

Programação 2

Variáveis indexadas (vetores)

Como resolver o problema abaixo?

PROBLEMA:

Escreva um programa para ler a idade de cada componente de um grupo de 5 alunos. Contar e escrever quantos alunos possuem idade acima da média de idade do grupo.

[Entrada]	[Saída]
21	
10	
9	
17	
18	3

A média de idade do grupo é 15

Solução incorreta 1

```
...
main()
{
    int s,i,idade,c;
    float m;

    s=0;
    for (i=1; i<=5; i++)
    {
        scanf("%d",&idade);
        s = s +idade;
    }

    m = (float) s / 5;

    c = 0;
    for (i=1; i<=5; i++)
        if (idade > m)
            c++;
    printf("%d\n",c);
}
```

Por que está incorreta?

Solução incorreta 2

Por que está incorreta?

```
...
main()
{
    int s,i,idade,c;
    float m;

    s=0;
    c=0;
    for (i=1; i<=5; i++)
    {
        scanf("%d",&idade);
        s = s +idade;

        m = (float) s / 5;
        if (idade > m)
            c++;
    }
    printf("%d\n",c);
}
```

Solução 3

```
...
main()
{
int ct,a,b,c,d,e;
float m;

ct=0;
scanf("%d %d %d %d %d",&a,&b,&c,&d,&e);
m = (float)(a+b+c+d+e)/5;
if (a>m)
    ct++;
if (b>m)
    ct++;
if (c>m)
    ct++;
if (d>m)
    ct++;
if (e>m)
    ct++;
printf("%d\n",ct);
}
```

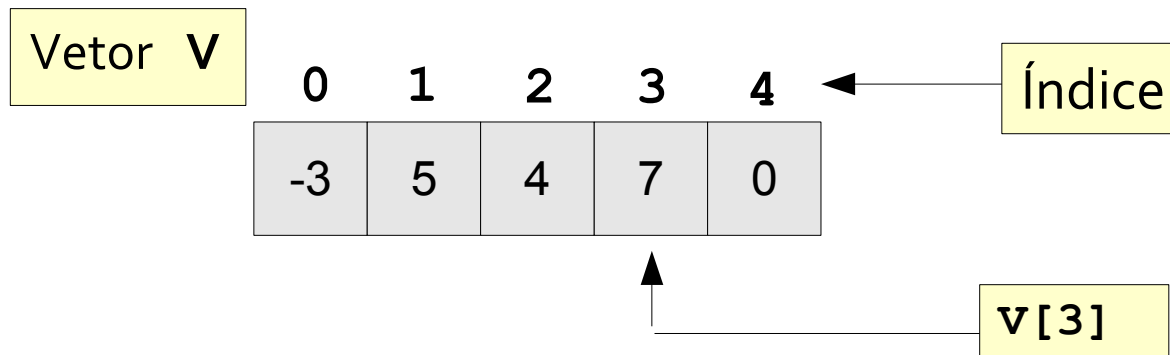
Correta, mas inadequada

Variáveis indexadas - Vetores

É um conjunto de variáveis do **mesmo tipo** que compartilham um **mesmo nome**. Índices são associados a este nome com a finalidade de permitir a individualização dos elementos do conjunto.

Quando possuí apenas um índice (uma dimensão) chamamos de **matriz unidimensional** ou **vetor**.

Ex: Um vetor V de 5 elementos



Variáveis indexadas

Um vetor deve ser declarado com a especificação da quantidade de elementos.

```
#include <stdio.h>
```

```
main()
```

```
{  
  int a, v[5];
```

Qde de
elementos

```
v[0] = -3;
```

```
v[1] = 5;
```

```
v[2] = 4;
```

```
v[3] = 7;
```

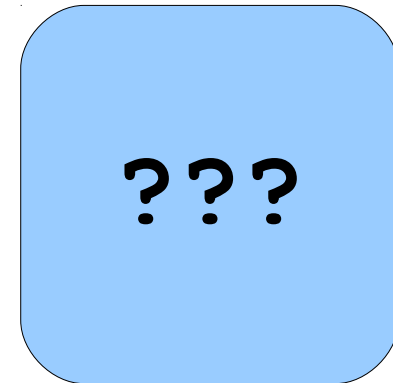
```
v[4] = 0;
```

```
printf("%d\n", v[3]);
```

```
a=1;
```

```
printf("%d\n", v[a]);
```

```
}
```



Podemos colocar uma variável como índice

Variáveis indexadas

O índice deve ser um **inteiro** não negativo.

O primeiro elemento possui índice igual a 0.

Pode-se colocar uma **constante**, **variável** ou **expressão** no lugar de um índice.

```
#include <stdio.h>
```

```
main()
```

```
{
```

```
int a,x,v[5];
```

```
v[0] = -3;
```

```
v[1] = 5;
```

```
v[2] = 4;
```

```
v[3] = 7;
```

```
v[4] = 0;
```

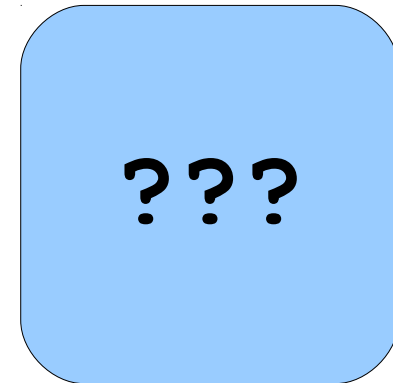
```
a = 1;
```

```
printf("%d\n",v[a+1]);
```

```
x = v[a]+1;
```

```
printf("%d\n",x);
```

```
}
```



Variáveis indexadas

O elemento de um vetor também pode ser um índice, desde que seja um **inteiro** com um valor **válido** para o índice.

```
#include <stdio.h>
```

```
main()
```

```
{
```

```
int v[5];
```

```
v[0] = -3;
```

```
v[1] = 5;
```

```
v[2] = 4;
```

```
v[3] = 7;
```

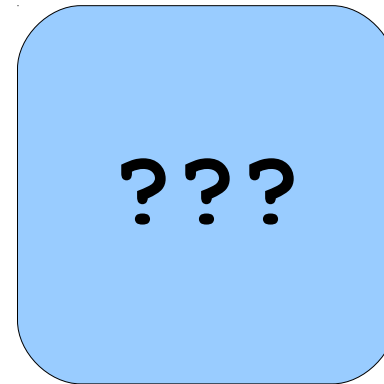
```
v[4] = 0;
```

```
printf("%d\n", v[v[4]]);
```

```
printf("%d\n", v[v[4]+1]);
```

```
printf("%d\n", v[v[v[2]]]);
```

```
}
```



Variáveis indexadas

```
#include <stdio.h>

main()
{
    int v[5], i;

    v[0] = -3;
    v[1] = 5;
    v[2] = 4;
    v[3] = 7;
    v[4] = 0;
    for (i=0; i<=4; i++)
        printf("%d\n", v[i]);
}
```



???

Leitura e escrita de um vetor

A leitura do vetor é executada de forma que cada elemento seja lido (ou escrito) individualmente.

Exemplo de leitura e escrita de um vetor de 5 elementos

```
...      /* Leitura de um vetor */
for (i=0; i<=4; i++) {
    printf("Informe o elemento %d:", i);
    scanf("%d", &v[i]);
}
...
          /* Escrita de um vetor */
for (i=0; i<=4; i++)
    printf("v[%d]=%d\n", i, v[i])
```

Observações sobre variáveis indexadas

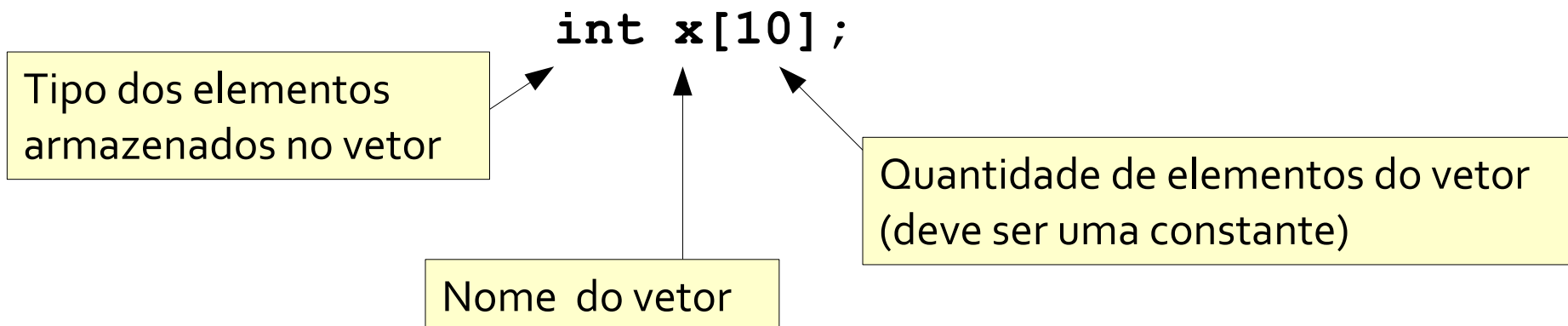
- O índice deve ser um **inteiro** (constante, variável ou expressão).
- O **primeiro** elemento do vetor possui **índice 0**.
- Qualquer operação definida para uma variável simples é válida para um componente isolado de um vetor.

```
...  
for (i=0; i<=4; i++)  
do  
{  
    printf("Informe um número positivo:");  
    scanf("%d", &num[i]);  
}  
while (num[i]<=0);  
...
```

- Não é possível operar diretamente com o conjunto completo.

Observações sobre variáveis indexadas

- Ao acessar os elementos de um vetor não utilizar índices **fora** da faixa.
- Um vetor pode ser declarado para armazenar valores de **qualquer** tipo.
Ex: `float vet[100];`
- Todos os elementos do vetor são do mesmo tipo.
- A constante definida na declaração indica a quantidade máxima de elementos de um vetor.



Resolvendo o problema proposto

```
#include <stdio.h>

main()
{
    int idade[5], i, s, c;
    float m;

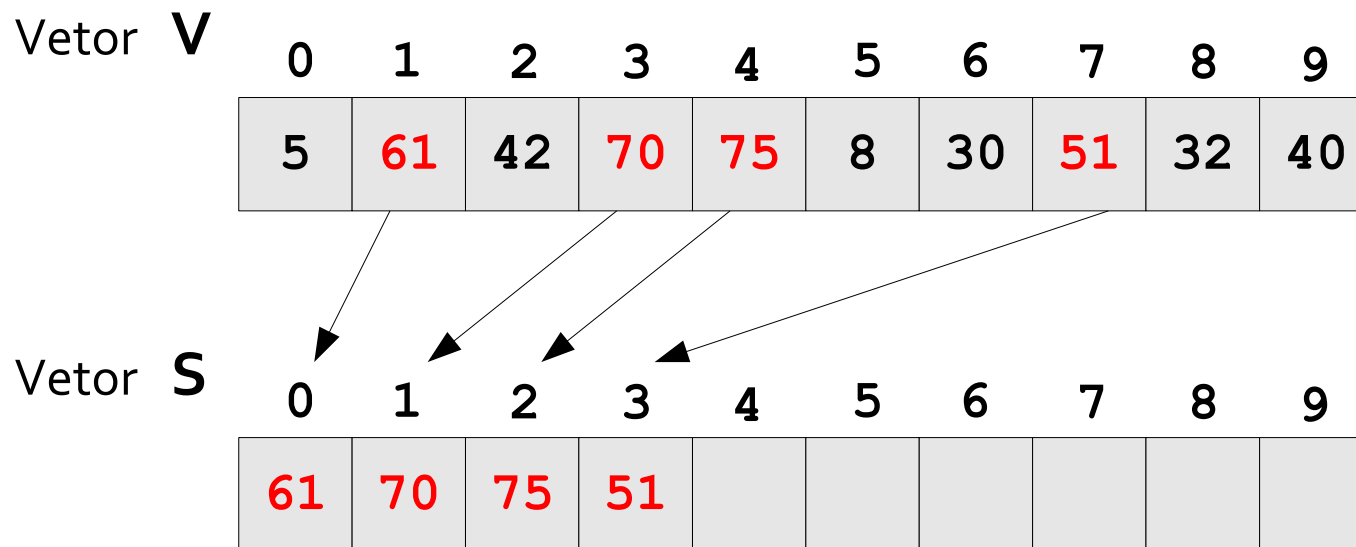
    s = 0;
    for (i=0; i<=4; i++)
    {
        printf("Informe a idade %d:", i);
        scanf("%d", &idade[i]);
        s = s + idade[i];
    }

    m = (float) s/5;
    c = 0;
    for (i=0; i<=4; i++)
        if (idade[i]>m)
            c++;
    printf("%d\n", c);
}
```

Como resolver o problema abaixo?

PROBLEMA:

Ler um vetor de 10 elementos. Copiar os valores maiores que 50 para um vetor S sem deixar espaços vagos entre os elementos de S. Após o término da cópia escrever o vetor S.



Devemos declarar S com quantos elementos?

Solução incorreta (por que?)

```
#include <stdio.h>

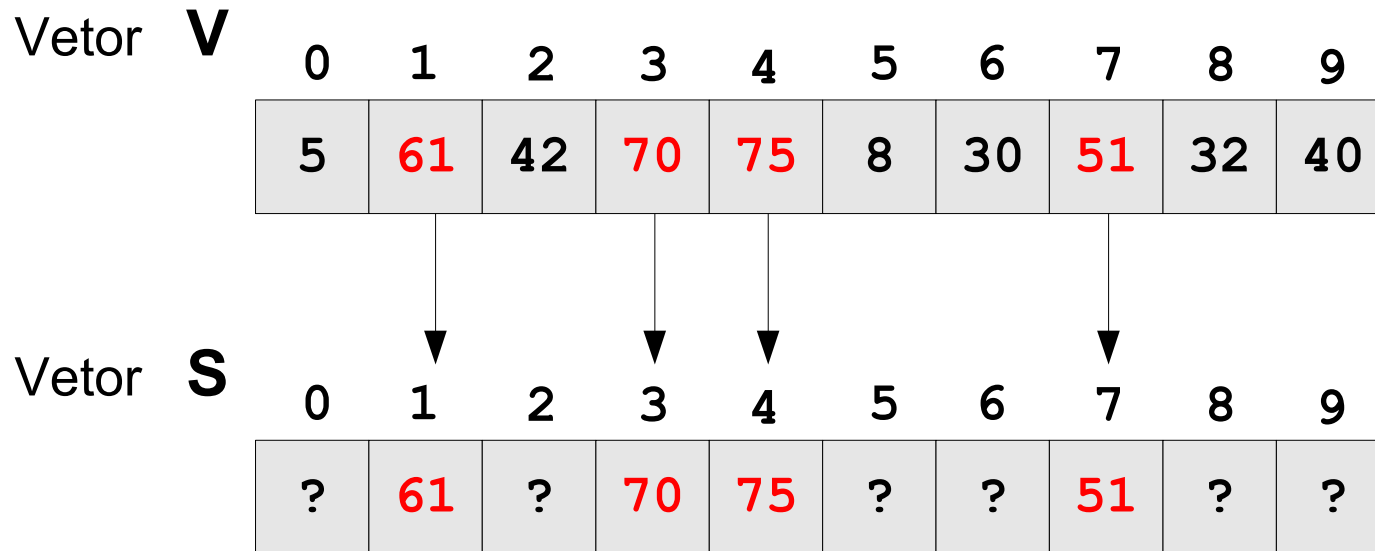
main()
{
    int v[10], i, s[10];

    for (i=0; i<=9; i++)
        scanf("%d", &v[i]);

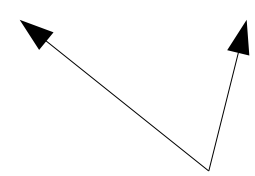
    for (i=0; i<=9; i++)
        if (v[i]>50)
            s[i] = v[i];

    for (i=0; i<=9; i++)
        printf("%d\n", s[i]);
}
```


Copia deixando espaços



```
...  
for (i=0; i<=9; i++)  
    if (v[i]>50)  
        s[i] = v[i];  
...
```



O erro está na utilização da mesma variável

E a impressão?

```
for (i=0; i<=9; i++)  
    printf("%d\n", s[i]);
```

Solução

```
#include <stdio.h>

main()
{
    int v[10], i, s[10], ns;

    for (i=0; i<=9; i++)
        scanf("%d", &v[i]);

    ns=0;
    for (i=0; i<=9; i++)
        if (v[i]>50) {
            s[ns] = v[i];
            ns++;
        }
    for (i=0; i<ns; i++)
        printf("%d\n", s[i]);
}
```