

PERTEMUAN XIII

QUEUE

TUJUAN PRAKTIKUM

- a) Mahasiswa dapat menjelaskan pengertian dan pembuatan Queue dengan C++.
- b) Mahasiswa dapat melakukan operasi penyisipan dan penghapusan elemen dalam queue dengan C++.
- c) Mahasiswa dapat mengimplementasikan Queue dengan C++.

TEORI DASAR

a) Pendahuluan

Queue atau Antrian merupakan kumpulan elemen dengan penyisipan dan penghapusan elemen yang dilakukan dari sisi/gerbang yang berbeda. Penyisipan dilakukan dari gerbang belakang dan penghapusan dilakukan dari gerbang depan. Hal ini menunjukkan bahwa untuk Queue mempunyai dua gerbang yaitu gerbang depan dan gerbang belakang. Dengan demikian dapat dilihat bahwa Queue mempunyai sifat FIFO (*first In First Out*), yaitu elemen yang pertama masuk akan keluar pertama juga. Queue dapat direpresentasikan dengan menggunakan Array atau Linked List.

b) Operasi-operasi pada Queue

Sama halnya dengan stack, operasi yang dapat dilakukan pada suatu Queue pada dasarnya adalah penyisipan elemen dan penghapusan elemen. Disamping itu juga dapat dilakukan operasi untuk mengecek apakah Queue kosong atau penuh, macam-macam operasi untuk mengecek Queue antara lain :

- 1) Operasi Inisialisasi
- 2) Operasi Queue Kosong
- 3) Operasi Queue Penuh
- 4) Operasi Mengosongkan Queue
- 5) Operasi Penyisipan Elemen Queue
- 6) Operasi Penghapusan Elemen Queue
- 7) Operasi Pencetakan Isi Queue

TUGAS PRAKTIKUM

- a) Buatlah program untuk operasi Queue dengan menggunakan Singly Linked List (simpan dengan nama lat13_1.cpp)

```
#include <iostream.h>
#include <conio.h>
#include <stdlib.h>
typedef struct node *simpul ;
struct node
{
    char Isi ;
    simpul Next ;
};
//=====
//==Prototype Function=====
//=====
void Sisip_Belakang (simpul &L, char elemen ) ;
void Hapus_Depan (simpul &L);
void Cetak (simpul L);
//=====
//==Function Main=====
//=====
main()
{
    char huruf ;
    simpul L = NULL; //Pastikan bahwa L kosong
    int i;
    cout<<"==OPERASI PADA SINGLE LINKED LIST=="<<endl<<endl;
    //=====
    //==Sisip Belakang=====
    //=====
    cout<<"\nPenyisipan Simpul \n\n";
    for(i=1; i<=3;i++)
    {
        cout<<"Masukan Huruf :";
        cin>>huruf;
        Sisip_Belakang (L, huruf);
    }
    Cetak (L) ;
    //=====
```

```

//==Hapus simpul Depan====
//=====
cout<<"\nSetelah Hapus Simpul "<<endl;
Hapus_Depan (L) ;
Cetak(L) ;
cout<<"\nSetelah Hapus Simpul "<<endl ;
Hapus_Depan ( L ) ;
Cetak (L) ;
cout<<"\nSetelah Hapus Simpul "<<endl;
Hapus_Depan (L) ;
Cetak(L) ;
cout<<"\nPenyisipan simpul \n\n";
for (i=1 ; i<=3;i++)
{
    cout<<"Masukan Huruf :";
    cin>>huruf;
    Sisip_Belakang (L, huruf);
}
Cetak (L) ;
cout<<"\nSetelah Hapus Simpul "<<endl;
Hapus_Depan ( L ) ;
Cetak (L) ;
cout<<"\nSetelah Hapus Simpul "<<endl;
Hapus_Depan (L) ;
Cetak (L) ;
getch ( ) ;
}
//*****
//**FUNCTION SISIP SIMPUL DI BELAKANG****
//*****
void Sisip_Belakang (simpul &L, char elemen)
{
    simpul bantu, baru;
    baru = (simpul) malloc (sizeof(simpul));
    baru->Isi = elemen;
    baru -> Next = NULL;
    if (L == NULL)
        L= baru;
    else
    {
        bantu =L;

```

```

        while (bantu->Next != NULL)
            bantu =bantu->Next;
        bantu->Next=baru;
    }
}
//*****
//**FUNCTION MENCETAK ISI LINKED LIST**
//*****
void Cetak (simpul L)
{
    simpul bantu;
    if(L==NULL)
        cout<<"Linked List Kosong.....\n";
    else
    {
        bantu=L;
        cout<<"\nIsi Linked List : ";
        while (bantu->Next != NULL)
        {
            cout<<bantu->Isi<<"-";
            bantu=bantu->Next;
        }
        cout<<bantu->Isi;
    }
}
//*****
//**FUNCTION HAPUS SIMPUL DEPAN**
//*****
void Hapus_Depan (simpul &L)
{
    simpul Hapus;
    if(L==NULL)
        cout<<"Linked List Kosong.....\n";
    else
    {
        Hapus=L;
        L = L->Next;
        Hapus->Next = NULL;
        free(Hapus);
    }
}
//=====eof=====

```

TUGAS PENDAHULUAN

1. Apa yang dimaksud dengan Queue!
2. Tuliskan Deklarasi sintaks Queue!
3. Sebutkan dan Jelaskan operasi-operasi pada Queue!
4. Jelaskan Aplikasi-Aplikasi Queue dalam dunia nyata!

TUGAS AKHIR

1. Buatlah program **lat13_1.cpp** di atas, dengan menggunakan sistem menu!