

**INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
PERNAMBUCO**

CAMPUS AFOGADOS DA INGAZEIRA

Introdução à Programação

Vetores

Prof. Dr. Diego Rodrigues de Almeida
diego.rodrigues@afogados.ifpe.edu.br
<https://sites.google.com/site/ifpediego>

Vetores

- Imagine a seguinte situação:
- Vamos elaborar um programa em que um usuário digita 4 valores e o programa determina a sua média
- Solução:

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <locale.h>
4
5  int main(int argc, char *argv[]) {
6      setlocale(LC_ALL, "portuguese");
7      float v1, v2, v3, v4, resultado;
8      printf("Digite o valor 1: ");
9      scanf("%f", &v1);
10     printf("Digite o valor 2: ");
11     scanf("%f", &v2);
12     printf("Digite o valor 3: ");
13     scanf("%f", &v3);
14     printf("Digite o valor 4: ");
15     scanf("%f", &v4);
16     resultado = (v1 + v2 + v3 + v4)/4;
17     printf("A média dos valores é: %f", resultado);
18     return 0;
19 }
```

Vetores

- ▶ Agora imagine que ao invés de 4 números fossem 400 números
- ▶ Seriam necessárias 400 variáveis para armazenar todos os 400 valores
- ▶ O código seria imenso e difícil de se trabalhar
- ▶ Pensando nisso em C foi implementado os conjuntos de valores chamados Vetores
- ▶ Um vetor é uma coleção de variáveis de um mesmo tipo, que compartilham o mesmo nome e ocupam posições consecutivas de memória
- ▶ Em outras linguagens, vetores são também conhecidos como array

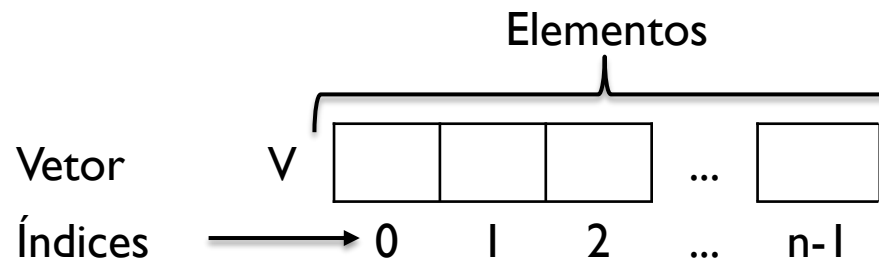
Vetores

- ▶ Cada um dos valores que o vetor armazena é denominado de elemento e é identificado por um índice
- ▶ Uma das vantagens de usar vetores é que o conjunto recebe um nome comum e elementos deste conjunto são referenciados através de índices.
- ▶ O índice começa em 0 e não em 1 como se espera
- ▶ Por exemplo:
 - ▶ Os valores 50, 11, 33, 41, 2 seriam armazenados em um vetor V da seguinte forma:

Vetor	V	50	11	33	41	2
Índices	→	0	1	2	3	4

Vetores

- De forma geral se o vetor v possui n posições, seus elementos são $v[0], v[1], v[2], \dots, v[n - 1]$



Em C os vetores são sempre indexados a partir de zero e, portanto, o último elemento de um vetor de tamanho n ocupa a posição $n - 1$

Vetores

- ▶ Forma geral de declarar um vetor:
 - ▶ `tipo nome[tamanho]`
- ▶ Exemplos:

```
int numeros[1000]; //conjunto de 1000 números inteiros
float notas[65];   //conjunto de 65 números reais
char nome[40];     //conjunto de 40 caracteres
```

IMPORTANTE

Em C não há verificação de limites em vetores. É tarefa do programador fazer com que os índices dos vetores estejam sempre dentro dos limites estabelecidos pela declaração do vetor.

Vetores

- ▶ Voltando ao problema dos 4 valores teremos a seguinte solução

```
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <locale.h>
4
5 int main(int argc, char *argv[]) {
6     setlocale(LC_ALL, "portuguese");
7     float v1, v2, v3, v4, resultado;
8     printf("Digite o valor 1: ");
9     scanf("%f", &v1);
10    printf("Digite o valor 2: ");
11    scanf("%f", &v2);
12    printf("Digite o valor 3: ");
13    scanf("%f", &v3);
14    printf("Digite o valor 4: ");
15    scanf("%f", &v4);
16    resultado = (v1 + v2 + v3 + v4)/4;
17    printf("A média dos valores é: %f", resultado);
18    return 0;
19 }
```

```
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <locale.h>
4
5 int main(int argc, char *argv[]) {
6     setlocale(LC_ALL, "Portuguese");
7     float v[4];
8     float resultado;
9     printf("Digite o valor 1: ");
10    scanf("%f", &v[0]);
11    printf("Digite o valor 2: ");
12    scanf("%f", &v[1]);
13    printf("Digite o valor 3: ");
14    scanf("%f", &v[2]);
15    printf("Digite o valor 4: ");
16    scanf("%f", &v[3]);
17    resultado = (v[0] + v[1] + v[2] + v[3])/4;
18    printf("A média dos valores é: %f", resultado);
19    return (0);
20 }
```

Atenção!!!

- ▶ Você não pode fazer coisas do tipo...

```
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <locale.h>
4
5 int main(int argc, char *argv[]) {
6     setlocale(LC_ALL, "Portuguese");
7     int v1[2], v2[2];
8     v1[0] = 1;
9     v1[1] = 2;
10    v2 = v1;
11    printf("%i, %i", v2[0], v2[1]);
12    return 0;
13 }
```

Erro de compilação!!!
Um vetor não pode receber
outro.

Atenção!!!

- ▶ Você não pode fazer coisas do tipo...

```
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <locale.h>
4
5 int main(int argc, char *argv[]) {
6     setlocale(LC_ALL, "Portuguese");
7     int v1[2], v2[2];
8     v1[0] = 1;
9     v1[1] = 2;
10    v2[0] = v1[0];
11    v2[1] = v1[1];
12    printf("%i, %i", v2[0], v2[1]);
13    return 0;
14 }
```

O correto é realizar a
passagem de valores
elemento por elemento

Exercício

- ▶ Crie um programa em que o usuário digita 5 números inteiros.
- ▶ Os números devem ser armazenados em um vetor.
- ▶ Após digitar os números, o programa exibe os números na ordem que foram digitados

Exercício

- ▶ Crie um programa em que o usuário digita 5 números inteiros.
- ▶ Os números devem ser armazenados em um vetor.
- ▶ Após digitar os números, o programa exibe os números na ordem inversa que foram digitados

Exercício

- ▶ Crie um programa em que o usuário digita 5 letras e o programa exibe as letras que foram digitadas na ordem em que foram digitadas

Resolução 1

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <locale.h>
4
5  int main(int argc, char *argv[]) {
6      setlocale(LC_ALL, "Portuguese");
7      char letras[5]
8      printf("Digite a letra 1: ");
9      fflush(stdin);
10     scanf("%c", &letras[0]);
11     printf("Digite a letra 2: ");
12     fflush(stdin);
13     scanf("%c", &letras[1]);
14     printf("Digite a letra 3: ");
15     fflush(stdin);
16     scanf("%c", &letras[2]);
17     printf("Digite a letra 4: ");
18     fflush(stdin);
19     scanf("%c", &letras[3]);
20     printf("Digite a letra 5: ");
21     fflush(stdin);
22     scanf("%c", &letras[4]);
23     printf("As letras digitadas foram: ");
24     printf("{%c, %c, %c, %c, %c}", letras[0], letras[1], letras[2], letras[3], letras[4]);
25     return (0);
26 }
```

Resolução 2

```
5 int main(int argc, char *argv[]) {
6     setlocale(LC_ALL, "Portuguese");
7     int tamanho = 5;
8     char letras[tamanho];
9     int i;
10    for(i = 0; i < tamanho; i++){
11        printf("Digite a letra %i: ", (i + 1));
12        fflush(stdin);
13        scanf("%c", &letras[i]);
14    }
15    printf("As letras digitadas foram: ");
16    for(i = 0; i < tamanho; i++){
17        printf("%c ", letras[i]);
18    }
19    return (0);
20 }
```

Exercício

- Faça o programa do exercício anterior imprimir na ordem inversa do que foi digitado

Resolução 1

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <locale.h>
4
5  int main(int argc, char *argv[]) {
6      setlocale(LC_ALL, "Portuguese");
7      char letras[5]
8      printf("Digite a letra 1: ");
9      fflush(stdin);
10     scanf("%c", &letras[0]);
11     printf("Digite a letra 2: ");
12     fflush(stdin);
13     scanf("%c", &letras[1]);
14     printf("Digite a letra 3: ");
15     fflush(stdin);
16     scanf("%c", &letras[2]);
17     printf("Digite a letra 4: ");
18     fflush(stdin);
19     scanf("%c", &letras[3]);
20     printf("Digite a letra 5: ");
21     fflush(stdin);
22     scanf("%c", &letras[4]);
23     printf("As letras digitadas foram: ");
24     printf("{%c, %c, %c, %c, %c}", letras[4], letras[3], letras[2], letras[1], letras[0]);
25     return (0);
26 }
```


Resolução 2

```
5 int main(int argc, char *argv[]) {
6     setlocale(LC_ALL, "Portuguese");
7     int tamanho = 5;
8     char letras[tamanho];
9     int i;
10    for(i = 0; i < tamanho; i++){
11        printf("Digite a letra %i: ", (i + 1));
12        fflush(stdin);
13        scanf("%c", &letras[i]);
14    }
15    printf("As letras digitadas na ordem inversa foram: ");
16    for(i = tamanho - 1; i >= 0; i--){
17        printf("%c ", letras[i]);
18    }
19    return (0);
20 }
```

Exercício

- ▶ Crie um programa em que o usuário digita 3 números em um vetor
- ▶ No final o programa informa quais foram os valores digitados

Resolução

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <locale.h>
4
5  int main(int argc, char *argv[]) {
6      setlocale(LC_ALL, "Portuguese");
7      int numeros[3];
8      int i;
9      for(i = 0; i < 3; i++){
10         printf("Digite o número %i: ", i);
11         scanf("%i", &numeros[i]);
12     }
13     printf("Os números digitados foram: %i, %i, %i\n", numeros[0], numeros[1], numeros[2]);
14     return (0);
15 }
```

Exercício

- ▶ Crie um programa em que o usuário digita 1000 números e o programa os imprime na ordem inversa

Resolução

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <locale.h>
4
5  int main(int argc, char *argv[]) {
6      setlocale(LC_ALL, "Portuguese");
7      int numeros[1000];
8      int i;
9      for(i = 0; i < 1000; i++){
10         printf("Digite o número %i: ", i + 1);
11         scanf("%i", &numeros[i]);
12     }
13     printf("Os números digitados na ordem inversa foram: {");
14     for(i = 999; i >= 0; i--){
15         printf("%i ", numeros[i]);
16     }
17     printf("}");
18     return (0);
19 }
```

Exercício

- ▶ Crie um programa que o usuário digita 1000 números e o programa exibe os números das posições par

Resolução 1

```
5 int main(int argc, char *argv[]) {
6     setlocale(LC_ALL, "Portuguese");
7     int numeros[1000];
8     int i;
9     for(i = 0; i < 1000; i++){
10         printf("Digite o número %i: ", i);
11         scanf("%i", &numeros[i]);
12     }
13     for(i = 0; i < 1000; i = i + 2){
14         printf("%i\n", numeros[i]);
15     }
16     return 0;
17 }
```

Resolução 2

```
5 int main(int argc, char *argv[]) {  
6     setlocale(LC_ALL, "Portuguese");  
7     int numeros[1000];  
8     int i;  
9     for(i = 0; i < 1000; i++){  
10         printf("Digite o número %i: ", i);  
11         scanf("%i", &numeros[i]);  
12     }  
13     for(i = 0; i < 1000; i++){  
14         if(i % 2 == 0){  
15             printf("%i\n", numeros[i]);  
16         }  
17     }  
18     return 0;  
19 }
```


Exercício

- ▶ Crie um programa em que o usuário digita o tamanho do vetor e os seus elementos
- ▶ Após isso, o programa deve imprimir os elementos em ordem invertida da ordem digitada

Resolução

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <locale.h>
4
5  int main(int argc, char *argv[]) {
6      setlocale(LC_ALL, "portuguese");
7      int tamanho;
8      printf("Digite o tamanho: ");
9      scanf("%i", &tamanho);
10     int i, vetor[tamanho];
11     for(i = 0; i < tamanho; i++){
12         printf("Digite o elemento de índice %i: ", i);
13         scanf("%i", &vetor[i]);
14     }
15     printf("O vetor em ordem decrescente é: ");
16     for(i = tamanho - 1; i >= 0; i--){
17         printf("%i ", vetor[i]);
18     }
19     return 0;
20 }
```

Exercício

- ▶ Crie um programa em que o usuário digita 5 valores a serem salvos em um vetor
- ▶ Após digitar os 5 valores o usuário digita mais um valor
- ▶ O programa deve verificar se o último valor digitado é elemento do vetor de 5 elementos

Resolução

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <locale.h>
4
5  int main(int argc, char *argv[]) {
6      setlocale(LC_ALL, "portuguese");
7      int vetor[5], numero, contador, i;
8      for(i = 0; i < 5; i++){
9          printf("Digite o elemento de índice %i: ", i);
10         scanf("%i", &vetor[i]);
11     }
12     printf("Digite o número a ser verificado: ");
13     scanf("%i", &numero);
14     contador = 0;
15     for(i = 0; i < 5; i++){
16         if(numero == vetor[i]){
17             contador++;
18         }
19     }
20     if(contador > 0){
21         printf("O número é elemento do vetor\n");
22     }else{
23         printf("O número não é elemento do vetor\n");
24     }
25     return 0;
26 }
```

Exercício

- ▶ Crie um programa em que o usuário digita um vetor de 10 valores
- ▶ O programa **NÃO** permite que o usuário digite um elemento que já existe no vetor
- ▶ Caso o usuário digite um elemento que já existe no vetor o programa informa que o número já foi digitado e não continua pedindo os valores seguintes
- ▶ O programa só pede o próximo elemento quando o usuário digitar um valor que não seja repetido

Resolução

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <locale.h>
4
5  int main(int argc, char *argv[]) {
6      setlocale(LC_ALL, "portuguese");
7      const tamanho = 10;
8      int vetor[tamanho], contador, i;
9      for(i = 0; i < tamanho; i++){
10         do{
11             printf("Digite o elemento de índice %i: ", i);
12             scanf("%i", &vetor[i]);
13             contador = 0;
14             int j;
15             for(j = 0; j < i; j++){
16                 if(vetor[i] == vetor[j]){
17                     contador++;
18                 }
19             }
20             if(contador > 0){
21                 printf("O valor digitado é repetido!!!\nTente novamente\n");
22             }
23         }while(contador > 0);
24     }
25     return 0;
26 }
```

Obrigado

