECT DISTINCT Nome, Continente
FROM Países
WHERE Continente NOT IN (Ásia)
ORDER BY 1
```
- Essa consulta seleciona nomes de países
- (A) da Ásia, ordenados pelo nome dos países.
  - (B) e seus continentes, com exceção dos países da Ásia, ordenados pelo nome dos continentes.
  - (C) e seus continentes, com exceção dos países da Ásia, ordenados pelo nome dos países.
  - (D) e seus continentes, de todos os continentes, ordenados pelo nome dos continentes.
  - (E) e seus continentes, de todos os continentes, ordenados pelo nome do país.

80. Considere a seguinte consulta feita em SQL.

```
SELECT A, B
FROM X
WHERE A > 20 OR B IS NOT NULL
```

Indique qual das alternativas a seguir apresenta uma consulta que produz o mesmo resultado.

(A) SELECT A, B  
FROM X  
WHERE A > 20 AND B IS NOT NULL

(B) SELECT A, B  
FROM X  
WHERE A,B > 20, IS NOT NULL

(C) SELECT A, B  
FROM X  
WHERE A > 20  
INTERSECT  
SELECT A, B  
FROM X  
WHERE B IS NOT NULL

(D) SELECT A, B  
FROM X  
WHERE A > 20  
MINUS  
SELECT A, B  
FROM X  
WHERE B IS NOT NULL

(E) SELECT A, B  
FROM X  
WHERE A > 20  
UNION  
SELECT A, B  
FROM X  
WHERE B IS NOT NULL

**RASCUNHO**

## RASCUNHO

## RASCUNHO

