

LAPORAN AKHIR

ALGORITMA DAN PEMROGRAMAN

LAPORAN KE-13



Disusun Oleh:

Nama: Andri Firman Saputra

NIM : 201011402125

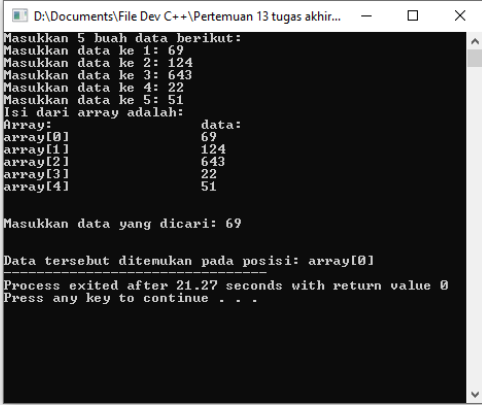
Kelas : 02TPLP023 – Pagi

TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PAMULANG

Jl. Surya Kencana No. 1 Pamulang Telp (021)7412566, Fax. (021)7412566
Tangerang Selatan - Banten

Tugas Akhir – Pertemuan 13

```
Pertemuan 13 tugas akhir.cpp
1  #include<iostream>
2  using namespace std;
3
4  int linearSearch(int data[], int dataSize, int element);
5
6  int main()
7  {
8      int dataSize = 5;
9      int data[dataSize];
10     int searchingElement = 0;
11     int flag = 0;
12     cout<<"Masukkan 5 buah data berikut: "<<endl;
13     for(int i = 0; i < dataSize; i++)
14     {
15         cout<<"Masukkan data ke "<<i+1<<": ";
16         cin>>data[i];
17     }
18
19     cout<<"Isi dari array adalah: "<<endl;
20     cout<<"Array: "<<"\t\t"<<"data: "<<endl;
21     for(int count = 0; count < dataSize; count++)
22     {
23         cout<<"array["<<count<<"]"<<"\t\t";
24         cout<<data[count]<<endl;
25     }
26
27     cout<<"\n\nMasukkan data yang dicari: ";
28     cin>>searchingElement;
29
30     flag = linearSearch(data, dataSize, searchingElement);
31
32     if(flag != -1)
33     {
34         cout<<"\n\nData tersebut ditemukan pada posisi: array["<<flag<<"]";
35     }
36     else
37     {
38         cout<<"\n\nData tersebut tidak ditemukan"<<endl;
39     }
40
41     return 0;
42 }
43
44 int linearSearch(int data[], int dataSize, int element)
45 {
46     int flag = -1;
47     for(int count = 0; count < dataSize; count++)
48     {
49         if(element == data[count])
50         {
51             flag = count;
52             break;
53         }
54     }
55     return flag;
56 }
```



The screenshot shows the output of the program in a console window. It prompts the user to enter 5 numbers. The user enters 69, 124, 643, 22, and 51. The program then displays the array contents: array[0] is 69, array[1] is 124, array[2] is 643, array[3] is 22, and array[4] is 51. Next, it prompts for a number to search for. The user enters 69. The program outputs that the data was found at position array[0]. Finally, it shows the process exited after 21.27 seconds with a return value of 0.

```
D:\Documents\File Dev C++\Pertemuan 13 tugas akhir...
Masukkan 5 buah data berikut:
Masukkan data ke 1: 69
Masukkan data ke 2: 124
Masukkan data ke 3: 643
Masukkan data ke 4: 22
Masukkan data ke 5: 51
Isi dari array adalah:
Array:          data:
array[0]        69
array[1]        124
array[2]        643
array[3]        22
array[4]        51

Masukkan data yang dicari: 69

Data tersebut ditemukan pada posisi: array[0]
Process exited after 21.27 seconds with return value 0
Press any key to continue . . .
```

Source Code:

```
#include<iostream>

using namespace std;

int linearSearch(int data[], int dataSize, int element);

int main()
{
    int dataSize = 5;
    int data[dataSize];
    int searchingElement = 0;
    int flag = 0;
    cout<<"Masukkan 5 buah data berikut: "<<endl;
    for(int i = 0; i < dataSize; i++)
    {
        cout<<"Masukkan data ke "<<i+1<<": ";
        cin>>data[i];
    }

    cout<<"Isi dari array adalah: "<<endl;
    cout<<"Array: "<<"\t\t\t"<<"data: "<<endl;
    for(int count = 0; count < dataSize; count++)
    {
        cout<<"array["<<count<<"]"<<"\t\t";
        cout<<data[count]<<endl;
    }

    cout<<"\n\nMasukkan data yang dicari: ";
    cin>>searchingElement;

    flag = linearSearch(data, dataSize, searchingElement);
```

```

        if(flag != -1)
        {
            cout<<"\n\nData tersebut ditemukan pada posisi: array["<<flag<<"]";
        }
        else
        {
            cout<<"\n\nData tersebut tidak ditemukan"<<endl;
        }

        return 0;
    }

int linearSearch(int data[], int dataSize, int element)
{
    int flag = -1;
    for(int count = 0; count < dataSize; count++)
    {
        if(element == data[count])
        {
            flag = count;
            break;
        }
    }
    return flag;
}

```

Kesimpulan:

Pada pertemuan ke 13 ini saya dapat menarik kesimpulan mengenai searching yaitu, searching adalah proses pencarian suatu elemen di dalam array, ada 2 macam pencarian yaitu pencarian sekuensial (sequential searching) dan pencarian biner (binary searching). Perbedaannya terletak pada keadaan suatu elemen atau data yang berada pada array. Pencarian sekuensial digunakan apabila data dalam keadaan acak atau tidak urut. Sedangkan pencarian biner digunakan pada data yang sudah dalam keadaan urut.

Pencarian Sekuensial

Pencarian sekuensial menggunakan prinsip sebagai berikut: data yang ada pada suatu array dibandingkan satu persatu dengan data yang dicari. Pencarian ini dilakukan dengan melakukan suatu pengulangan dari 1 sampai semua data yang ada. Pada setiap kali pengulangan, dibandingkan data yang posisinya ke-i dengan data yang dicari atau maksud. Apabila sama, maka data tersebut telah ditemukan dan proses pengulangan dihentikan. Sebaliknya, kalau sampai pengulangan selesai dan data yang dicari tidak ditemukan, maka tersebut tidak ada.

Pencarian Biner

Pencarian biner adalah sebuah teknik untuk menemukan nilai tertentu dalam sebuah larik (array), dengan menghilangkan setengah data pada setiap langkah, dipakai secara luas tetapi tidak secara eksklusif dalam ilmu komputer. Sebuah pencarian biner mencari nilai tengah (median), melakukan sebuah perbandingan untuk menentukan apakah nilai yang dicari ada sebelum atau sesudahnya, kemudian mencari setengah sisanya dengan cara yang sama. Pencarian biner digunakan pada data yang sudah dalam keadaan urut.