

Algoritmos de busca

Murilo Dantas

Tópicos desta aula

1. Busca sequencial (não ordenado)
2. Busca sequencial (ordenado)
3. Busca binária

Algoritmo de busca sequencial

- Roda em vetor ordenado ou não ordenado
- Vetor não ordenado
 - ▶ Busca até que o dado seja encontrado ou se chegar ao fim do vetor.
- Vetor ordenado
 - ▶ Busca até que o dado seja encontrado e enquanto for maior que o número do vetor.

Vetor não ordenado

1ª execução do laço

nº procurado

8 = 5 ⇨ Falso

0	1	2	3	4
5	3	1	8	2

2ª execução do laço

nº procurado

8 = 3 ⇨ Falso

0	1	2	3	4
5	3	1	8	2

3ª execução do laço

nº procurado

8 = 1 ⇨ Falso

0	1	2	3	4
5	3	1	8	2

4ª execução do laço

nº procurado

8 = 8 ⇨ Verdadeiro, nº encontrado

0	1	2	3	4
5	3	1	8	2

Vetor não ordenado

```
#include <iostream.h>
#include <conio.h>
void main()
{
    int X[10], n, i, achou;
    clrscr();
    // carregando os números no vetor
    // vetor com números NAO ORDENADOS
    for(i=0;i<=9;i++)
    {
        cout<<"Digite o "<<i+1<<"º número: ";
        cin>>X[i];
    }
    // digitando o número a ser buscado no vetor
    cout<<"Digite o número a ser buscado no vetor: ";
    cin>>n;
    // buscando o número digitado no vetor
    achou = 0;
    i = 0;
    while (i <= 9 && achou == 0)
    {
        if (X[i] == n)
            achou = 1;
        else
            i++;
    }
    if (achou == 0)
        cout<<"Número não encontrado no vetor";
    else
        cout<<"Número encontrado na posição "<<i+1;
    getch();
}
```

Vetor ordenado

1ª execução do laço

nº procurado

4 4 = 1 \Rightarrow Falso, 4 > 1 \Rightarrow Verdadeiro, continua.

0	1	2	3	4
1	3	5	7	9

2ª execução do laço

nº procurado

4 4 = 3 \Rightarrow Falso, 4 > 3 \Rightarrow Verdadeiro, continua.

0	1	2	3	4
1	3	5	7	9

3ª execução do laço

nº procurado

4 4 = 5 \Rightarrow Falso, 4 > 5 \Rightarrow Falso, nº Não encontrado.

0	1	2	3	4
1	3	5	7	9

Vetor ordenado

```
#include <iostream.h>
#include <conio.h>
void main()
{
    int X[10], n, i, achou;
    clrscr();
    // carregando os números no vetor
    // vetor com números ORDENADOS
    for(i=0;i<=9;i++)
    {
        cout<<"Digite o "<<i+1<<"º número: ";
        cin>>X[i];
    }
    // digitando o número a ser buscado no vetor
    cout<<"Digite o número a ser buscado no vetor: ";
    cin>>n;
    // buscando o número digitado no vetor
    achou = 0;
    i = 0;
    while (i <= 9 && achou == 0 && n >= X[i])
    {
        if (X[i] == n)
            achou = 1;
        else
            i++;
    }
    if (achou == 0)
        cout<<"Número não encontrado no vetor";
    else
        cout<<"Número encontrado na posição "<<i+1;
    getch();
}
```

Algoritmo de busca sequencial

- Pior caso
 - ▶ Número procurado é o último elemento do vetor ou não se encontra no vetor.
- Melhor caso
 - ▶ Número procurado é o primeiro elemento do vetor.

Algoritmo de busca binária

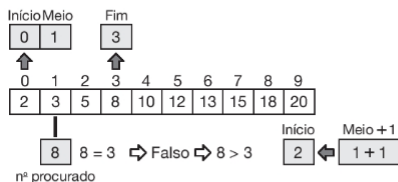
- Só executa em vetores ordenados
- Procedimento:
 - ▶ Vetor é dividido ao meio.
 - ▶ Número do meio é comparado ao procurado.
 - ▶ Se forem iguais, termina.
 - ▶ Se procurado for menor que o meio: esquerda.
 - ▶ Se procurado for maior que o meio: direita.

Algoritmo de busca binária

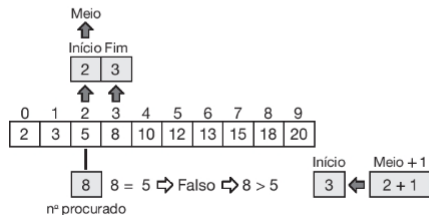
1ª execução do laço



2ª execução do laço



3ª execução do laço

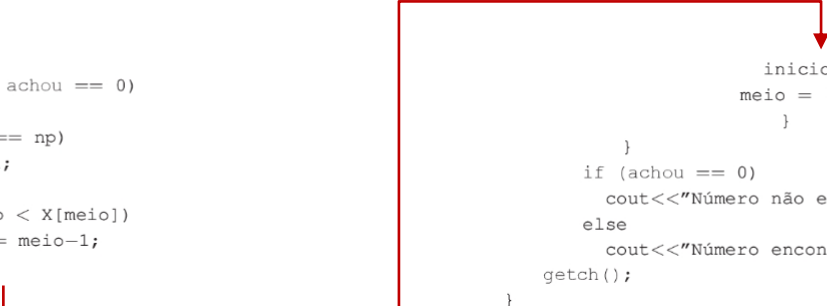


4ª execução do laço



Algoritmo de busca binária

```
#include <iostream.h>
#include <conio.h>
void main()
{
    int X[10], np, i, inicio, fim, meio, achou;
    clrscr();
    // carregando os números no vetor - ORDENADOS
    for(i=0;i<=9;i++)
    {
        cout<<"Digite o "<<i+1<<"º número: ";
        cin>>X[i];
    }
    // digitando o número a ser buscado no vetor
    cout<<"Digite o número a ser buscado no vetor: ";
    cin>>np;
    // buscando o número digitado no vetor
    achou = 0;
    inicio = 0;
    fim = 9;
    meio = (inicio+fim)/2;
    while (inicio <= fim && achou == 0)
    {
        if (X[meio] == np)
            achou = 1;
        else {
            if (np < X[meio])
                fim = meio-1;
            else
                inicio = meio+1;
                meio = (inicio+fim)/2;
        }
        if (achou == 0)
            cout<<"Número não encontrado no vetor";
        else
            cout<<"Número encontrado na posição "<<meio+1;
        getch();
    }
}
```



Perguntas?

Bibliografia da aula

- ASCENCIO, A. F. G.; ARAÚJO, G. S. Estrutura de dados. Algoritmos, análise da complexidade e implementação em Java e C/C++. Pearson. 2010.