

# 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 somatório correspondente a cada elemento do vetor 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

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.

Vetor A	
0	5
1	3
2	4
3	3
:	:
9	0

Para cada elemento verifica se já existe no vetor final

Vetor C	
0	5
1	3
2	4
3	0
4	8
5	6
6	1
:	
19	

← tamC

← Máximo

Vetor B	
0	8
1	6
2	4
3	5
:	:
9	1

# Exercícios de fixação

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.

Procedimento para  
ler vetor

Procedimento para  
imprimir vetor

Função existe  
elemento x em C?

A cada elemento a ser incluído em C devemos verificar se o valor já existe!

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.

Procedimento para  
ler vetor

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

```
27
28 void leVetor (int *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("%d",&v[i]);
35     }
36 }
```

Em C os vetores são sempre por referência

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.

Procedimento para  
imprimir vetor

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

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

Em C os vetores são  
sempre por  
referência

Mostrar o nome do vetor a ser impresso

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.

Função existe  
elemento x em C?

função existeElemento (v:vetor[0..N-1] de inteiro,tamV:inteiro,elemento:inteiro):lógico

var

i:inteiro

achei:lógico

Início

achei ← falso

i ← 0

enquanto (não achei e i<tamV)

faca se (v[i]==elemento)

então achei ← verdadeiro

fimse

i ← i+1

fimenquanto

retorne achei

fimfunção

```
61  int existeElemento (int *v,int tamV, int elemento)
62  {
63      //Verifica se o Elemento pertence ao vetor V de tamanho tamV
64      int i,achei;
65      achei=0; // 0 em C representa falso outro valor verdadeiro
66      i=0;
67      while (!achei && i<tamV)
68      {
69          if (v[i]==elemento)
70          {
71              achei=1;
72          }
73          i=i+1;
74      }
75      return achei;
76  }
```

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.

Programa principal

var

A,B: vetor[0..9] de inteiro

C:vetor [0..19] de inteiro

i,tamC: inteiro

inicio

  leVetor(A,10)

  leVetor(B,10)

  tamC ← 0

  para i de 0 até 9 passo 1

    faça se (NÃO existeElemento(C,tamC,A[i]))

      então C[tamC] ← A[i]

      tamC ← tamC+1

    fimse

  fimpara

  para i de 0 até 9 passo 1

    faça se (NÃO existeElemento(C,tamC,B[i]))

      então C[tamC] ← B[i]

      tamC ← tamC+1

    fimse

  fimpara

  imprimeVetor(C,tamC)

fimalgoritmo

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <locale.h>
4  void leVetor (int *, int );
5  void imprimeVetor(int *, int,  char *);
6  int existeElemento (int *,int, int);
7  /*
8   Construir um programa que leia dois vetores A e B com 10 elementos
9   quaisquer inteiros. Construir um vetor C, sendo este o resultado da
10  união dos elementos de A e B - sem repetição. Apresentar C.
11  */
12  int main()
13  {
14      setlocale(LC_ALL, "portuguese");
15      int A[10],B[10],C[20],i,tamC;
16      leVetor(A,10);
17      leVetor(B,10);
18      tamC=0;
19      // verifica cada elemento de A se já existe em C
20      for (i=0;i<10;i=i+1)
21      {
22          if (!existeElemento(C,tamC,A[i]))
23          {
24              C[tamC]=A[i];
25              tamC=tamC+1;
26          }
27      }
28      // verifica cada elemento de V se já existe em C
29      for (i=0;i<10;i=i+1)
30      {
31          if (!existeElemento(C,tamC,B[i]))
32          {
33              C[tamC]=B[i];
34              tamC=tamC+1;
35          }
36      }
37      printf("Impressão do vetor C\n");
38      imprimeVetor(C,tamC,"C");
39      return 0;
40  }
```



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.

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <locale.h>
4  void leVetor (int *, int );
5  void imprimeVetor(int *, int,  char *);
6  int existeElemento (int *,int, int);
7  /*
8   Construir um programa que leia dois vetores A e B com 10 elementos
9   quaisquer inteiros. Construir um vetor C, sendo este o resultado da
10  união dos elementos de A e B - sem repetição. Apresentar C.
11  */
12  int main()
13  {
14      setlocale(LC_ALL, "portuguese");
15      int A[10],B[10],C[20],i,tamC;
16      leVetor(A,10);
17      leVetor(B,10);
18      tamC=0;
19      // verifica cada elemento de A se já existe em C
20      for (i=0;i<10;i=i+1)
21      {
22          if (!existeElemento(C,tamC,A[i]))
23          {
24              C[tamC]=A[i];
25              tamC=tamC+1;
26          }
27      }
28      // verifica cada elemento de V se já existe em C
29      for (i=0;i<10;i=i+1)
30      {
31          if (!existeElemento(C,tamC,B[i]))
32          {
33              C[tamC]=B[i];
34              tamC=tamC+1;
35          }
36      }
37      printf("Impressão do vetor C\n");
38      imprimeVetor(C,tamC,"C");
39      return 0;
40  }
```

```
Digite o 0 termo do vetor:1
Digite o 1 termo do vetor:2
Digite o 2 termo do vetor:3
Digite o 3 termo do vetor:4
Digite o 4 termo do vetor:5
Digite o 5 termo do vetor:1
Digite o 6 termo do vetor:2
Digite o 7 termo do vetor:3
Digite o 8 termo do vetor:4
Digite o 9 termo do vetor:5
Digite o 0 termo do vetor:3
Digite o 1 termo do vetor:4
Digite o 2 termo do vetor:5
Digite o 3 termo do vetor:6
Digite o 4 termo do vetor:7
Digite o 5 termo do vetor:8
Digite o 6 termo do vetor:9
Digite o 7 termo do vetor:10
Digite o 8 termo do vetor:3
Digite o 9 termo do vetor:4
Impressão do vetor C
C[0]=1
C[1]=2
C[2]=3
C[3]=4
C[4]=5
C[5]=6
C[6]=7
C[7]=8
C[8]=9
C[9]=10
```



**PUC Minas**  
**Virtual**