

Ex 1.

```
1  Algoritmo "Exercicio 1"
2
3  Var
4      N, i: inteiro
5      fib: vetor[1..50] de inteiro
6
7  Inicio
8      escreva("Digite a quantidade de termos que deseja imprimir: ")
9      leia(N)
10
11      fib[1] := 0
12      fib[2] := 1
13
14      para i de 3 ate N faca
15          fib[i] := fib[i-1] + fib[i-2]
16      fimpara
17
18      para i de 1 ate N faca
19          escreva(fib[i], " ")
20      fimpara
21  FimAlgoritmo
```

Ex 2.

```
programa {  
  
    inteiro num[5] = {1,2,3,4,5}  
  
    funcao inicio() {  
        para(inteiro i = 4; i ≥ 0; i--){  
            escreva(num[i])  
        }  
    }  
}
```

Ex 3.

Programa Ex3

var

 numeros: vetor[1..5] de inteiro

 maior, menor: inteiro

 i: inteiro

inicio

 para i de 1 ate 5 faca

 escreva("Digite o numero ", i, ": ")

 leia(numeros[i])

 fimpara

 maior <- numeros[1]

 menor <- numeros[1]

 para i de 2 ate 5 faca

 se numeros[i] > maior entao

 maior <- numeros[i]

 fimse

 se numeros[i] < menor entao

 menor <- numeros[i]

 fimse

 fimpara

 escreva("Maior valor: ", maior, " | Menor valor: ", menor)

fim

Ex 4.

```
1  programa_Ex4
2
3  Var
4      n:vetor[1..5] de inteiro;
5      media: real;
6      i: inteiro
7
8  Inicio
9      para i de 1 ate 5 faca
10         escreva ("Digite o numero", i, ":");
11         leia (n[i])
12     fimpara
13
14     Media:= (n[1] + n[2] + n[3] + n[4] + n[5]) / 5;
15     Imprima ("media final =", media)
16 Fim
```

Ex 5.

```
programa {  
  
    cadeia nome[5]  
    inteiro tel[5]  
  
    funcao inicio() {  
  
        para (inteiro i = 0; i < 5; i++){  
            escreva("Posição ", i, "\n")  
            escreva ("Digite o seu nome:\n")  
            leia(nome[i])  
            escreva ("Digite o seu telefone: \n")  
            leia(tel[i])  
        }  
  
        para (inteiro i = 0; i < 5; i++){  
            escreva("Nome: ", nome[i], " ")  
            escreva("Telefone: ", tel[4-i], " \n")  
        }  
    }  
}
```

Ex 6.

```
1  Algoritmo "ex 6"
2
3
4  Var
5      numeros: vetor[1..2] de inteiro
6      resultado: vetor[1..4] de inteiro
7      divisao_valida: logico
8
9
10 Para i de 1 ate 2 faca
11     Escreva("Digite o ", i, "º número: ")
12     Leia(numeros[i])
13 FimPara
14
15
16 Se numeros[2] <> 0 entao
17     divisao_valida := verdadeiro
18 Senao
19     divisao_valida := falso
20 FimSe
21
22
23 resultado[1] := numeros[1] + numeros[2]
24 resultado[2] := numeros[1] - numeros[2]
25 resultado[3] := numeros[1] * numeros[2]
26 Se divisao_valida entao
27     resultado[4] := numeros[1] / numeros[2]
28 Senao
29     resultado[4] := 0
30 FimSe
31
32
33 Escreva("A soma dos números é: ", resultado[1])
34 Escreva("A subtração dos números é: ", resultado[2])
35 Escreva("A multiplicação dos números é: ", resultado[3])
36 Se divisao_valida entao
37     Escreva("A divisão dos números é: ", resultado[4])
38 Senao
39     Escreva("Não é possível dividir por zero")
40 FimSe
41
42 FimAlgoritmo
```

Ex 7.

algoritmo faixa-etária

var

idades: vetor[1..10] de inteiro;
i, crianca=0, jovem=0, adulto=0: inteiro;

inicio

para i de 1 ate 10 faca

escreva("Digite a idade da pessoa ", i, ": ");

leia(idades[i]);

enquanto idades[i] < 0 ou idades[i] > 120 faca

escreva("Idade inválida! Por favor, digite novamente: ");

leia(idades[i]);

fimenquanto

se idades[i] >= 0 e idades[i] <= 10 entao

crianca = crianca + 1;

senao se idades[i] >= 11 e idades[i] <= 20 entao

jovem = jovem + 1;

senao

adulto = adulto + 1;

fimse

fimpara

escreva("Número de pessoas com idade entre 0 e 10 anos: ", crianca, "\n");

escreva("Número de pessoas com idade entre 11 e 20 anos: ", jovem, "\n");

escreva("Número de pessoas com idade acima de 20 anos: ", adulto);

fimalgoritmo