

## Exercícios:

1. Faça um programa que possua um vetor denominado A que armazene 6 números inteiros. O programa deve executar os seguintes passos:
  - (a) Atribua os seguintes valores a esse vetor: 1, 0, 5, -2, -5, 7.
  - (b) Armazene em uma variável inteira (simples) a soma entre os valores das posições A[0], A[1] e A[5] do vetor e mostre na tela esta soma.
  - (c) Modifique o vetor na posição 4, atribuindo a esta posição o valor 100.
  - (d) Mostre na tela cada valor do vetor A, um em cada linha.

Algoritmo "semnome"

// Autor(a) : Pedro Henrique Figueira e Nilton

// Data atual : 08/04/2022

Var

a: vetor[1..6] de real

i: inteiro

s: real

Inicio

para i de 1 até 6 faca

escreval ("digite a nota", i, "i")

leia (a[i])

fimpara

s <- (a[1] + a[2] + a[6])

escreval ("a nota foi:", s)

Fimalgoritmo

2. Crie um programa que le<sup>6</sup> valores inteiros e, em seguida, mostre na tela os valores lidos.

Algoritmo "semnome"

// Autor(a) : Pedro Henrique Figueira e Nilton

// Data atual : 08/04/2022

Var

v: vetor [1..6] de real

i: inteiro

Inicio

para i de 1 ate 6 faca

escreval ("digite um valor", i)

leia(v[i])

fimpara

para i de 1 ate 6 faca

escreval("")

escreval("o valor", i , " igual a:", (v[i]))

fimpara

## Fimalgoritmo

3. Ler um conjunto de números reais, armazenando-o em vetor e calcular o quadrado das componentes deste vetor, armazenando o resultado em outro vetor. Os conjuntos têm 10 elementos cada. Imprimir todos os conjuntos.

### Algoritmo "semnome"

// Autor(a) : Pedro Henrique Figueira e Nilton

// Data atual : 08/04/2022

Var

v:vetor [0..9] de real

indice:inteiro

vet:vetor [0..9] de real

i: inteiro

Inicio

para indice de 0 ate 9 faca

escreva("Digite 10 numeros")

leia(v[indice])

fimpara

para i de 0 ate 9 faca

vet[i] <- v[indice]^2

escreval(vet[i])

fimpara

## Fimalgoritmo

4. Faça um programa que leia um vetor de 8 posições e, em seguida, leia também dois valores  $X$  e  $Y$  quaisquer correspondentes a duas posições no vetor. Ao final seu programa deverá escrever a soma dos valores encontrados nas respectivas posições  $X$  e  $Y$ .

Algoritmo "Vetor"

// Disciplina : [Linguagem e Lógica de Programação]

// Professor : João Carlos

```
Var
v:vetor [1..8] de real
indice:inteiro
x,y:inteiro
num1,num2,soma:inteiro
```

Inicio

```
escreva("digite o valor de x numero:")
leia(x)
escreva("digite o valor de y numero:")
leia(y)
```

```
para indice de 1 ate 8 faca
escreval("digite 8 numeros:")
leia(v[indice])
```

```
se (x=v[indice]) entao
num1 <- indice
fimse
```

```
se (y=v[indice]) entao
num2 <- indice
fimse
```

```
fimpara
```

```
soma <- num1+num2
escreva("a soma das duas posições é:",soma)
```

Fimalgoritmo

5. Leia um vetor de 10 posições. Contar e escrever quantos valores pares ele possui.

Algoritmo "semnome"

// Autor(a) : Pedro Henrique Figueira e Nilton

// Data atual : 08/04/2022

Var

v: vetor [1..5] de real

n: real

i: inteiro

Inicio

para i de 1 ate 5 faca

escreval ("digite um valor", i)

leia(v[i])

fimpara

para i de 1 ate 5 faca

se ((v[i]%2)=0) entao

escreval("")

escreval("os numeros pares são o ",  
v[i])

fimse

fimpara

Fimalgoritmo

6. Faça um programa que receba do usuário um vetor com 10 posições. Em seguida deverá ser impresso o maior e o menor elemento do vetor.

Algoritmo "semnome"

// Autor(a) : Pedro Henrique Figueira e Nilton

// Data atual : 08/04/2022

Var

v:vetor [0..9] de real

indice:inteiro

maior\_num,menor\_num:real

Inicio

para indice de 0 ate 9 faca

escreva("Digite 10 numeros")

leia(v[indice])

se maior\_num > v[indice] entao

maior\_num <- v[indice]

fimse

se menor\_num < v[indice] entao

menor\_num <- v[indice]

fimse

fimpara

escreval("o menor numero é:", menor\_num)

escreval("o maior numero é:", maior\_num)

Fimalgoritmo

7. Escreva um programa que leia 10 números inteiros e os armazene em um vetor. Imprima o vetor, o maior elemento e a posição que ele se encontra.

Algoritmo "semnome"



// Autor(a) : Pedro Henrique Figueira e Nilton

// Data atual : 08/04/2022

Var

v:vetor [1..10] de real

p: real

i:inteiro

maior\_num,menor\_num:real

Inicio

para i de 1 ate 10 faca

escreval("Digite o ", i, "º","numero")

leia(v[i])

se maior\_num < v[i] entao

maior\_num <- v[i]

p <- i

fimse

fimpara

escreval("o maior numero é:",maior\_num, " Posição: ",p)

Fimalgoritmo

8. Crie um programa que le<sup>6</sup> valores inteiros e, em seguida, mostre na tela os valores lidos na ordem inversa.

Algoritmo "semnome"

// Autor(a) : Pedro Henrique Figueira e Nilton

// Data atual : 08/04/2022

Var

v: vetor [1..6] de real

i: inteiro

Início

para i de 1 ate 6 faca

escreval ("digite um valor", i)

leia(v[i])

fimpara

para i de 1 ate 6 faca

escreval("")

fimpara

escreval("ordem ao contrario é:", (v[6]))

escreval("ordem ao contrario é:", (v[5]))

escreval("ordem ao contrario é:", (v[4]))

escreval("ordem ao contrario é:", (v[3]))

escreval("ordem ao contrario é:", (v[2]))

escreval("ordem ao contrario é:", (v[1]))

9. Faça um programa que preencha um vetor com 10 números reais, calcule e mostre a quantidade de números negativos e a soma dos números positivos desse vetor.

Algoritmo "semnome"

// Autor(a) : Pedro Henrique Figueira e nilton

// Data atual : 08/04/2022

Var

v:vetor [0..9] de real

indice:inteiro

soma,num\_negativo:real

Inicio

para indice de 0 ate 9 faca

escreva("Digite 10 numeros")

leia(v[indice])

se  $0 < v[indice]$  entao

soma <- soma+v[indice]

se 0>v[indice] entao

num\_negativo <- num\_negativo+1

fimse

fimpara

escreval("quantidade de numeros negativos:",num\_negativo)

escreval("soma dos numeros positivos:",soma)

Fimalgoritmo

10. Faça um programa que leia um vetor de 10 posições e verifique se existem valores iguais e os escreva na tela.

Algoritmo "semnome"

// Disciplina : [Linguagem e Lógica de Programação]

// Professor : João Carlos de Souza

// Autor(a) : Nilton Dionisio Guerra

// Data atual : 20/04/2022

Var

v:vetor [0..9] de real

vet:vetor [0..9] de real

vt:vetor [0..9] de real

i,ii:inteiro

repet:inteiro

Inicio

escreval("digite 10 números:")

para i de 0 ate 9 faca

leia(v[i])

fimpara

i <- 0

enquanto (i<=9) faca

ii <- i+1

enquanto (ii<=9) faca

se v[i]=v[ii] entao

vt[i] <- v[i]

fimse

ii <- ii+1

fimenquanto

i <- i+1

fimenquanto

escreval("Os numeros que você digitou foram:")

para i de 0 ate 9 faca

escreva(v[i])

fimpara

escreval(", Os numeros que são repetidos sao:")

para i de 0 ate 9 faca

escreva(vt[i])

fimpara

11. Leia um vetor com 20 números inteiros.  
Escreva os elementos do vetor eliminando elementos repetidos.

Algoritmo "semnome"

// Disciplina : [Linguagem e Lógica de Programação]

// Professor : Antonio Carlos Nicolodi

// Descrição : Aqui você descreve o que o programa faz! (função)

// Autor(a) : Nome do(a) aluno(a)

// Data atual : 20/04/2022

var

v:vetor [0..9] de real

i,ii:inteiro

Inicio

escreval("digite 10 números:")

para i de 0 ate 9 faca

leia(v[i])

fimpara

i <- 0

enquanto (i<=9) faca

ii <- i+1

enquanto (ii<=9) faca

se v[i]=v[ii] entao

v[i] <- 0

fimse

ii <- ii+1

fimenquanto

i <- i+1

fimenquanto

escreval("Os numeros tirando os repetido:")

para i de 0 ate 9 faca

escreva(v[i])

fimpara

Fimalgoritmo

12. Faça um programa que leia um vetor de 5 posições para números reais e, depois, um código inteiro. Se o código for zero, finalize o programa; se for 1, mostre o vetor na ordem direta; se for 2, mostre o vetor na ordem inversa. Caso, o código for diferente de 1 e 2 escreva uma mensagem informando que o código é inválido.

Algoritmo "Vetor"

// Disciplina : [Linguagem e Lógica de Programação]

// Professor : João Carlos

Var

v:vetor [1..5] de real

indice:inteiro

codigo:inteiro

Inicio

escreva("para ver os numeros na ordem direta digite 1 para ver na")  
 escreva(" ordem inversa digite 2.")

para indice de 1 ate 5 faca  
 escreva("Digite 5 numeros:")  
 leia(v[indice])  
 fimpara

escreva("qual e o codigo:")  
 leia(codigo)

se (codigo=0) entao  
 escreva("codigo finalizado")  
 fimse

se (codigo=1) entao  
 para indice de 1 ate 5 faca  
 escreva("o numero ",indice,"é:",v[indice] )  
 fimpara  
 fimse

se (codigo=2) entao  
 indice <-5  
 repita  
 escreva(v[indice])  
 indice <- indice-1  
 ate (indice=0)  
 fimse

Fimalgoritmo

13. Leia um vetor de 10 posições e atribua valor 0 para todos os elementos que possuírem valores negativos.

Algoritmo "semnome"

// Disciplina : [Linguagem e Lógica de Programação]

// Professor : Antonio Carlos Nicolodi

// Descrição : Aqui você descreve o que o programa faz! (função)

// Autor(a) : Nome do(a) aluno(a)



// Data atual : 20/04/2022

var

v:vetor [0..9] de real

i,ii:inteiro

Inicio

escreval("digite 10 números:")

para i de 0 ate 9 faca

leia(v[i])

fimpara

i <- 0

enquanto (i<=9) faca

ii <- i+1

enquanto (ii<=9) faca

se v[i]<=0 entao

v[i] <- 0

fimse

ii <- ii+1

fimenquanto

i <- i+1

fimenquanto

escreval("Os numeros tirando os repetido:")

para i de 0 ate 9 faca

escreva(v[i])

fimpara

Fimalgoritmo

14. Faça um programa que leia um vetor de 10 números. Leia um número  $x$ . Conte os múltiplos de um número inteiro  $x$  no vetor e mostre-os na tela.

Algoritmo "semnome"

// Disciplina : [Linguagem e Lógica de Programação]

// Professor : Antonio Carlos Nicolodi

// Descrição : Aqui você descreve o que o programa faz! (função)

// Autor(a) : Nome do(a) aluno(a)

// Data atual : 20/04/2022

var

v:vetor [0..9] de real

vet:vetor [0..9] de real

num,i:inteiro

Inicio

escreval("digite um numero:")

leia(num)

escreval("digite 10 números:")

para i de 0 ate 9 faca

leia(v[i])

fimpara

para i de 0 ate 9 faca

vet[i] <- v[i]\*num

fimpara

escreval("o resultado da multiplicação dos 10 numeros pelo outro numero é:")

para i de 0 ate 9 faca

escreval(v[i], "", i+1, "=", vet[i])

fimpara

Fimalgoritmo

15. Escreva um programa que leia números inteiros no intervalo [0,50] e os armazene em um vetor com 10 posições. Preencha um segundo vetor apenas com os números ímpares do primeiro vetor. Imprima os dois vetores, 2 elementos por linha.

Algoritmo "semnome"

// Disciplina : [Linguagem e Lógica de Programação]

// Professor : Antonio Carlos Nicolodi

// Descrição : Aqui você descreve o que o programa faz! (função)

// Autor(a) : Nome do(a) aluno(a)

// Data atual : 20/04/2022

var

v:vetor [0..9] de real

vet:vetor [0..9] de real

imp:vetor [0..9] de real

num,i:inteiro

escreval("digite 10 números entre 0 e 50:")

enquanto (i<=9) faça

leia(num)

se (num>=0) e (num<=50) entao

v[i] <- num

i <- i+1

senao

escreva("número não aceito, por favor escreva um número entre 0 e 50:")

fimse

fimenquanto

para i de 0 ate 9 faça

imp[i] <- v[i]%2

fimpara

i <- 0

enquanto (i<=9) faça

se imp[i]=1 entao

vet[i] <- v[i]

fimse

i <- i+1

fimenquanto

para i de 0 ate 9 faça

escreval(v[i],vet[i])

fimpara

Fimalgoritmo

16. Faça um programa que receba do usuário dois vetores,  $A$  e  $B$ , com 10 números inteiros cada. Crie um novo vetor denominado  $C$  calculando  $C = A - B$ . Mostre na tela os dados do vetor  $C$ .

Algoritmo "semnome"

// Disciplina : [Linguagem e Lógica de Programação]

// Professor : Antonio Carlos Nicolodi

// Descrição : Aqui você descreve o que o programa faz! (função)

// Autor(a) : Nome do(a) aluno(a)

// Data atual : 20/04/2022

var

a:vetor [0..9] de real

b:vetor [0..9] de real

c:vetor [0..9] de real

num,i:inteiro

Inicio

escreval("digite 10 números:")

para i de 0 ate 9 faca

leia(num)

a[i] <- num

fimpara

escreval("digite mais 10 números:")

para i de 0 ate 9 faca

leia(num)

b[i] <- num

fimpara

para i de 0 ate 9 faca

c[i] <- a[i]-b[i]

fimpara

escreval("a subtração dos numeros de a e b são:")

para i de 0 ate 9 faca

escreval(c[i])

fimpara

Fimalgoritmo

17. Faça um programa que leia dois vetores de 10 posições e calcule outro vetor contendo, nas posições pares os valores do primeiro e nas posições impares os valores do segundo.

Algoritmo "semnome"

// Disciplina : [Linguagem e Lógica de Programação]

// Professor : Antonio Carlos Nicolodi

// Descrição : Aqui você descreve o que o programa faz! (função)

// Autor(a) : Nome do(a) aluno(a)

// Data atual : 20/04/2022

var

a:vetor [1..10] de real

b:vetor [1..10] de real

c:vetor [1..20] de real

num,i:inteiro

Início

escreval("digite 10 números:")

para i de 1 ate 10 faca

leia(num)

a[i] <- num

fimpara

escreval("digite mais 10 números:")

para i de 1 ate 10 faca

leia(num)

b[i] <- num

fimpara

c[1] <- b[1]

c[2] <- a[1]

c[3] <- b[2]

c[4] <- a[2]

c[5] <- b[3]

c[6] <- a[3]

c[7] <- b[4]

c[8] <- a[4]

c[9] <- b[5]

c[10] <- a[5]

c[11] <- b[6]

c[12] <- a[6]

c[13] <- b[7]

c[14] <- a[7]

c[15] <- b[8]

c[16] <- a[8]

c[17] <- b[9]

c[18] <- a[9]

c[19] <- b[10]

c[20] <- a[10]

escreval("os numeros são:")

para i de 1 ate 20 faca

escreval(c[i])

fimpara

Fimalgoritmo

18. Faça um programa que leia dez conjuntos de dois valores, o primeiro representando o número do aluno e o segundo representando a sua altura em metros. Encontre o aluno mais baixo e o mais alto. Mostre o número do aluno mais baixo e do mais alto, juntamente com suas alturas.

Algoritmo "semnome"

// Disciplina : [Linguagem e Lógica de Programação]

// Professor : Antonio Carlos Nicolodi

// Descrição : Aqui você descreve o que o programa faz! (função)

// Autor(a) : Nome do(a) aluno(a)

// Data atual : 21/04/2022

Var

m:vetor[0..9,0..1] de real

maior:vetor[0..0,0..0] de real

menor:vetor[0..0,0..0] de real

i:inteiro

Inicio



escreval("digite o numero de 10 alunos:")

para i de 0 ate 9 faca

leia(m[i,0])

fimpara

escreval("digite a altura do aluno:")

para i de 0 ate 9 faca

leia(m[i,1])

se (i=0) entao

maior[0,0] <- m[i,1]

menor[0,0] <- m[i,1]

fimse

//altura mais alto//

se (m[i,1]<=maior[0,0]) entao

maior[0,0] <- m[i,1]

fimse

//altura mais baixa//

se m[i,1]>=menor[0,0] entao

menor[0,0] <- m[i,1]

fimse

fimpara

escreval("o maior aluno é:",maior[0,0])

escreval("o menor aluno é:",menor[0,0])

19. Faça um programa que preencha um vetor de tamanho 100 com os 100 primeiros naturais que não são múltiplos de 7 ou que terminam com 7.

Algoritmo "semnome"

// Disciplina : [Linguagem e Lógica de Programação]

// Professor : Antonio Carlos Nicolodi

// Descrição : Aqui você descreve o que o programa faz! (função)

// Autor(a) : Nome do(a) aluno(a)

// Data atual : 21/04/2022

Var

v:vetor[0..99] de real

i:inteiro

Inicio

para i de 0 ate 99 faca

se ((i%7)<>0) entao

se (i<>7) entao

se (i<>17)entao

se (i<>27)entao

se (i<>37)entao

se (i<>47)entao

se (i<>57)entao

se (i<>67)entao

se (i<>77)entao

se (i<>87)entao

se (i<>97)entao

v[i] <- i

fimse

fimse

fimse

fimse

fimse

fimse

fimse

fimse  
fimse  
fimse  
fimse  
fimpara

escreval("os numeros são:")  
para i de 0 ate 99 faca  
escreval(v[i])  
fimpara

Fimalgoritmo

20. Leia 10 números inteiros e armazene em um vetor. Em seguida escreva os elementos que são primos e suas respectivas posições no vetor.

Algoritmo "semnome"

// Disciplina : [Linguagem e Lógica de Programação]  
// Professor : Antonio Carlos Nicolodi  
// Descrição : Aqui você descreve o que o programa faz! (função)  
// Autor(a) : Nome do(a) aluno(a)  
// Data atual : 21/04/2022

Var

v:vetor[1..10,0..1] de inteiro

c,i,n,p,g:inteiro

indice,j:inteiro

max:inteiro

Inicio

max <- 4

g <- 1

escreval("digite 10 números:")  
para indice de 1 ate max faca  
leia(v[indice,0])  
fimpara

```
para indice de 1 ate max faca
v[indice,1] <- g
g <- g+1
fimpara
```

```
para indice de 1 ate max faca
  j<- v[indice]
  p<-0
  para i de 1 ate j faca
    se (j mod i=0) entao
      p<- p+1
    fimse
  fimpara
```

```
se (p=2) entao
  escreva(v[indice,0])
  escreva(v[indice,1])
  escreval()
fimse
fimpara
```

Fimalgoritmo

21. Faça um programa que receba 6 números inteiros e mostre:

- Os números pares digitados;
- A soma dos números pares digitados;
- Os números ímpares digitados;
- A quantidade de números ímpares digitados;

Algoritmo "semnome"

// Disciplina : [Linguagem e Lógica de Programação]

// Professor : Antonio Carlos Nicolodi

// Descrição : Aqui você descreve o que o programa faz! (função)

// Autor(a) : Nome do(a) aluno(a)

// Data atual : 21/04/2022

Var

v:vetor[0..5] de inteiro

par:vetor[0..5] de inteiro

imp:vetor[0..5] de inteiro

max,i,soma,numimp:inteiro

Inicio

max <- 5

numimp <- 0

escreval("digite 6 números:")

para i de 0 ate max faca

leia(v[i])

fimpara

para i de 0 ate max faca

se ((v[i]%2)=0) entao

par[i] <- v[i]

fimse

se ((v[i]%2)<>0) entao

numimp <- numimp+1

imp[i] <- v[i]

fimse

fimpara

para i de 0 ate max faca

soma <- soma+par[i]

fimpara

```
escreval("os numeros pares:")
```

```
para i de 0 ate 5 faca
```

```
escreval(par[i])
```

```
fimpara
```

```
escreval("os numeros impares:")
```

```
para i de 0 ate 5 faca
```

```
escreval(imp[i])
```

```
fimpara
```

```
escreval("A soma dos números pares:", soma)
```

```
escreval("A quantidade de números impares:", numimp)
```

Fimalgoritmo

22. Faça um programa que leia dois vetores de 10 elementos. Crie um vetor que seja a intersecção entre os 2 vetores anteriores, ou seja, que contém apenas os números que estão em ambos os vetores. Não deve conter números repetidos.

Algoritmo "semnome"

// Disciplina : [Linguagem e Lógica de Programação]

// Professor : Antonio Carlos Nicolodi

// Descrição : Aqui você descreve o que o programa faz! (função)

// Autor(a) : Nome do(a) aluno(a)

// Data atual : 21/04/2022

Var

max,maxcd,i,soma,roda,contador\_repet:inteiro

a:vetor[0..9] de inteiro

b:vetor[0..9] de inteiro

c:vetor[0..19] de inteiro

d:vetor[0..19] de inteiro

Inicio

max <- 9

roda<- 0

maxcd <- 19

escreval("digite 10 números:")

para i de 0 ate max faca

leia(a[i])

fimpara

escreval("digite mais 10 números:")

para i de 0 ate max faca

leia(b[i])

fimpara

//a em comparação com b, indo para c//

para roda de 0 ate max faca

para i de 0 ate max faca

se a[roda]=b[i] entao

```

        contador_repet <- contador_repet+1
    fimse
fimpara
    se (contador_repet<=1) entao
        c[roda]<- a[roda]
    fimse
    contador_repet <- 0
fimpara

```

```

//b em comparação com a, indo para c//
para roda de 0 ate max faca
    para i de 0 ate max faca
        se a[i]=b[roda] entao
            contador_repet <- contador_repet+1
        fimse
    fimpara
    se (contador_repet<=1) entao
        c[roda+max+1]<- b[roda]
    fimse
    contador_repet <- 0
fimpara

```

```

//conjunto união--reaproveitar depois, kkkk//
//contador_repet <- 0
//para roda de 0 ate maxcd faca
//    para i de 0 ate maxcd faca
//        se c[roda]=c[i] entao

```



```
//      contador_repet <- contador_repet+1
//      fimse
//      fimpara
//
//      se (contador_repet>1) entao
//      c[roda] <- 0
//      fimse
//      contador_repet <- 0
//fimpara
```

```
//intersecção dos números
```

```
// intersecção dos dois vetores
contador_repet <- 0
para roda de 0 ate maxcd faca
    para i de 0 ate maxcd faca
        se c[roda]=c[i] entao
            contador_repet <- contador_repet+1
        fimse
    fimpara

    se (contador_repet=1) entao
        c[roda] <- 0
    fimse
    contador_repet <- 0
fimpara
```

//tirar a repetição dos dois vetores ou fazer a união  
dos dois vetores

```
contador_repet <- 0
```

```
para roda de 0 ate maxcd faca
```

```
para i de 0 ate maxcd faca
```

```
se c[roda]=c[i] entao
```

```
contador_repet <- contador_repet+1
```

```
fimse
```

```
fimpara
```

```
se (contador_repet>1) entao
```

```
c[roda] <- 0
```

```
fimse
```

```
contador_repet <- 0
```

```
fimpara
```

```
escreval("A intercecção dos numeros que você escreveu foi:")
```

```
para i de 0 ate maxcd faca
```

```
escreval(c[i])
```

```
fimpara
```

Fimalgoritmo

23. Faça um programa que leia dois vetores de 10 elementos. Crie um vetor que seja a união entre os 2 vetores anteriores, ou seja, que contém os números dos dois vetores. Não deve conter números repetidos.

Algoritmo "semnome"

// Disciplina : [Linguagem e Lógica de Programação]

// Professor : Antonio Carlos Nicolodi

// Descrição : Aqui você descreve o que o programa faz! (função)

// Autor(a) : Nome do(a) aluno(a)

// Data atual : 21/04/2022

Var

max,maxcd,i,soma,roda,contador\_repet:inteiro

a:vetor[0..4] de inteiro

b:vetor[0..4] de inteiro

c:vetor[0..10] de inteiro

d:vetor[0..10] de inteiro

Inicio

max <- 2

roda<- 0

maxcd <- 10

escreval("digite 10 números:")

para i de 0 ate max faca

leia(a[i])

fimpara

escreval("digite mais 10 números:")

para i de 0 ate max faca

leia(b[i])

fimpara

//a em comparação com b, indo para c//

para roda de 0 ate max faca

para i de 0 ate max faca

```

se a[roda]=b[i] entao
    contador_repet <- contador_repet+1
fimse
fimpara
se (contador_repet<=1) entao
    c[roda]<- a[roda]
fimse
contador_repet <- 0
fimpara

```

```

//b em comparação com a, indo para c//
para roda de 0 ate max faca
    para i de 0 ate max faca
        se a[i]=b[roda] entao
            contador_repet <- contador_repet+1
        fimse
    fimpara
    se (contador_repet<=1) entao
        c[roda+max+1]<- b[roda]
    fimse
    contador_repet <- 0
fimpara

```

```

//conjunto união--reaproveitar depois, kkkk//
//contador_repet <- 0
//para roda de 0 ate maxcd faca
//    para i de 0 ate maxcd faca

```

```
//      se c[roda]=c[i] entao
//      contador_repet <- contador_repet+1
//      fimse
//      fimpara
//
//      se (contador_repet>1) entao
//      c[roda] <- 0
//      fimse
//      contador_repet <- 0
//fimpara

//intersecção dos números
```

//tirar a repetição dos dois vetores ou fazer a união dos dois vetores

```
contador_repet <- 0
```

```
para roda de 0 ate maxcd faca
```

```
    para i de 0 ate maxcd faca
```

```
        se c[roda]=c[i] entao
```

```
            contador_repet <- contador_repet+1
```

```
        fimse
```

```
    fimpara
```

```
se (contador_repet>1) entao
```

```
    c[roda] <- 0
```

```
fimse
```

contador\_repet <- 0

fimpara

escreval("A intersecção dos numeros que você escreveu foi:")

para i de 0 ate maxcd faca

escreval(c[i])

fimpara

Fimalgoritmo

24. Leia dois vetores de inteiros  $x$  e  $y$ , cada um com 5 elementos (assume que o usuário não informa elementos repetidos). Calcule e mostre os vetores resultantes em cada caso abaixo:

- Soma entre  $x$  e  $y$ : soma de cada elemento de  $x$  com o elemento da mesma posição em  $y$ .
- Produto entre  $x$  e  $y$ : multiplicação de cada elemento de  $x$  com o elemento da mesma posição em  $y$ .
- Diferença entre  $x$  e  $y$ : todos os elementos de  $x$  que não existam em  $y$ .
- Interseção entre  $x$  e  $y$ : apenas os elementos que aparecem nos dois vetores.
- União entre  $x$  e  $y$ : todos os elementos de  $x$ , e todos os elementos de  $y$  que não estão em  $x$ .

Algoritmo

"semnome"

// Disciplina : [Linguagem e Lógica de Programação]

// Professor : Antonio Carlos Nicolodi

// Descrição : Aqui você descreve o que o programa faz! (função)

// Autor(a) : Nome do(a) aluno(a)

// Data atual : 21/04/2022

Var

max,maxcd,i,roda,contador\_repet:inteiro

a:vetor[0..4] de inteiro  
b:vetor[0..4] de inteiro  
soma:vetor[0..4] de inteiro  
mult:vetor[0..4] de inteiro  
c:vetor[0..10] de inteiro  
d:vetor[0..10] de inteiro

Inicio

max <- 2

roda<- 0

maxcd <- 10

//aceita os valores//

escreval("digite 10 números:")

para i de 0 ate max faca

leia(a[i])

fimpara

escreval("digite mais 10 números:")

para i de 0 ate max faca

leia(b[i])

fimpara

para i de 0 ate max faca

soma[i] <- a[i]+b[i]

fimpara

para i de 0 ate max faca

mult[i] <- a[i]\*b[i]

```
//a em comparação com b, indo para c//
```

```
para roda de 0 ate max faca
```

```
para i de 0 ate max faca
```

```
se a[roda]=b[i] entao
```

```
contador_repet <- contador_repet+1
```

```
fimse
```

```
fimpara
```

```
se (contador_repet<=1) entao
```

```
c[roda]<- a[roda]
```

```
d[roda]<- a[roda]
```

```
fimse
```

```
contador_repet <- 0
```

```
fimpara
```

```
//b em comparação com a, indo para c//
```

```
para roda de 0 ate max faca
```

```
para i de 0 ate max faca
```

```
se a[i]=b[roda] entao
```

```
contador_repet <- contador_repet+1
```

```
fimse
```

```
fimpara
```

```
se (contador_repet<=1) entao
```

```
c[roda+max+1]<- b[roda]
```

```
d[roda+max+1]<- b[roda]
```



fimse

contador\_repet <- 0

fimpara

//tirar a repetção dos dois vetores ou fazer a união dos dois vetores

//esse irá ser usado para mostrar na tela

contador\_repet <- 0

para roda de 0 ate maxcd faca

para i de 0 ate maxcd faca

se d[roda]=d[i] entao

contador\_repet <- contador\_repet+1

fimse

fimpara

se (contador\_repet>1) entao

d[roda] <- 0

fimse

contador\_repet <- 0

fimpara

// interpecção dos dois vetores

contador\_repet <- 0

para roda de 0 ate maxcd faca

para i de 0 ate maxcd faca

se c[roda]=c[i] entao

contador\_repet <- contador\_repet+1

fimse

fimpara

se (contador\_repet=1) entao

c[roda] <- 0

fimse

contador\_repet <- 0

fimpara

//tirar a repetção dos dois vetores ou fazer a união dos dois vetores

contador\_repet <- 0

para roda de 0 ate maxcd faca

para i de 0 ate maxcd faca

se c[roda]=c[i] entao

contador\_repet <- contador\_repet+1

fimse

fimpara

se (contador\_repet>1) entao

c[roda] <- 0

fimse

contador\_repet <- 0

fimpara

escreval("A soma é:")

para i de 0 ate max faça

escreval(soma[i])

fimpara

escreval("A multiplicação é:")

para i de 0 ate max faça

escreval(mult[i])

fimpara

escreval("A união é:")

para i de 0 ate maxcd faça

escreval(d[i])

fimpara

escreval("A intersecção é:")

para i de 0 ate maxcd faça

escreval(c[i])

fimpara

Fimalgoritmo

25. Faça um programa que leia um vetor de 15 posições e o compacte, ou seja, elimine as posições com valor zero. Para isso, todos os elementos a` frente do valor zero, devem ser movidos uma posição para trás no vetor.

Algoritmo "semnome"

// Disciplina : [Linguagem e Lógica de Programação]

// Professor : Antonio Carlos Nicolodi

// Descrição : Aqui você descreve o que o programa faz! (função)

// Autor(a) : Nome do(a) aluno(a)

// Data atual : 21/04/2022

Var

v:vetor[0..15] de inteiro

i,g,max:inteiro

Inicio

max <- 14

//aceita os valores//

escreval("digite 15 números:")

para i de 0 ate max faca

leia(v[i])

fimpara

para g de 0 ate max faca

para i de 0 ate g faca

se v[i]=0 entao

v[i] <- v[i+1]

v[i+1] <- 0

fimse

fimpara

fimpara

escreval("os numeros compactador é:")

para i de 0 ate max faca

escreval(v[i])

fimpara

Fimalgoritmo

26. Faça um programa para ler 10 números DIFERENTES a serem armazenados em um vetor. Os dados deverão ser armazenados no vetor na ordem que forem sendo lidos, sendo que caso o usuário digite um número que já foi digitado anteriormente, o programa deverá pedir para ele digitar outro número. Note que cada valor digitado pelo usuário deve ser pesquisado no vetor, verificando se ele existe entre os números que já foram fornecidos. Exibir na tela o vetor final que foi digitado.

Algoritmo "semnome"

// Disciplina : [Linguagem e Lógica de Programação]

// Professor : Antonio Carlos Nicolodi

// Descrição : Aqui você descreve o que o programa faz! (função)

// Autor(a) : Nome do(a) aluno(a)

// Data atual : 21/04/2022

Var

v:vetor[0..4] de inteiro

i,g,p,max:inteiro

Inicio

max <- 4

//aceita os valores//

escreval("digite 10 números:")

para i de 0 ate max faca

leia(v[i])

para g de 0 ate i faça

se (v[i]=v[g]) e (i<>g) então

escreva("você digitou um número igual, por favor digite outro número:")

leia(v[i])

fimse

fimpara

fimpara

escreva("os números que você digitou são:")

para i de 0 ate max faça

escreva(v[i])

fimpara

Fimalgoritmo

27. Faça um programa que leia dois números  $a$  e  $b$  (positivos menores que 10.000) e:

- Crie um vetor onde cada posição é um algarismo do número. A primeira posição é o algarismo menos significativo;
- Crie um vetor que seja a soma de  $a$  e  $b$ , mas faça-o usando apenas os vetores construídos anteriormente.

Dica: some as posições correspondentes.  
a soma ultrapassar 10, subtraia 10 do  
resultado e some 1 à próxima posição.

Algoritmo "semnome"

// Disciplina : [Linguagem e Lógica de Programação]

// Professor : Antonio Carlos Nicolodi

// Descrição : Aqui você descreve o que o programa faz! (função)

// Autor(a) : Nome do(a) aluno(a)

// Data atual : 21/04/2022

Var

va:vetor[0..3] de inteiro

vb:vetor[0..3] de inteiro

i,g,max,soma:inteiro

Inicio

max <- 3

soma <- 0

//aceita os valores//

escreval("digite um número para 'A' entre 0 e 9999:")

leia(i)

escreval("digite um número para 'B' entre 0 e 9999:")

leia(g)

va[0] <- ((i/1) %10)

va[1] <- ((i/10) %10)

va[2] <- ((i/100) %10)

va[3] <- ((i/1000) %10)

vb[0] <- ((g/1) %10)

vb[1] <- ((g/10) %10)

vb[2] <- ((g/100) %10)

vb[3] <- ((g/1000) %10)

para i de 0 ate 3 faca

soma<- soma+vb[i]+va[i]

fimpara

escreval("os números do vetor 'A':")

para i de 0 ate max faca

escreva(" ",va[i])

fimpara

escreval()

escreval("os números do vetor 'B':")

para i de 0 ate max faca

escreva(" ",vb[i])

fimpara

escreval()

escreval("a soma é:", soma)

Fimalgoritmo

28. Leia um vetor com 10 números reais, ordene os elementos deste vetor, e no final escreva os elementos do vetor ordenado.

Algoritmo "semnome"

// Disciplina : [Linguagem e Lógica de Programação]



// Professor : Antonio Carlos Nicolodi

// Descrição : Aqui você descreve o que o programa faz! (função)

// Autor(a) : Nome do(a) aluno(a)

// Data atual : 21/04/2022

Var

v:vetor[0..10] de inteiro

i,g,max,alxiliar:inteiro

Inicio

max <- 4

//aceita os valores//

escreval("digite 10 números:")

para i de 1 ate max faca

leia(v[i])

para g de 1 ate i faca

se v[g]<v[g-1] entao

alxiliar <- v[g-1]

v[g-1] <- v[g]

v[g] <- alxiliar

fimse

fimpara

fimpara

escreval("os numeros compactador é:")

para i de 0 ate max faca

escreval(v[i])

fimpara

Fimalgoritmo

29. Peça ao usuário para digitar dez valores numéricos e ordene por ordem crescente esses valores, guardando-os num vetor. Ordene o valor assim que ele for digitado. Mostre ao final na tela os valores em ordem.

Algoritmo "semnome"

// Disciplina : [Linguagem e Lógica de Programação]

// Professor : Antonio Carlos Nicolodi

// Descrição : Aqui você descreve o que o programa faz! (função)

// Autor(a) : Nome do(a) aluno(a)

// Data atual : 21/04/2022

Var

v:vetor[0..10] de inteiro

i,g,max,alxiliar:inteiro

Inicio

max <- 4

//aceita os valores//

escreval("digite 10 números:")

para i de 1 ate max faca

leia(v[i])

para g de 1 ate i faca

se v[g]<v[g-1] entao

alxiliar <- v[g-1]

v[g-1] <- v[g]

v[g] <- alxiliar

fimse

fimpara

fimpara

escreval("os numeros compactador é:")

para i de 0 ate max faca

escreval(v[i])

fimpara

Fimalgoritmo

30. Escreva um programa que leia um número inteiro positivo  $n$  e em seguida imprima  $n$  linhas do chamado Triangulo de Pascal:

```
1
1 1
1 2 1
1 3 3 1
1 4 6 4 1
1 5 10 10 5 1
```

Algoritmo  
"semnome"

```
Var
v1:vetor[0..10]
de inteiro
v2:vetor[0..10]
de inteiro
v3:vetor[0..10]
de inteiro
v4:vetor[0..10]
de inteiro
v5:vetor[0..10]
de inteiro
v6:vetor[0..10]
de inteiro
```

```
v7:vetor[0..10]
de inteiro
v8:vetor[0..10]
de inteiro
v9:vetor[0..10]
de inteiro
v10:vetor[0..10]
] de inteiro
i,j:inteiro
```

Inicio

```
v1[1] <- 1
```

```
para j de 2 ate
10 faca
  para i de 1
  ate 10 faca
    v2[i] <-
v1[i]+v1[i-1]
  fimpara
fimpara
```

```
para j de 2 ate
10 faca
  para i de 1
  ate 10 faca
    v3[i] <-
v2[i]+v2[i-1]
  fimpara
fimpara
```

```
para j de 2 ate
10 faca
  para i de 1
  ate 10 faca
    v4[i] <-
v3[i]+v3[i-1]
  fimpara
fimpara
```

```
para j de 2 ate
10 faca
  para i de 1
  ate 10 faca
    v5[i] <-
v4[i]+v4[i-1]
  fimpara
fimpara
```

```
para j de 2 ate
10 faca
  para i de 1
  ate 10 faca
```

```

        v6[i] <-
        v5[i]+v5[i-1]
        fimpara
    fimpara

    para j de 2 ate
    10 faca
        para i de 1
        ate 10 faca
            v7[i] <-
            v6[i]+v6[i-1]
            fimpara
        fimpara

    para j de 2 ate
    10 faca
        para i de 1
        ate 10 faca
            v8[i] <-
            v7[i]+v7[i-1]
            fimpara
        fimpara

    para j de 2 ate
    10 faca
        para i de 1
        ate 10 faca
            v9[i] <-
            v8[i]+v8[i-1]
            fimpara
        fimpara

    para j de 2 ate
    10 faca
        para i de 1
        ate 10 faca
            v10[i] <-
            v9[i]+v9[i-1]
            fimpara
        fimpara

    escreval("digit
    e o numero de
    linhas que
    você gostaria
    do triângulo de
    Pascal:")
    leia(i)

    se (i=1) entao
        para i de 1
        ate 10 faca

    escreva(v1[i])

```

```
fimpara  
  escreval(" ")  
fimse
```

```
se (i=2) entao  
  para i de 1  
  ate 10 faca
```

```
  escreva(v1[i])
```

```
  fimpara  
  escreval(" ")
```

```
  para i de 1  
  ate 10 faca
```

```
    escreva(v2[i])
```

```
    fimpara  
    escreval(" ")
```

```
  fimse
```

```
se (i=3) entao  
  para i de 1 ate  
  10 faca
```

```
  escreva(v1[i])
```

```
  fimpara  
  escreval(" ")
```

```
  para i de 1  
  ate 10 faca
```

```
    escreva(v2[i])
```

```
    fimpara  
    escreval(" ")
```

```
  para i de 1  
  ate 10 faca
```

```
    escreva(v3[i])
```

```
    fimpara  
    escreval(" ")
```

```
  fimse
```

```
se (i=4) entao
```

para i de 1 ate  
10 faca

escreva(v1[i])

fimpara  
escreval(" ")

para i de 1  
ate 10 faca

escreva(v2[i])

fimpara  
escreval(" ")

para i de 1  
ate 10 faca

escreva(v3[i])

fimpara  
escreval(" ")

para i de 1  
ate 10 faca

escreva(v4[i])

fimpara  
escreval(" ")

fimse

se (i=5) entao  
para i de 1  
ate 10 faca

escreva(v1[i])

fimpara  
escreval(" ")

para i de 1  
ate 10 faca

escreva(v2[i])

fimpara  
escreval(" ")

para i de 1  
ate 10 faca

escreva(v3[i])

fimpara  
escreval(" ")

para i de 1  
ate 10 faca

escreva(v4[i])

fimpara  
escreval(" ")

para i de 1  
ate 10 faca

escreva(v5[i])

fimpara  
escreval(" ")

fimse

se (i=6) entao  
para i de 1 ate  
10 faca

escreva(v1[i])

fimpara  
escreval(" ")

para i de 1  
ate 10 faca

escreva(v2[i])

fimpara  
escreval(" ")

para i de 1  
ate 10 faca

escreva(v3[i])

fimpara  
escreval(" ")

para i de 1  
ate 10 faca

escreva(v4[i])



```
fimpara  
escreval(" ")
```

```
para i de 1  
ate 10 faca
```

```
escreva(v5[i])
```

```
fimpara  
escreval(" ")
```

```
para i de 1  
ate 10 faca
```

```
escreva(v6[i])
```

```
fimpara  
escreval(" ")
```

```
fimse
```

```
se (i=7) entao  
para i de 1 ate  
10 faca
```

```
escreva(v1[i])
```

```
fimpara  
escreval(" ")
```

```
para i de 1  
ate 10 faca
```

```
escreva(v2[i])
```

```
fimpara  
escreval(" ")
```

```
para i de 1  
ate 10 faca
```

```
escreva(v3[i])
```

```
fimpara  
escreval(" ")
```

```
para i de 1  
ate 10 faca
```

```
escreva(v4[i])
```

```
fimpara  
escreval(" ")
```

```
para i de 1  
ate 10 faca
```

```
escreva(v5[i])
```

```
fimpara  
escreval(" ")
```

```
para i de 1  
ate 10 faca
```

```
escreva(v6[i])
```

```
fimpara  
escreval(" ")
```

```
para i de 1  
ate 10 faca
```

```
escreva(v7[i])
```

```
fimpara  
escreval(" ")
```

```
fimse
```

```
se (i=8) entao  
para i de 1 ate  
10 faca
```

```
escreva(v1[i])
```

```
fimpara  
escreval(" ")
```

```
para i de 1  
ate 10 faca
```

```
escreva(v2[i])
```

```
fimpara  
escreval(" ")
```

```
para i de 1  
ate 10 faca
```

```
escreva(v3[i])
```

```
fimpara  
escreval(" ")
```

para i de 1  
ate 10 faca

escreva(v4[i])

fimpara  
escreval(" ")

para i de 1  
ate 10 faca

escreva(v5[i])

fimpara  
escreval(" ")

para i de 1  
ate 10 faca

escreva(v6[i])

fimpara  
escreval(" ")

para i de 1  
ate 10 faca

escreva(v7[i])

fimpara  
escreval(" ")

para i de 1  
ate 10 faca

escreva(v8[i])

fimpara  
escreval(" ")

fimse

se (i=9) entao  
para i de 1 ate  
10 faca

escreva(v1[i])

fimpara  
escreval(" ")

para i de 1  
ate 10 faca

escreva(v2[i])

fimpara  
escreval(" ")

para i de 1  
ate 10 faca

escreva(v3[i])

fimpara  
escreval(" ")

para i de 1  
ate 10 faca

escreva(v4[i])

fimpara  
escreval(" ")

para i de 1  
ate 10 faca

escreva(v5[i])

fimpara  
escreval(" ")

para i de 1  
ate 10 faca

escreva(v6[i])

fimpara  
escreval(" ")

para i de 1  
ate 10 faca

escreva(v7[i])

fimpara  
escreval(" ")

para i de 1  
ate 10 faca

escreva(v8[i])

fimpara  
escreval(" ")

para i de 1  
ate 10 faca

escreva(v9[i])

fimpara  
escreval(" ")

fimse

se (i=10)  
entao

para i de 1  
ate 10 faca

escreva(v1[i])

fimpara  
escreval(" ")

para i de 1  
ate 10 faca

escreva(v2[i])

fimpara  
escreval(" ")

para i de 1  
ate 10 faca

escreva(v3[i])

fimpara  
escreval(" ")

para i de 1  
ate 10 faca

escreva(v4[i])

fimpara  
escreval(" ")

para i de 1  
ate 10 faca

escreva(v5[i])

fimpara  
escreval(" ")

para i de 1  
ate 10 faca

escreva(v6[i])

```
fimpara  
escreval(" ")
```

```
para i de 1  
ate 10 faca
```

```
escreva(v7[i])
```

```
fimpara  
escreval(" ")
```

```
para i de 1  
ate 10 faca
```

```
escreva(v8[i])
```

```
fimpara  
escreval(" ")
```

```
para i de 1  
ate 10 faca
```

```
escreva(v9[i])
```

```
fimpara  
escreval(" ")
```

```
para i de 1  
ate 10 faca
```

```
escreva(v10[i])
```

```
fimpara  
escreval(" ")  
fimse
```

Fimalgoritmo

...