

INSCRIÇÃO	TURMA	NOME DO CANDIDATO
ASSINO DECLARANDO QUE LI E COMPREENDI AS INSTRUÇÕES ABAIXO:		ORDEM

107 – Ciência da Computação

INSTRUÇÕES

1. Confira, acima, o seu número de inscrição, turma e nome. Assine no local indicado.
2. Aguarde autorização para abrir o caderno de prova. Antes de iniciar a resolução das questões, confira a numeração de todas as páginas.
3. Esta prova é constituída de 20 questões objetivas.
4. Nesta prova, as questões objetivas são de múltipla escolha, com 5 alternativas cada uma, sempre na sequência **a, b, c, d, e**, das quais somente uma deve ser assinalada.
5. A interpretação das questões é parte do processo de avaliação, não sendo permitidas perguntas aos aplicadores de prova.
6. Ao receber o cartão-resposta, examine-o e verifique se o nome impresso nele corresponde ao seu. Caso haja qualquer irregularidade, comunique-a imediatamente ao aplicador de prova.
7. O cartão-resposta deverá ser preenchido com caneta esferográfica preta, tendo-se o cuidado de não ultrapassar o limite do espaço para cada marcação.
8. Não será permitido ao candidato:
 - a) Manter em seu poder relógios e aparelhos eletrônicos ou qualquer objeto identificável pelo detector de metais. Tais aparelhos deverão ser desligados e colocados OBRIGATORIAMENTE dentro do saco plástico, que deverá ser acomodado embaixo da carteira ou no chão. É vedado também o porte de armas.
 - b) Usar bonés, gorros, chapéus ou quaisquer outros acessórios que cubram as orelhas.
 - c) Usar fone ou qualquer outro dispositivo no ouvido. O uso de tais dispositivos somente será permitido quando indicado para o atendimento especial.
 - d) Levar líquidos, exceto se a garrafa for transparente e sem rótulo.
 - e) Comunicar-se com outro candidato, usar calculadora e dispositivos similares, livros, anotações, régua de cálculo, impressos ou qualquer outro material de consulta.
 - f) Portar carteira de documentos/dinheiro ou similares.
 - g) Usar óculos escuros, ressalvados os de grau, quando expressamente por recomendação médica, devendo o candidato, então, respeitar o subitem 5.5.5 do Edital.
 - h) Emprestar materiais para realização das provas.

Caso alguma dessas exigências seja descumprida, o candidato será excluído do processo.

9. A duração da prova é de 4 horas. Esse tempo inclui a resolução das questões e a transcrição das respostas para o cartão-resposta.
10. Ao concluir a prova, permaneça em seu lugar e comunique ao aplicador de prova. Aguarde autorização para entregar o caderno de prova e o cartão-resposta.
11. Se desejar, anote as respostas no quadro abaixo, recorte na linha indicada e leve-o consigo.

DURAÇÃO DESTA PROVA: 4 horas

✂.....

RESPOSTAS			
01 -	06 -	11 -	16 -
02 -	07 -	12 -	17 -
03 -	08 -	13 -	18 -
04 -	09 -	14 -	19 -
05 -	10 -	15 -	20 -

Conhecimentos
Específicos

01 - Se a expressão $((2 + 3)/4 + 5 * (6 + 7) * 8) + 9$ é avaliada com * e / tendo precedência sobre +, então o valor obtido é o mesmo que o valor de qual das seguintes expressões pré-fixas?

- a) + + / + 2 3 4 * * 5 + 6 7 8 9
- b) + * + + 2 3 4 / * 5 + 6 7 8 9
- c) / + + 2 3 4 * * 5 + + 6 7 8 9
- d) * + + + 2 3 4 * / 5 + 6 7 8 9
- e) + * + / 2 3 4 + + 5 * 6 7 8 9

02 - Assinale a alternativa que representa o resultado da execução do programa ao lado.

- a) 1 2
- b) 2 1
- c) 1 1
- d) 2 2
- e) 0 1

```

Program P;

Var X, Y : Integer;

Procedure P(Var A: Integer; B: Integer);
Begin
    A := A + 1;
    B := B + 1;
End;

Begin
    X := 1;
    Y := 1;
    P(X, Y);
    WriteLn(X, ' ', Y);
End.

```

03 - Considere o seguinte trecho de código ao lado. Qual é o valor em meuArray[1][2]?

- a) 1
- b) 2
- c) 3
- d) 4
- e) 5

```

int i, j;

int cont = 0;

int meuArray[2][3];

for (i=0; i<3; i++)
    for (j=0; j<2; j++)
    {
        meuArray[j][i] = cont;
        cont++;
    }

```

04 - Se x, y e z são variáveis booleanas, considere as seguintes igualdades:

1. $x \text{ e } (y \text{ ou } z) = (x \text{ e } y) \text{ ou } (x \text{ e } z)$
2. $x \text{ ou } (y \text{ e } z) = (x \text{ ou } y) \text{ e } (x \text{ ou } z)$
3. $(y \text{ e } x) \text{ ou } z = (z \text{ e } x) \text{ ou } (z \text{ e } y)$

Nesse caso, está/estão correta(s):

- a) 1 somente.
- b) 2 somente.
- c) 1 e 2 somente.
- d) 2 e 3 somente.
- e) 1, 2 e 3.

05 - Nas declarações ao lado, o tipo Polinômio representa o polinômio $a_0X^0 + a_1X^1 + \dots + a_NX^N$ por um vetor $A[0..N]$ de tal maneira que $A[i] = a_i$ para cada $0 \leq i \leq N$. A chamada $\text{Produto}(P, Q)$ deve devolver o produto dos polinômios P e Q , assumindo que as somas dos seus graus não ultrapassa N .

Segundo essas convenções, assinale a alternativa correta para preencher o trecho indicado com ???.

- a) $R[I] + P[J] * Q[J]$
- b) $R[I] + P[J] * Q[N-J]$
- c) $R[J] + P[J] * Q[N-I]$
- d) $R[J] + P[I] * Q[N-I]$
- e) $R[I] + P[J] * Q[I-J]$

```
Const N = 1024;

Type Polinomio = Array [0..N] of Integer;

Function Produto(P, Q : Polinomio) : Polinomio;

Var
    I, J : Integer;
    R : Polinomio;

Begin
    For I:=0 to N do
        Begin
            R[I] := 0;
            For J:=0 to I do
                R[I] := ???;
            End;
        End;

    Produto := R;
End.
```

06 - Considere o seguinte programa:

```
#include <stdio.h>

int main (void)
{
    char nome[25], v[6];
    int j, i = 0, c[5];
    v[0] = 'a'; v[1] = 'e'; v[2] = 'i'; v[3] = 'o'; v[4] = 'u'; v[5] = '\0';
    for (i = 0; i < 5; i++)
        c[i]=0;

    gets(nome);
    i = 0;
    while (nome[i]) {
        for (j = 0; v[j]; j++)
            if (nome[i] == v[j])
                c[j]++;

        i++;
    }

    for (i = 0; i < 5; i++)
        printf ("%d", c[i]);

    return 0;
}
```

O que será impresso caso sejam fornecidos para o programa os seguintes caracteres no vetor `nome`?

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
H	o	j	e		o	c	o	r	r	e	u		u	m	a		p	r	o	v	a	\0		

- a) 0 2 3 4 2.
- b) 1 0 5 5 2.
- c) 2 2 0 4 2.
- d) 2 3 2 7 1.
- e) 2 4 6 3 1.

07 - Assinale a alternativa que indica o valor devolvido pela chamada F(N), em que F é a função ao lado.

- a) N^2 .
- b) $N(N+1)/2$.
- c) $N^2/2$.
- d) N^2+N .
- e) N^2-N .

```
Function F(N : Integer) : Integer;

Var
  I, J, C : Integer;

Begin
  C := 0;

  For I:=1 to N do
    For J:=I to N do
      C := C+1;

  F := C;
End;
```

08 - Considere os seguintes problemas:

1. Reduzir o número de portas lógicas no somador.
2. Reduzir o tempo de propagação dos sinais.
3. Aumentar a precisão.
4. Garantir que o resultado seja correto.

Num circuito somador para números inteiros representados em complemento de dois, emprega-se a técnica de adiantamento do vai-um para resolver o(s) problema(s):

- a) 1 apenas.
- b) 2 apenas.
- c) 1 e 3 apenas.
- d) 2 e 4 apenas.
- e) 3 e 4 apenas.

09 - Considere o seguinte pseudocódigo ao lado:

Assinale a alternativa correta para a interpretação desse algoritmo.

- a) O algoritmo está calculando a multiplicação dos números inseridos no vetor "vet" de 10 posições, realizando o cálculo no aninhamento da primeira estrutura de repetição, armazenando esses novos valores no vetor "geravet" de 10 posições e imprimindo esses números armazenados na última estrutura de repetição.
- b) O algoritmo está calculando a multiplicação dos números inseridos no vetor "vet" de 10 posições, realizando o cálculo no aninhamento da primeira estrutura de repetição, armazenando esses novos valores na variável "j", sempre acrescentando o valor 1 nas posições atuais do vetor "geravet".
- c) O algoritmo está calculando o fatorial dos números inseridos no vetor "vet" de 10 posições, realizando o cálculo no aninhamento da primeira estrutura de repetição, armazenando esses novos valores no vetor "geravet" de 10 posições e imprimindo esses números armazenados na última estrutura de repetição.
- d) O algoritmo está calculando o fatorial dos números inseridos no vetor "vet" de 10 posições, realizando um cálculo no aninhamento da primeira estrutura de repetição, armazenando esses novos valores no mesmo vetor "vet" de 10 posições e imprimindo esses números armazenados na última estrutura de repetição.
- e) O algoritmo está calculando o fatorial dos números inseridos no vetor "vet" de 10 posições, realizando o cálculo no aninhamento da primeira estrutura de repetição, armazenando esses novos valores no vetor "geravet" de 10 posições e imprimindo esse vetor adicionando o valor 1 a cada iteração.

```
programa vetor
  int vet[10], geravet[10], i, j;
  para (i := 0, i < 10; i++)
    inicio
      imprimir ("digite o ", i + 1, "º elemento: ");
      ler vet[i];
      geravet[i] := 1;
      para (j := 2; j <= vet[i]; j++)
        geravet[i] := geravet[i] * j;
    fim-para;
  para (i := 0, i < 10; i++)
    inicio
      imprimir;
      imprimir (geravet[i]);
    fim-para;
  fim vetor.
```

10 - Assinale a alternativa correta com relação às declarações ao lado.

- a) Se ??? for substituído por >, a chamada F(V) devolve um elemento máximo de V.
- b) Se ??? for substituído por >, a chamada F(V) devolve um elemento mínimo de V.
- c) Se ??? for substituído por >=, a chamada F(V) devolve um elemento mínimo de V.
- d) Se ??? for substituído por <, a chamada F(V) devolve um elemento mínimo de V.
- e) Se ??? for substituído por >=, a chamada F(V) devolve um elemento máximo de V.

```
Const N = 1024;

Type Vetor = Array [1..N] of Integer;

Function F(V : Vetor) : Integer;

Var
    A, B : Integer;

Begin
    B := V[1];
    For A:=2 to N do
        If V[A] ??? B Then
            B := V[A];

    F := B;
End;
```

11 - Considerando o Mapa de Karnaugh mostrado ao lado, qual é o menor número de agrupamentos possível?

- a) 5.
- b) 4.
- c) 3.
- d) 2.
- e) 1.

1	0	0	1
0	1	0	0
0	0	0	0
1	0	0	1

12 - Considere uma função lógica de quatro variáveis – F(a,b,c,d) –, que é descrita pelos mintermos 0,1,2,7,9,10 e 15. Assinale a alternativa que apresenta a versão de F simplificada com um Mapa de Karnaugh.

(O complemento – negação – de uma variável é indicado por um ponto de exclamação, a disjunção é representada por '+' e a conjunção por '.').

- a) !a . !b . c + b . !c . d + !b . !c . !d + !b . c . !d
- b) !a . !b . c + a . b . d + b . !c . d + !b . c . !d
- c) !a . !b . !c + a . !b . d + b . c . d + !b . !c . !d
- d) !a . !b . c + b . !c . d + !b . c . !d + !b . c . d
- e) !a . !b . !c + b . c . d + !b . c . !d + !b . !c . d

13 - A expressão booleana $A+AB'+AB'C$ depende de:

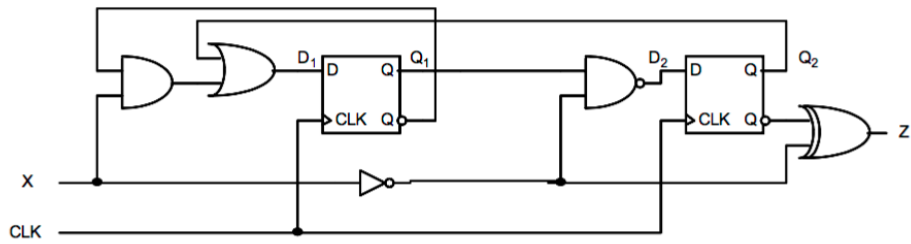
- a) A apenas.
- b) A, B e C.
- c) C apenas.
- d) B apenas.
- e) B e C apenas.

14 - Considere o programa ao lado. Ele imprime:

- a) -1 -55
- b) 0 -55
- c) -10 -55
- d) -55 -10
- e) -55 0

```
#include <stdio.h>
int main (void)
{
    int v[10] = {-1, -2, -3, -4, -5, -6, -7, -8, -9, -10};
    int i, maior, s;
    maior = s = 0;
    for (i = 0; i < 10; i++) {
        s = s + v[i];
        if (v[i] > maior )
            maior = v[i];
    }
    printf ("%d %d \n", maior, s);
    return 0;
}
```

15 - Considere a máquina de estados abaixo:



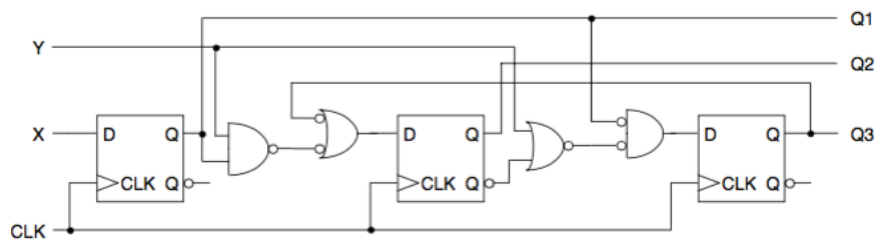
Sobre essa máquina, considere as seguintes afirmativas:

1. Trata-se de uma máquina de Mealy.
2. Trata-se de uma máquina de Moore.
3. Pode ser alterada para funcionar somente com portas NOR.
4. Pode ser alterada para funcionar somente com portas NAND.

Assinale a alternativa correta.

- a) Somente a afirmativa 1 é verdadeira.
- b) Somente a afirmativa 2 é verdadeira.
- c) Somente as afirmativas 1 e 3 são verdadeiras.
- d) Somente as afirmativas 2 e 4 são verdadeiras.
- e) Somente as afirmativas 1, 3 e 4 são verdadeiras.

16 - Considere a máquina de estados abaixo:



Sobre essa máquina, considere as seguintes afirmativas:

1. Trata-se de uma máquina de Mealy.
2. Trata-se de uma máquina de Moore.
3. Pode ser alterada para funcionar somente com portas NOR.
4. Pode ser alterada para funcionar somente com portas NAND.

Assinale a alternativa correta.

- a) Somente a afirmativa 1 é verdadeira.
- b) Somente a afirmativa 2 é verdadeira.
- c) Somente as afirmativas 1 e 3 são verdadeiras.
- d) Somente as afirmativas 2 e 4 são verdadeiras.
- e) Somente as afirmativas 2, 3 e 4 são verdadeiras.

17 - Qual é a simplificação da função booleana $Y = (AB')' + A'BC' + (A+C')' + B$?

- a) $A' + B$
- b) B
- c) A'
- d) A
- e) $(A+B)'$

18 - No código ao lado, IntSet é um tipo abstrato de dados para conjuntos de inteiros.

Qual das seguintes operações abstratas sobre “conjunto de inteiros” ele implementa?

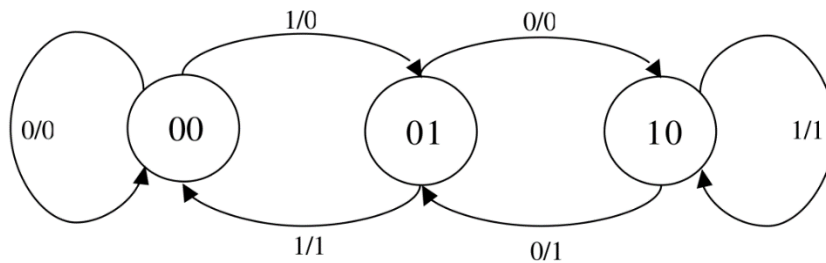
- a) Adiciona x a S .
- b) Remove x de S .
- c) Faz a interseção de $\{x\}$ e S .
- d) Faz a união de $\{x\}$ e S .
- e) Faz uma cópia de S .

```

Procedure P(var S : IntSet, x : integer);
  var k : integer;
begin
  k := 1;
  while k <= S.Last do
    begin
      if S.V[k] = x then
        begin
          S.V[k] := S.V[S.Last];
          S.Last := S.Last - 1;
        end
      else
        k := k + 1;
      end
    end
  end
end

```

19 - Considere a máquina de estados de Mealy a seguir e usando dois flip-flops tipo D, com saídas Q1 e Q0, respectivamente.



Denominando X a entrada do circuito, qual das equações abaixo representa o comportamento da saída?

- a) $Q1 + X \cdot Q0$
- b) $X \cdot Q1 \cdot Q0 + X \cdot Q1$
- c) $Q0 + X \cdot Q1$
- d) $X \cdot Q0 + X \cdot Q1$
- e) $X \cdot Q1 + X \cdot Q1 \cdot Q0$

20 - Assinale a alternativa que apresenta o número de linhas que serão escritas na chamada de F(4), em que F é a função ao lado.

- a) 12.
- b) 11.
- c) 9.
- d) 7.
- e) 6.

```

Function F(X : Real) : Real;
Var
  A, B, R : Real;
Begin
  A := 0;
  B := X;
  While B-A > 0.01 Do
    Begin
      R := (A+B)/2;
      WriteLn(R);
      If R*R > X Then
        B := R
      Else
        A := R
      End;
    End;
  F := R;
End;

```