

PERTEMUAN VI

SEARCHING

TUJUAN PRAKTIKUM

- a) Mahasiswa dapat menjelaskan pengertian Searching dengan C++
- b) Mahasiswa dapat Searching untuk beberapa metode yang ada dengan C++
- c) Mahasiswa dapat mengimplementasikan Queue dengan C++

TEORI DASAR

a) Pendahuluan

Pencarian merupakan proses yang mendasar di dalam pemrograman. Pencarian (*Searching*) merupakan tindakan untuk mendapatkan suatu data dalam kumpulan data berdasarkan suatu kunci (*key*) atau acuan data. Dalam kehidupan sehari-hari seringkali kita berurusan dengan pencarian; misalnya menemukan nomor telepon seseorang pada buku telepon atau mencari istilah dalam kamus, dan masih banyak lagi. Pada aplikasi komputer pencarian kerap kali dilakukan. Misalnya untuk proses penghapusan data/record atau mengubah data/record tertentu didalam suatu tabel atau file.

b) Sequential Search

Sequential Search (pencarian beruntun) adalah metode pencarian yang paling mudah. Pencarian berurutan adalah proses membandingkan setiap elemen array satu per satu secara berurutan yang dimulai dari elemen pertama hingga elemen yang dicari ditemukan atau hingga elemen terakhir dari array. Pencarian beruntun dapat dilakukan terhadap elemen array yang belum terurut atau terhadap elemen array yang terurut.

Dengan kata lain sequential search akan mencari data dengan cara membandingkannya satu-persatu dengan data yang ada. Metode ini disarankan untuk digunakan pada data yang sedikit saja.

c) Binary Search

Binary search adalah metode pencarian suatu data atau elemen di dalam suatu array dengan kondisi data dalam keadaan terurut. Proses pencarian binary search hanya dapat dilakukan pada kumpulan data yang sudah diurutkan terlebih dahulu (menaik atau menurun). Prinsip dari binary search terhadap N elemen dapat dijelaskan seperti berikut :

- 1) Tentukan posisi awal = 0 dan posisi akhir = N-1.
- 2) Hitung posisi tengah = $\lfloor \text{posisi awal} + \text{posisi akhir} \rfloor / 2$
- 3) Bandingkan data yang dicari dengan elemen posisi tengah.
 - Jika sama maka catat posisi dan cetak kemudian berhenti.
 - Jika lebih besar maka akan dilakukan pencarian kembali ke bagian kanan dengan posisi awal = posisi tengah + 1 dan posisi akhir tetap kemudian ulangi mulai poin 2.
 - Jika lebih kecil maka akan dilakukan pencarian kembali ke bagian kiri dengan nilai posisi awal tetap dan nilai posisi akhir = posisi tengah - 1 kemudian ulangi mulai poin 2.

TUGAS PRAKTIKUM

a) Buatlah program Sequential Search (simpan dengan nama lat6_1.cpp)

```
#include<iostream.h>
#include<conio.h>
main ()
{
    int Nilai [ 20 ] ;
    int i, N, angka, Bilangan ;
    cout<<"Masukan Banyaknya Bilangan =";
    cin>>N;

    //Membaca elemen array
    for(i=0; i<N; i++)
    {
        cout<<"Masukan elemen ke-"<<i<<" = ";
        cin>>Nilai [ i ] ;
    }
}
```

```

//Mencetak elemen array
cout<<"\n\nDeretan Bilangan =";
for(i=0; i<N; i++)
    cout<<Nilai [ i ]<<" ";
cout<<"\n\nMasukan Bilangan yang akan dicari = ";
cin>>Bilangan ;

//melakukan pencarian
i=0;
do
{
    if(Nilai [ i ]==Bilangan)
        angka = Nilai [i];
    i++;
}
while (i<N);
if (angka==Bilangan)
    cout<<"Bilangan "<<Bilangan <<" ditemukan ";
else
    cout<<"Bilangan "<<Bilangan<<" tidak ditemukan ";
getch ();
}

```

b) Buatlah program Sequential Search (simpan dengan nama lat6_2.cpp)

```

#include<iostream.h>
#include<conio.h>
main()
{
    int Nilai [20];
    int Posisi [ 20 ];
    int i, N, Bilangan, Banyak= 0;
    bool Ketemu;
    cout<<"Masukan Banyaknya Bilangan = ";
    cin>>N;

    //Membaca elemen Array
    for(i=0; i<N; i++)
    {
        cout<<"Masukan elemen ke-"<<i<<" = ";
        cin>>Nilai [ i ];
    }
}

```

```

}

//Mencetak Elemen Array
cout<<"\n\nDeretan Bilangan =";
for (i=0; i<N; i++)
    cout<<Nilai [ i ] <<" ";
cout<<"\n\nMasukan Bilangan yang akan dicari = ";
cin>> Bilangan;

//Melakukan Pencarian
for (i=0; i<N; i++)
{
    if (Nilai [ i ] ==Bilangan)
    {
        Ketemu = true;
        Posisi [Banyak] = i;
        Banyak++;
    }
}
if (Ketemu)
{
    cout<<"Bilangan "<<Bilangan<<" ditemukan Sebanyak "<<Banyak;
    cout<<"\npada posisi ke =";
    for(i=0; i<Banyak; i++)
        cout<<Posisi [ i ]<<" ";
}
else
{
    cout<<"Bilangan "<<Bilangan<<" tidak ditemukan";
}
getch ();
}

```

- c) **Buatlah program untuk pencarian dengan metode binary search (simpan dengan nama lat6_3.cpp)**

```

#include<stdio.h>
#include<iostream.h>
#include<conio.h>
#include<iomanip.h>
main ( )
{

```

```

//deklarasi variabel
int Nilai [ 20 ];
int i, j, N;
int temp, awal, akhir, tengah, Bilangan ;
//proses penginputan data
cout<<"Banyak bilangan : ";
cin>>N;
for (i=0; i<N; i++)
{
    cout<<"Elemen ke-"<<i<<"= ";
    cin>>Nilai [ i ] ;
}
cout<<"\nElemen Sebelumnya diurut = ";
for (i=0; i<N; i++ )
    cout<<setw ( 3 )<<Nilai[i] ;

//proses pengurutan data
for (i=0; i<N; i++ )
{
    for (j=i+1; j<N; j++ )
    {
        if (Nilai [ i ] > Nilai [ j ] )
        {
            temp = Nilai [ i ];
            Nilai [ i ] = Nilai [ j ];
            Nilai [ j ] = temp;
        }
    }
}
cout<<"\nElemen Setelah diurut = ";
for ( i=0; i<N; i++ )
    cout<<setw ( 3 )<< Nilai [ i ];
cout<<"\nindeks Elemen          = " ;
for ( i=0; i<N; i++ ) cout<<setw ( 3 )<<i;
cout<<"\nMasukan data yang akan anda cari : " ;
cin>>Bilangan;

//proses pencarian data
awal = 0 ;
akhir = N-1;
do
{

```

```

    tengah =( akhir + awal ) / 2;
    if ( Bilangan < Nilai [ tengah ] )
        akhir = tengah - 1;
    else
        awal = tengah + 1 ;
    }
    while (( akhir >= awal) && ( Nilai [ tengah ] != Bilangan )) ;
    if ( Nilai [ tengah ] == Bilangan )
    {
        cout<<"\nData "<<Bilangan<<" ada dalam array ";
        cout<<" pada posisi "<<tengah;
    }
    else
        cout<<"\nData "<<Bilangan<<" tidak ada dalam array\n";
    getch () ;
}

```

- d) **Buatlah program dengan Array of Struct seperti pada Pertemuan III :** program biodata pegawai yang terdiri dari field-field seperti NIP, Nama, Alamat, Agama dan Jabatan. **Kemudian tambahkan program menggunakan metode sequential search untuk mencari seorang pegawai lalu tampilkan biodatanya. (simpan dengan nama lat6_4.cpp)**

TUGAS PENDAHULUAN

1. Apa yang dimaksud dengan Searching!
2. Jelaskan perbedaan dari Sequential Search, Binary Search, dan Interpolation Search sebagai metode-metode searching !
3. Jelaskan apa saja yang mempengaruhi kecepatan proses pencarian data di dalam penyimpanan data !
4. Buatlah contoh program sederhana menggunakan Searching!

TUGAS AKHIR

1. Buatlah program untuk melakukan pencarian terhadap data 12,15 dan 37 dari sederetan data 34, 8, 50, 74, 87, 90, 12, 25, 20, 30, 35, 45, 40, 22, 29, 72, 60, 55, 53, 12, 32, 33, 12, 41, 12 ! Jika data yang dicari terdapat lebih dari satu, tentukan banyaknya dan sebutkan berada pada posisi berapa saja data yang dicari berada pada sedereatan data!