

Arranjos

Roberto Rocha

Exercícios de fixação

- 1 - Elaborar um programa que leia um vetor A com 15 elementos inteiros. Construir um vetor B do mesmo tipo, em que cada elemento de B deva ser o resultado do fatorial correspondente de cada elemento da matriz A. Apresentar A e B.
- 2 - Construir um programa que leia dois vetores A e B com 10 elementos quaisquer inteiros. Construir um vetor C, sendo este o resultado da união dos elementos de A e B – sem repetição. Apresentar C.
- 3 - Elaborar um programa que leia 20 elementos do tipo real em um vetor A, em seguida crie um procedimento que inverta os elementos armazenados. Ou seja, o primeiro elemento de A passará a ser o último, o segundo elemento passará a ser o penúltimo e assim por diante. Apresentar A.
- 4 - Elaborar um programa que leia 10 elementos do tipo inteiro em um vetor A. Crie um vetor ParImpar de 2 posições e armazene no índice 0 quantos elementos de A são par e no índice 1 quantos elementos de A são ímpar. Apresentar o vetor ParImpar. Obs.: não utilize o comando condicional se.
- 5 - Elaborar um programa que leia 10 elementos do tipo inteiro em um vetor A. Ordene e imprima o vetor A.

Exercícios de fixação

3 - Elaborar um programa que leia 20 elementos do tipo real em um vetor A, em seguida crie um procedimento que inverta os elementos armazenados. Ou seja, o primeiro elemento de A passará a ser o último, o segundo elemento passará a ser o penúltimo e assim por diante. Apresentar A.

Tamanho do vetor par.

Vetor A	
0	5
1	3
2	4
3	7
4	2
5	0

Vetor A	
0	0
1	2
2	7
3	4
4	3
5	5

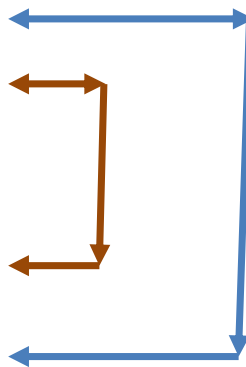
Precisamos trocar apenas até a metade -1 do vetor!

No caso ao lado $6/2 = 3$ iremos até o elemento da posição 2

Exercícios de fixação

Se o tamanho do vetor fosse ímpar!

Vetor A	
0	5
1	3
2	4
3	7
4	2



Vetor A	
0	2
1	7
2	4
3	3
4	5

Precisamos trocar apenas até a metade -1 do vetor!

No caso ao lado $5/2 = 2$ iremos até o elemento da posição 1

Exercícios de fixação

3 - Elaborar um programa que leia 20 elementos do tipo real em um vetor A, em seguida crie um procedimento que inverta os elementos armazenados. Ou seja, o primeiro elemento de A passará a ser o último, o segundo elemento passará a ser o penúltimo e assim por diante. Apresentar A.

Procedimento para
ler vetor

Procedimento para
imprimir vetor

Procedimento
inverte vetor

3 - Elaborar um programa que leia 20 elementos do tipo real em um vetor A, em seguida crie um procedimento que inverta os elementos armazenados. Ou seja, o primeiro elemento de A passará a ser o ultimo, o segundo elemento passará a ser o penúltimo e assim por diante. Apresentar A.

Procedimento para
ler vetor

```
procedimento leVetor (var v:vetor[0..N-1] de real,N:inteiro)
var
  i:inteiro
inicio
  para i de 0 ate N-1 passo 1
    faça leia(v[i])
  fimpara
fimprocedimento
```

```
28 void leVetor (float *v, int tam)
29 {
30     int i;
31     for (i=0; i<tam; i=i+1)
32     {
33         printf("Digite o %d termo do vetor:", i);
34         scanf("%f", &v[i]);
35     }
36 }
```

Em C os vetores são
sempre por
referência

3 - Elaborar um programa que leia 20 elementos do tipo real em um vetor A, em seguida crie um procedimento que inverta os elementos armazenados. Ou seja, o primeiro elemento de A passará a ser o ultimo, o segundo elemento passará a ser o penúltimo e assim por diante. Apresentar A.

Procedimento para
ler vetor

```
procedimento leVetor (var v:vetor[0..N-1] de real,N:inteiro)
var
  i:inteiro
inicio
  para i de 0 ate N-1 passo 1
    faça leia(v[i])
  fimpara
fimprocedimento
```

```
1 function [ vet ] = leVetor( tam )
2   %leVetor devolve um vetor de tamanho tam preenchido
3   for i=1:1:tam
4     vet(i)=input(sprintf('Digite o %d valor:',i));
5   end
6 end
7
```

3 - Elaborar um programa que leia 20 elementos do tipo real em um vetor A, em seguida crie um procedimento que inverta os elementos armazenados. Ou seja, o primeiro elemento de A passará a ser o ultimo, o segundo elemento passará a ser o penúltimo e assim por diante. Apresentar A.

Procedimento para imprimir vetor

```
procedimento imprimeVetor (v:vetor[0..N-1] de real,N:inteiro)
var
  i:inteiro
inicio
  para i de 0 ate N-1 passo 1
    faça escreva(v[i])
  fimpara
fimprocedimento
```

```
37 void imprimeVetor(float *v, int tam, char *nomeVetor)
38 {
39     int i;
40     for (i=0;i<tam;i=i+1)
41     {
42         printf("%s[%d]=%.2f\n", nomeVetor, i, v[i]);
43     }
44 }
```

Em C os vetores são
sempre por
referência

Mostrar o nome do vetor a ser impresso

3 - Elaborar um programa que leia 20 elementos do tipo real em um vetor A, em seguida crie um procedimento que inverta os elementos armazenados. Ou seja, o primeiro elemento de A passará a ser o ultimo, o segundo elemento passará a ser o penúltimo e assim por diante. Apresentar A.

Procedimento para
imprimir vetor

procedimento imprimeVetor (v:vetor[0..N-1] de real,N:inteiro)

var

i:inteiro

inicio

para i de 0 ate N-1 passo 1

faça escreva(v[i])

fimpara

fimprocedimento

```
1 function imprimeVetor(v,nomeVetor )
2     %imprimeVetor imprime o vetor
3     for i=1:1:size(v,2)
4         fprintf('%s(%d)=%d\n',nomeVetor,i,v(i));
5     end
6 end
7
```

3 - Elaborar um programa que leia 20 elementos do tipo real em um vetor A, em seguida crie um procedimento que inverta os elementos armazenados. Ou seja, o primeiro elemento de A passará a ser o ultimo, o segundo elemento passará a ser o penúltimo e assim por diante. Apresentar A.

Procedimento inverte vetor

Procedimento inverteVetor (var v:vetor[0..tam-1] de real,tam:inteiro)

var

i,meio:inteiro

aux:real

Inicio

meio \leftarrow tam/2

para i de 0 ate meio-1 passo 1

faca aux = v[i]

v[i]=v[tam-i-1]

v[tam-i-1]=aux

fimpara

fimprocedimento

```
45 void inverteVetor (float *v,int tam)
46 {
47     int i,meio;
48     float aux;
49     meio =tam/2;
50     for (i=0;i<meio;i=i+1)
51     {
52         aux=v[i];
53         v[i]=v[tam-i-1];
54         v[tam-i-1]=aux;
55     }
56 }
```

3 - Elaborar um programa que leia 20 elementos do tipo real em um vetor A, em seguida crie um procedimento que inverta os elementos armazenados. Ou seja, o primeiro elemento de A passará a ser o ultimo, o segundo elemento passará a ser o penúltimo e assim por diante. Apresentar A.

Procedimento inverte vetor

Procedimento inverteVetor (var v:vetor[0..tam-1] de real,tam:inteiro)

var

i,meio:inteiro

aux:real

Inicio

meio \leftarrow tam/2

para i de 0 ate meio-1 passo 1

faca aux = v[i]

v[i]=v[tam-i-1]

v[tam-i-1]=aux

fimpara

fimprocedimento

```
1  function [ v ] = inverteVetor(v)
2      %inverte as posições do vetor
3      tam=size(v,2);
4      meio=fix(tam/2);
5      for i=1:1:meio
6          aux=v(i);
7          v(i)=v(tam-i+1);
8          v(tam-i+1)=aux;
9      end
10 end
11
```

3 - Elaborar um programa que leia 20 elementos do tipo real em um vetor A, em seguida crie um procedimento que inverta os elementos armazenados. Ou seja, o primeiro elemento de A passará a ser o ultimo, o segundo elemento passará a ser o penúltimo e assim por diante. Apresentar A.

Programa principal

var

A: vetor[0..19] de real

tam: inteiro

Inicio

tam ← 20

leVetor(A,tam)

imprimeVetor(A,tam)

inverteVetor(A,tam)

imprmeVetor(A,tam)

fimalgoritmo

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <locale.h>
4  void leVetor (float *, int );
5  void imprimeVetor(float *, int,  char *);
6  void inverteVetor (float *,int );
7  /*
8   Elaborar um programa que leia 20 elementos do tipo real em um vetor A,
9   em seguida crie um procedimento que inverta os elementos armazenados.
10  Ou seja, o primeiro elemento de A passará a ser o ultimo, o segundo
11  elemento passará a ser o penúltimo e assim por diante. Apresentar A.
12  */
13
14  int main()
15  {
16      setlocale(LC_ALL, "portuguese");
17      float A[20];
18      int tam=20;
19      leVetor(A,tam);
20      printf("impressão de A conforme lido\n");
21      imprimeVetor(A,tam,"A");
22      inverteVetor(A,tam);
23      printf("impressão de A invertido\n");
24      imprimeVetor(A,tam,"A");
25      return 0;
26  }
```


3 - Elaborar um programa que leia 20 elementos do tipo real em um vetor A, em seguida crie um procedimento que inverta os elementos armazenados. Ou seja, o primeiro elemento de A passará a ser o penúltimo e assim por diante. Apresentar A.

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <locale.h>
4  void leVetor (float *, int );
5  void imprimeVetor(float *, int,  char *);
6  void inverteVetor (float *,int );
7  /*
8   Elaborar um programa que leia 20 elementos do tipo real em um v
9   em seguida crie um procedimento que inverta os elementos armaze
10  Ou seja, o primeiro elemento de A passará a ser o ultimo, o seg
11  elemento passará a ser o penúltimo e assim por diante. Apresent
12  */
13  */
14  int main()
15  {
16      setlocale(LC_ALL, "portuguese");
17      float A[20];
18      int tam=20;
19      leVetor(A,tam);
20      printf("impressão de A conforme lido\n");
21      imprimeVetor(A,tam,"A");
22      inverteVetor(A,tam);
23      printf("impressão de A invertido\n");
24      imprimeVetor(A,tam,"A");
25      return 0;
26  }
```

```
Digite o 0 termo do vetor:0
Digite o 1 termo do vetor:1
Digite o 2 termo do vetor:2
Digite o 3 termo do vetor:3
Digite o 4 termo do vetor:4
Digite o 5 termo do vetor:5
Digite o 6 termo do vetor:6
Digite o 7 termo do vetor:7
Digite o 8 termo do vetor:8
Digite o 9 termo do vetor:9
Digite o 10 termo do vetor:10
Digite o 11 termo do vetor:11
Digite o 12 termo do vetor:12
Digite o 13 termo do vetor:13
Digite o 14 termo do vetor:14
Digite o 15 termo do vetor:15
Digite o 16 termo do vetor:16
Digite o 17 termo do vetor:17
Digite o 18 termo do vetor:18
Digite o 19 termo do vetor:19
```

```
impressão de A conforme lido
A[0]=0,00
A[1]=1,00
A[2]=2,00
A[3]=3,00
A[4]=4,00
A[5]=5,00
A[6]=6,00
A[7]=7,00
A[8]=8,00
A[9]=9,00
A[10]=10,00
A[11]=11,00
A[12]=12,00
A[13]=13,00
A[14]=14,00
A[15]=15,00
A[16]=16,00
A[17]=17,00
A[18]=18,00
```

```
impressão de A invertido
A[0]=19,00
A[1]=18,00
A[2]=17,00
A[3]=16,00
A[4]=15,00
A[5]=14,00
A[6]=13,00
A[7]=12,00
A[8]=11,00
A[9]=10,00
A[10]=9,00
A[11]=8,00
A[12]=7,00
A[13]=6,00
A[14]=5,00
A[15]=4,00
A[16]=3,00
A[17]=2,00
A[18]=1,00
A[19]=0,00
```

3 - Elaborar um programa que leia 20 elementos do tipo real em um vetor A, em seguida crie um procedimento que inverta os elementos armazenados. Ou seja, o primeiro elemento de A passará a ser o último, o segundo elemento passará a ser o penúltimo e assim por diante. Apresentar A.

Programa principal

var

A: vetor[0..19] de real

tam: inteiro

Início

tam ← 20

leVetor(A,tam)

imprimeVetor(A,tam)

inverteVetor(A,tam)

imprmeVetor(A,tam)

fimalgoritmo

```

1      $laborar um programa que leia 20 elementos
2      $em seguida crie um procedimento que inver
3      $Ou seja, o primeiro elemento de A passará
4      $elemento passará a ser o penúltimo e assi
5 -    clc
6 -    clear
7 -    tam=20;
8 -    A=leVetor(tam);
9 -    fprintf('Impressao e A lido\n');
10 -   imprimeVetor(A,'A');
11 -   A=inverteVetor(A);
12 -   fprintf('Impressao e A invertido\n');
13 -   imprimeVetor(A,'A');
14

```

```

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

```

Impressao e A lido

```

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

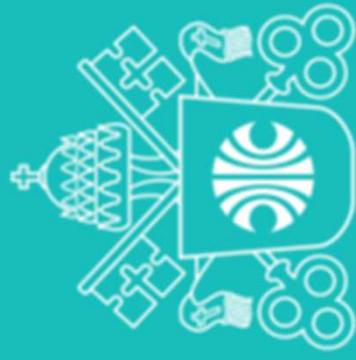
```

Impressao e A invertido

```

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

```



PUC Minas Virtual