

Lista de Exercícios - Vetores

- Elaborar um programa que leia dois vetores (vet1 e vet2) de números inteiros e, em seguida, calcule e imprima:

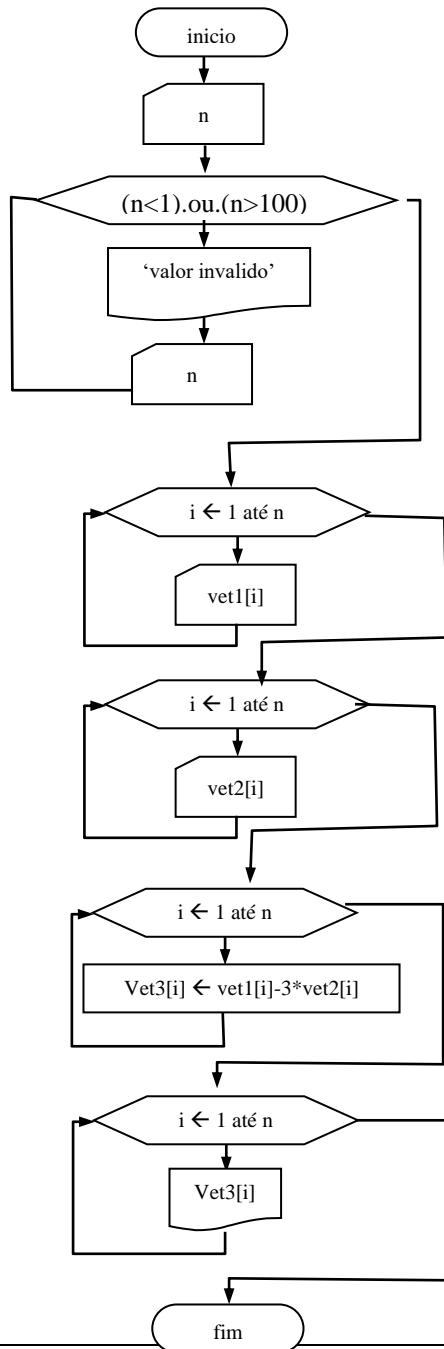
$$\text{vet3} = 2 * \text{vet1} - 3 * \text{vet2}$$

Pseudocódigo

```

Programa ex1;
Var
  N,i:inteiro;
  Vet1,vet2,vet3:arranjo de [1..100] de inteiro;
Início
  Escreva('Qual o numero de elementos dos vetores? ');
  Leia (n);
  enquanto (n<1) .ou. (n>100) faça
    escreva ('valor inválido');
    escreva('digite um novo valor');
    leia(n);
  fim_enquanto;
  para i← 1 até n faça
    leia (vet1[i]);
  fim_para;
  para i← 1 até n faça
    leia (vet2[i]);
  fim_para;
  para i←1 até n faça
    vet3[i] ← 2 * vet1[i] - 3 * vet2[i];
  fim_para;
  para i←1 até n faça
    escreva(vet3[i]);
  fim_para;
Fim.
  
```

Fluxograma



Lista de Exercícios - Vetores

2. Faça um programa que carregue dois vetores de 3 elementos literais cada um. Construa um novo vetor na qual seus valores resultam da união dos dois vetores.

Vetor1

A	B	C
1	2	3

Vetor2

D	E	F
1	2	3

VetorResultante

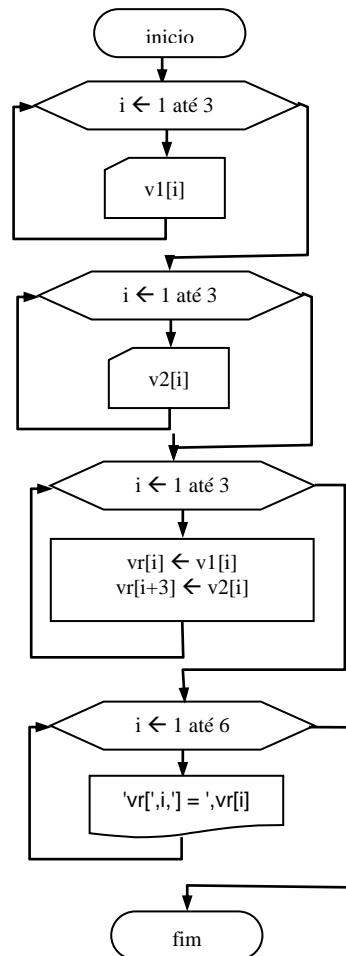
A	B	C	D	E	F
1	2	3	4	5	6

Pseudocódigo

```

programa uniao;
var
  i:inteiro;
  v1,v2:arranjo[1..3] de literal;
  vr:arranjo[1..6] de literal;
inicio
  {*** Entrada de dados ***}
  escreva ('Dados do primeiro vetor');
  para l←1 até 3 faça
    escreva ('Digite a ',i,'ª letra');
    leia(v1[i]);
  fim_para;
  escreva ('Dados do segundo vetor');
  para l←1 até 3 faça
    escreva ('Digite a ',i,'ª letra');
    leia(v2[i]);
  fim_para;
  {*** Unindo os vetores ***}
  para l←1 até 3 faça
    vr[i] := v1[i];
    vr[i+3] := v2[i];
  fim_para;
  {*** Saída de dados ***}
  para l←1 até 6 faça
    escreva('vr[' ,i,'] = ',vr[i]);
  fim.
  
```

Fluxograma



Lista de Exercícios - Vetores

3. Faça um programa que carregue dois vetores de dez elementos numéricos cada um e mostre um vetor resultante da intercalação desses dois vetores.

Vetor1

3	5	4	2	2	5	3	2	5	9
1	2	3	4	5	6	7	8	9	10

Vetor2

3	5	4	2	2	5	3	2	5	9
1	2	3	4	5	6	7	8	9	10

VetorResultante

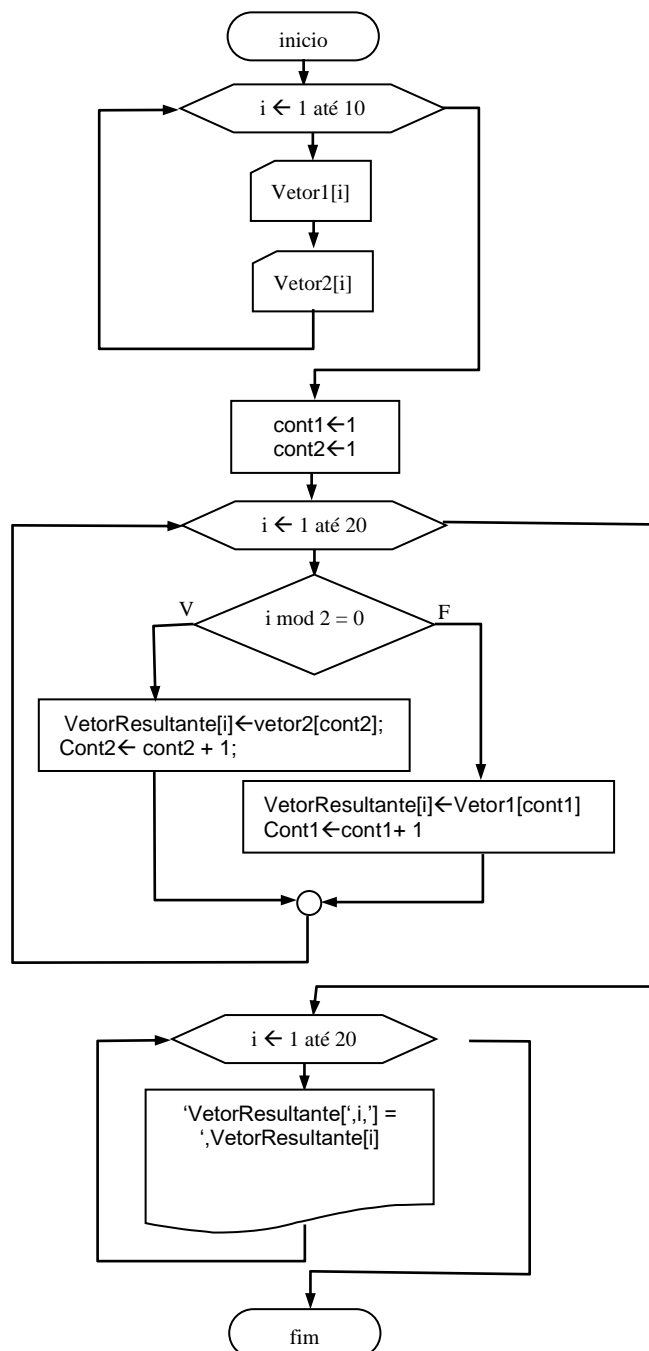
3	3	5	5	4	4	2	2	2	2	5	5	3	3	2	2	5	5	9	9
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20

Pseudocódigo

```

Programa intercalavetor;
Var
  l, cont1, cont2: inteiro;
  Vetor1: arranjo[1..10] de inteiro;
  Vetor2: arranjo[1..10] de inteiro;
  VetorResultante: arranjo[1..20] de inteiro;
Inicio
  {*** Entrada de dados no vetor ***}
  Para i ← 1 de 10 faça
    Leia(vetor1[i]);
    Leia(vetor2[i]);
  Fim_para;
  {*** Intercalando os vetores ***}
  cont1 ← 1;
  cont2 ← 1;
  Para i ← 1 de 20 faça
    Se i mod 2 = 0 então
      VetorResultante[i] ← vetor2[cont2];
      Cont2 ← cont2 + 1;
    senão
      VetorResultante[i] ← Vetor1[cont1];
      Cont1 ← cont1 + 1;
    Fim_se;
  Fim_para;
  {*** Saída de dados do vetor ***}
  Para i ← 1 de 20 faça
    Escreva("VetorResultante['i,'] = 'VetorResultante[i]");
  Fim_para;
Fim.
  
```

Fluxograma



Lista de Exercícios - Vetores

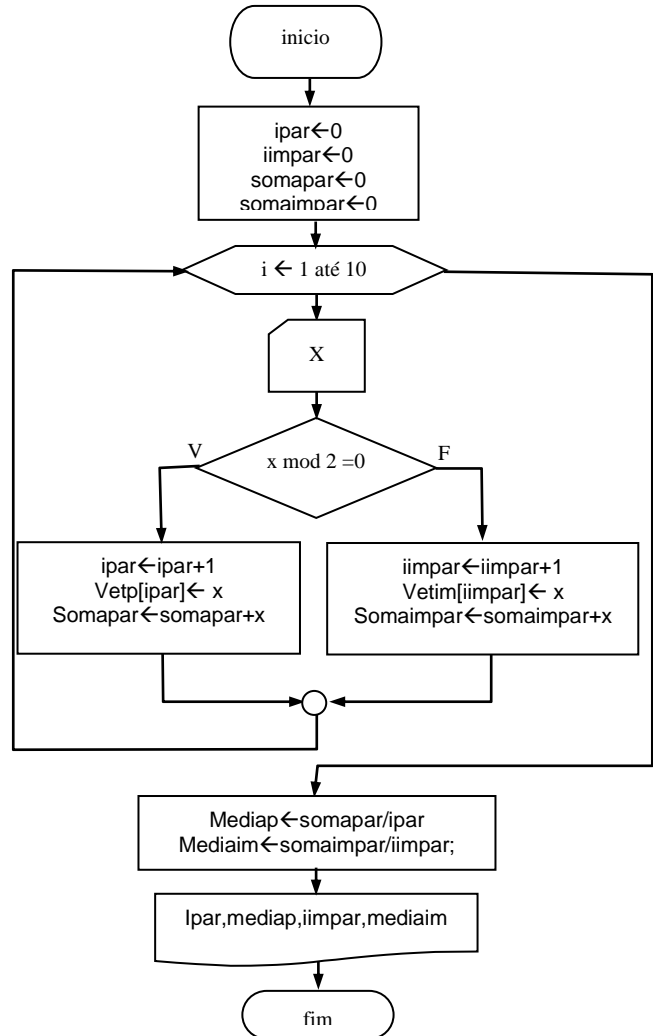
4. Leia cem números fornecidos pelo usuário e armazene os números ímpares em uma matriz A e os pares em uma matriz B; depois exiba o resultado da média e a quantidade de números em cada uma das matrizes.

Pseudocódigo

```

Programa exerc7;
Variáveis
  X , ipar, iimpar, n, i : inteiros;
  somapar, somaim: inteiros;
  Mediap, mediaim,: real;
  Vetp, vetim: arranjo [1..100] de inteiros;
Início
  ipar ← 0;
  iimpar ← 0;
  Somapar ← 0;
  Somaim ← 0;
  Para i ← 1 ate 100 faça
    Leia (X);
    Se (X mod 2 = 0) então
      Ipar ← ipar + 1;
      Vetp [ipar] ← x;
      Somapar ← somapar + x;
    Senão
      iimpar ← iimpar + 1;
      Vetim [iimpar] ← x;
      Somaim ← somaim + x;
  Fim se;
  Fim-para;
  Mediap ← somapar/ipar;
  Mediaim ← somaimpar/iimpar;
  Escreva (ipar);
  Escreva (mediap) ;
  Escreva (iimpar);
  Escreva (mediaim);
Fim.
  
```

Fluxograma



Lista de Exercícios - Vetores

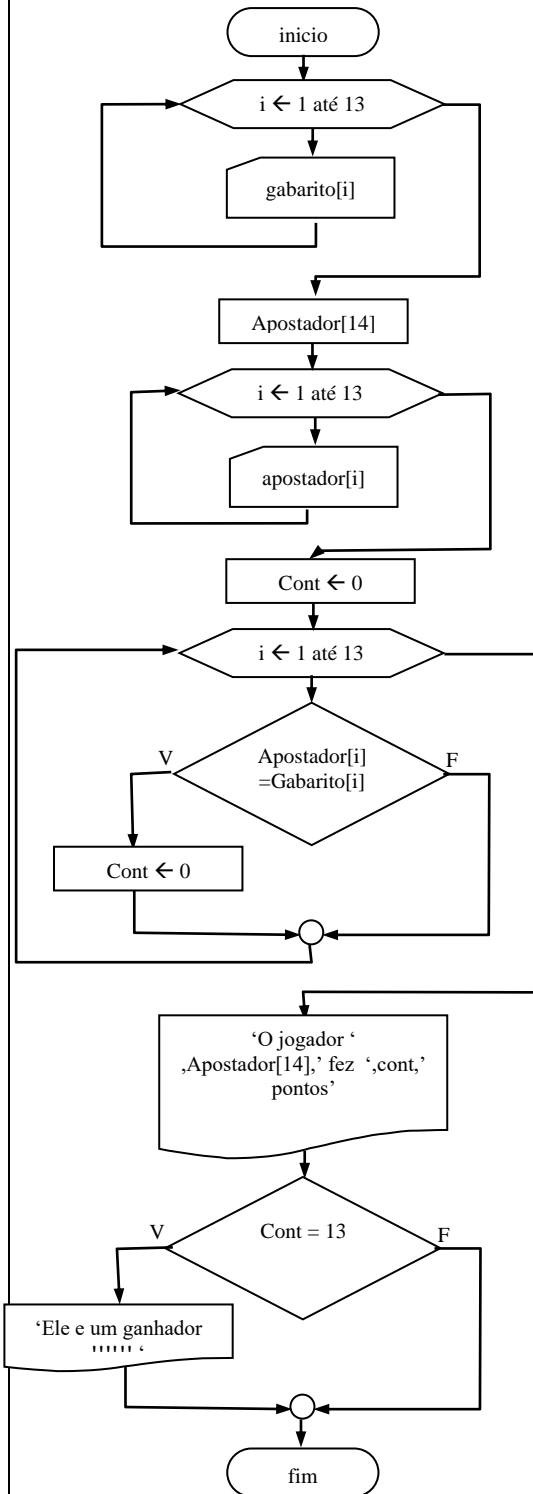
5. Escreva um algoritmo que leia um vetor de 13 elementos inteiros — o Gabarito de um teste da loteria esportiva, o qual contém os valores 1 (coluna 1), 2 (coluna 2) e 3 (coluna do meio). Leia, a seguir, para cada apostador, o número do seu cartão e um vetor de Respostas de 13 posições. Verifique para cada apostador os números de acertos, comparando o vetor de Gabarito com o vetor de Respostas. Escreva o número do apostador e o número de acertos. Se o apostador tiver 13 acertos, mostrar a mensagem 'Ganhador'.

Pseudocódigo

```

Programa ex9;
Var
  Gabarito: arranjo [1..13] de inteiros;
  Apostador: arranjo [1..14] de inteiros;
  Cont,i: inteiro;
Início
  Escreva('entrada do gabarito');
  para i ← 1 até 13 faça
    escreva ('Entre com o resultado do ',i,' jogo');
    leia (Gabarito [i]);
  fim_para;
  escreva ('Digite o numero da aposta');
  leia (Apostador[14]);
  para i ← 1 até 13 faça
    escreva ('Entre com os valores do ', (i-1), ' jogo');
    leia (Apostador [i]);
  fim_para;
  cont ← 0;
  para i ← 1 até 13 faça
    se (Apostador[i]=Gabarito[i]) então
      cont=cont+1;
    fim_se;
  fim_para;
  escreva ('O jogador ',Apostador[14],' fez ',cont,' pontos');
  se (cont=13) então
    escreva ('Ele e um ganhador !!!!! ');
  fim_se;
Fim.
  
```

Fluxograma



Lista de Exercícios - Vetores

6. Escreva um programa que leia um vetor de 100 posições e mostre-o em ordem crescente.

Pseudocódigo	Fluxograma
<pre> Programa ex10; Var A : arranjo [1..100] de reais; Aux: real; i, j: inteiro; Início para i ← 1 até 100 faça escreva ('Qual o 'i,' valor ? '); leia (A[i]); fim_para; para i ← 1 até 99 faça para j ← i+1 até 100 faça se (A[i]>A[j]) então aux=A[i]; A[i]=A[j]; A[j]=aux; fim_se; fim_para; fim_para; escreva ('Vetor em ordem crescente'); para i ← 1 até 100 faça escreva (A[i]); fim_para; Fim. </pre>	<pre> graph TD Start([Início]) --> Loop1{i ← 1 até 100} Loop1 --> ReadA[i] ReadA[i] --> Loop2{i ← 1 até 99} Loop2 --> Loop3{j ← i+1 até 100} Loop3 --> Compare{A[i] > A[j]} Compare -- V --> Swap[Aux ← A[i] A[i] ← A[j] A[j] ← aux] Compare -- F --> Join(()) Swap --> Join Join --> Loop2 Loop2 --> Loop4{i ← 1 até 100} Loop4 --> PrintA[i] PrintA[i] --> End([fim]) </pre>

7. Uma empresa deve desenvolver um programa para controlar o salário de todos os seus funcionários. As informações que ela mantém sobre cada funcionário são: Código do funcionario, nome, cargo e salário

Os dados devem ser armazenados em vetores onde cada posição contém um funcionário. Pede-se: Fazer um MENU de opções:

- 1 – Cadastrar os 20 funcionários.
- 2 – Exibir todos os funcionários no formato de tabela.
- 3 – Exibir todos os funcionários que ganham mais de R\$ 2000,00 no formato de tabela.
- 4 – Exibir o total de salários pagos a todos os funcionários.
- 5 – Exibir qual funcionário recebe o maior salário.
- 6 – Pesquisar dados dos funcionários pelo nome.
- 7 – Sair do Menu

observação:

- 1 – As opções do menu 2, 3, 4, 5 e 6 não podem ser executadas caso a 1 ainda não tenha sido executadas antes.
- 2 – Validar as opções do menu.