



LÓGICA DE PROGRAMAÇÃO
AULA 4

... - Assinale a alternativa que contém o valor final da variável X, após a execução do trecho de programa em Português Estruturado mostrado abaixo. Considere os seguintes valores iniciais para as variáveis: X = 1; A = 5.

```
enquanto (A < 9) faça
    se (X < 6) então
        X ← X * 2
    fim_se
    A ← A + 1
fim_enquanto
```

- a) 4
- b) 6
- c) 8
- d) 16

84 – Associe a coluna da direita com a da esquerda e, em seguida, marque a alternativa que contém a seqüência correta.

- | | | |
|---------------------------------------------------------|-----|-----------------------------------------------------------|
| 1- Desvio Condicional Simples | () | enquanto ... faça ...
fim_enquanto |
| 2- Desvio Condicional Composto | () | se ... então ... fim_se |
| 3- Repetição do Tipo: Teste lógico no início do looping | () | repita ... até_que |
| 4- Repetição do Tipo: Teste lógico no fim do looping | () | para ... de ... até ...
passo ... faça ...
fim_para |
| 5- Repetição do Tipo: Variável de controle | () | se ... então ... senão ... |
-
- a) 1, 2, 3, 5, 4
b) 3, 1, 4, 5, 2
c) 3, 5, 4, 1, 2
d) 4, 2, 3, 5, 1

– Assinale a alternativa que contém o texto impresso no final da execução do programa em Português Estruturado abaixo. Considere o valor inicial da variável $X = 5$.

```
X ← X + 2
se (X = 5) então
    escreve “sete”
fim_se
se (X = 7) então
    escreve “cinco”
fim_se
Y ← 0
enquanto (Y < X) faça
    escreve “V”
    Y ← Y + 1
fim_enquanto
```

- a) cincoVVVVVVV
- b) seteVVVVVVV
- c) cincoVVVVV
- d) seteVVVVV

- Interprete o algoritmo e responda qual seqüência de números será impressa:

```
programa prova
var
    x,r : inteiro
    cont: inteiro
inicio
    cont ← 1
    x ← 0
    enquanto (cont <= 5) faça
        r ← x*3
        escreva r
        cont ← cont + 1
        x ← x+1
    fim-enquanto
fim
```

- a) 3,6,9,12,15
- b) 0,9,12,15,30
- c) 0,6,9,12,15
- d) 0,3,6,9,12

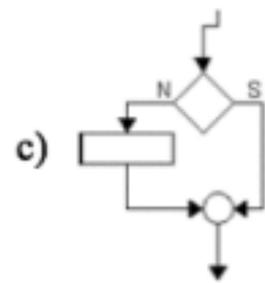
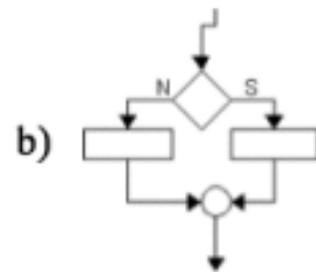
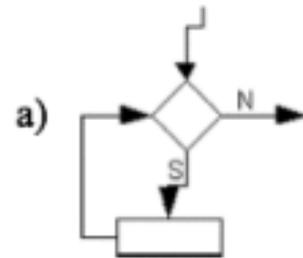
– Analise o algoritmo do programa abaixo, em português estruturado, e assinale qual será o valor final da variável X no momento da escrita.

```
programa SURPRESA
var
    CONTADOR : inteiro
    X : inteiro
inicio
    X ← 1
    CONTADOR ← 1
    enquanto (CONTADOR <= 5) faça
        X ← X * CONTADOR
        CONTADOR ← CONTADOR + 1
    fim_enquanto
    escreva "O valor de X é =", X
fim
```

- a) 5
- b) 24
- c) 1
- d) 120

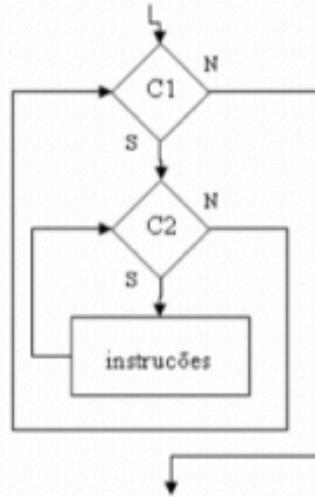
– Assinale a alternativa que apresenta o diagrama de blocos descrito a seguir.

“Caracteriza-se por uma estrutura que efetua um teste lógico no início de um *looping*, verificando se é permitido executar o trecho de instruções subordinado a esse *looping*”.



– Seleccione a alternativa com o trecho de programa em português estruturado correspondente ao diagrama de blocos da figura abaixo.

Considere: C1 = Condição 1 e C2 = Condição 2.



a) enquanto (<C1>) faça
 enquanto (<C2>) faça
 <instruções>
 fim_enquanto
fim_enquanto

b) enquanto (<C1>) faça
 repita
 <instruções>
 até_que (<C2>)
fim_enquanto

c) repita
 repita
 <instruções>
 até_que (<C2>)
até_que (<C1>)

d) enquanto (<C1> e <C2>) faça
 para <var> de <início> até <fim> passo <incr> faça
 <instruções>
 fim_para
fim_enquanto

– Assinale a alternativa que apresenta o que o programa SOMATÓRIO retornará na variável “TOTAL”:

Programa SOMATÓRIO

Var

TOTAL, CONTADOR : inteiro

Início

TOTAL \leftarrow 0

para CONTADOR de 1 até 50 passo 1 faça
TOTAL \leftarrow TOTAL + CONTADOR

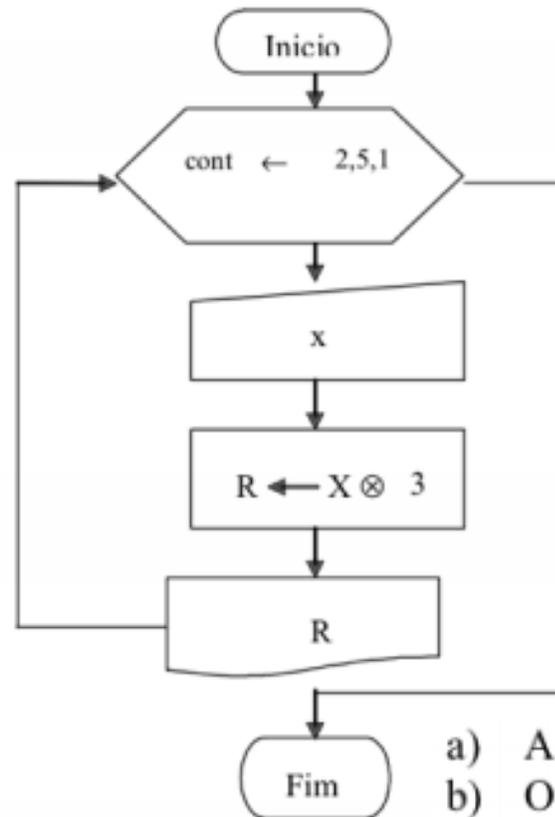
Fim-para

Escreva TOTAL

Fim

- a) Será retornado o valor 50.
- b) Será retornada a média aritmética dos primeiros 50 números inteiros.
- c) Não há como prever qual valor será retornado, pelo valor tender ao infinito.
- d) Será retornada a soma dos 50 primeiros números inteiros (1+2+3+4+ ... +47+48+49+50).

– Assinale a alternativa que retrata corretamente a utilização da estrutura de *looping* na figura abaixo, estrutura para...de...até...passo...faça...fim_para:



- a) A variável cont se inicia com 1.
- b) O incremento da variável cont é 5.
- c) A estrutura de repetição será executada 5 vezes.
- d) A estrutura de repetição pode ser usada quando houver necessidade de se repetir trechos finitos, em que os valores “inicial” e “final” são conhecidos.

```
var
    S, C: Inteiro
inicio
    S <- 0
    C <- 1
    Enquanto (C <= 10) faça
        S <- S + C
        C <- C + 4
    FimEnquanto
    Escreva(S)
fimalgoritmo
```

```
var
  X, Y, Z: Inteiro
inicio
  X <- 1
  Y <- 0
  Repita
    Se (X % 2 = 0) entao
      Z <- 5
    senao
      Z <- 8
    FimSe
    Y <- Y + Z
    X <- X + 1
  Ate (X > 4)
  Escreva (Y)
fimalgoritmo
```

```
var
  c, i, x: Inteiro
inicio
  c <- 0
  i <- 5
  Para x <- c ate 20 passo i faca
    Se (x < i + 10) entao
      Escreva(x)
    senao
      Escreva(x+i)
    FimSe
  FimPara
fimalgoritmo
```

```
var
  c1, c2, s: Inteiro
inicio
  s <- 0
  c1 <- 0
  Enquanto (c1 < 3) faca
    Para c2 <- 1 ate 9 passo 3 faca
      s <- s + 1
    FimPara
    c1 <- c1 + 1
  FimEnquanto
  Escreva (s)
fimalgoritmo
```

```
var
  m, n, z: Inteiro
inicio
  m <- 300
  n <- 50
  z <- 0
  Enquanto (z < 200) faca
    Se (m < 100) entao
      z <- z + 100
    senao
      Se (m < 200) entao
        z <- z + 10
      senao
        z <- z + n
      FimSe
    FimSe
  FimEnquanto
  Escreva (m, n, z)
fimalgoritmo
```

```
var
  peso: Real
  fator: Inteiro
  acabou: Logico
inicio
  peso <- 3.0
  fator <- 4
  acabou <- falso
  Enquanto (nao acabou) faca
    Se (peso >= 10) entao
      peso <- peso ^ fator
    FimSe
    peso <- peso + fator
    fator <- fator - 1
    Se (peso > 100) entao
      acabou <- verdadeiro
    FimSe
  FimEnquanto
  Escreva(peso)
finalgoritmo
```

```
var
  a, b, c, d: Inteiro
inicio
  a <- 3
  b <- 1
  c <- a - b
  d <- a % c
  Enquanto (d <= 4) faca
    Se ((a >= 3) e nao (d < b) xou (c > 4)) entao
      a <- a + 1
      b <- a
    senao
      c <- c + 3
    FimSe
    d <- d + 1
  FimEnquanto
  Escreva (a, b, c, d)
finalgoritmo
```

```
var
  a, b: Inteiro
inicio
  a <- 10
  b <- 5
  Enquanto (a < 20000) faça
    Se (a > 30) entao
      interrompa
    senao
      Se (a > 20) entao
        a <- a + 2
      senao
        a <- a + b
      FimSe
    FimSe
  FimEnquanto
  Escreva (a)
fimalgoritmo
```