

# PERTEMUAN XI

## STACK

### TUJUAN PRAKTIKUM

- Mahasiswa dapat menjelaskan pengertian dan pembuatan Stack dengan C++.
- Mahasiswa dapat melakukan operasi penyisipan dan penghapusan elemen dalam Stack dengan C++.
- Mahasiswa dapat mengimplementasikan Stack dengan C++.

### TEORI DASAR

#### a) Pendahuluan

Stack atau tumpukan adalah kumpulan elemen yang hanya dapat ditambah atau dihapus dari satu ujung (gerbang) yang sama. Hal ini menunjukkan bahwa seolah-olah suatu elemen diletakkan di atas elemen yang lain. Yang memberi gambaran bahwa Stack mempunyai sifat LIFO (Last In First Out) yang berarti bahwa elemen yang terakhir masuk akan pertama keluar.

Secara sederhana stack dimisalkan kita mempunyai 4 buah kotak (A, B, C, dan D) yang ditumpukan. Kotak A diletakkan paling bawah lalu diikuti kotak B, C dan yang teratas atau terakhir adalah D. Maka untuk mengambil tiap kotak harus dilakukan berurutan dari kotak D, C, B kemudian A. Karena jika kita mengambil kotak B tanpa terlebih dahulu mengambil kotak di atasnya maka tumpukan akan roboh.

#### b) Deklarasi Stack

Bentuk deklarasi pertama:

```
#define MaxSn
TypeData    Isi[MaxS]
TypeData    Top;
```

Bentuk deklarasi kedua :

```
#define MaxSn
Struct Stack
[
    TypeData Isi [MaxS];
    TypeData    Top;
];
```

#### c) Operasi Pada Stack

Ada dua operasi dasar yang dapat dilakukan terhadap sebuah Stack, yaitu operasi insert atau penyisipan elemen yang sering disebut istilah PUSH dan operasi Delete atau Penghapusan elemen yang sering disebut istilah POP.

#### d) Inisialisasi Stack

Sebelum stack dapat dioperasikan, terlebih dahulu diinisialisasikan dengan memberi harga  $S.Top = 0$ .

```
void INITS (Stack &S)
{
    S.Top = 0;
}
```

**e) Mencetak Stack**

Isi suatu Stack dapat dicetak dengan menggunakan fungsi berikut.

```
void CETAK (Stack &S)
{
    int i;
    cout<<endl<<"Isi Stack : ";
    if (S.Top !=0)
    {
        for(i=1;i<=S.Top;i++)
            cout<<S.Isi[ i ];
    }
    else
        cout<<"Stack Kosong....";
}
```

**f) Karakteristik Stack**

Karakteristik dari suatu stack meliputi : Elemen Stack, Top, Max, Stack Kosong, dan stack penuh.

**g) Aplikasi Stack**

- 1) Simulasi stack dalam dunia nyata
- 2) Pemanggilan fungsi/procedure
- 3) Rekursif
- 4) Penanganan interupsi
- 5) Evaluasi ekspresi
- 6) Konversi notasi infiks ke notasi postfiks
- 7) Konversi bilangan basis 10 (decimal) ke basis 2 (biner)

## TUGAS PENDAHULUAN

1. Apa yang dimaksud dengan Stack!
2. Bagaimana tahapan-tahapan proses operasi PUSH!
3. Bagaimana tahapan-tahapan proses operasi POP!
4. Jelaskan karakteristik-karakteristik dari Stack!

## JAWABAN

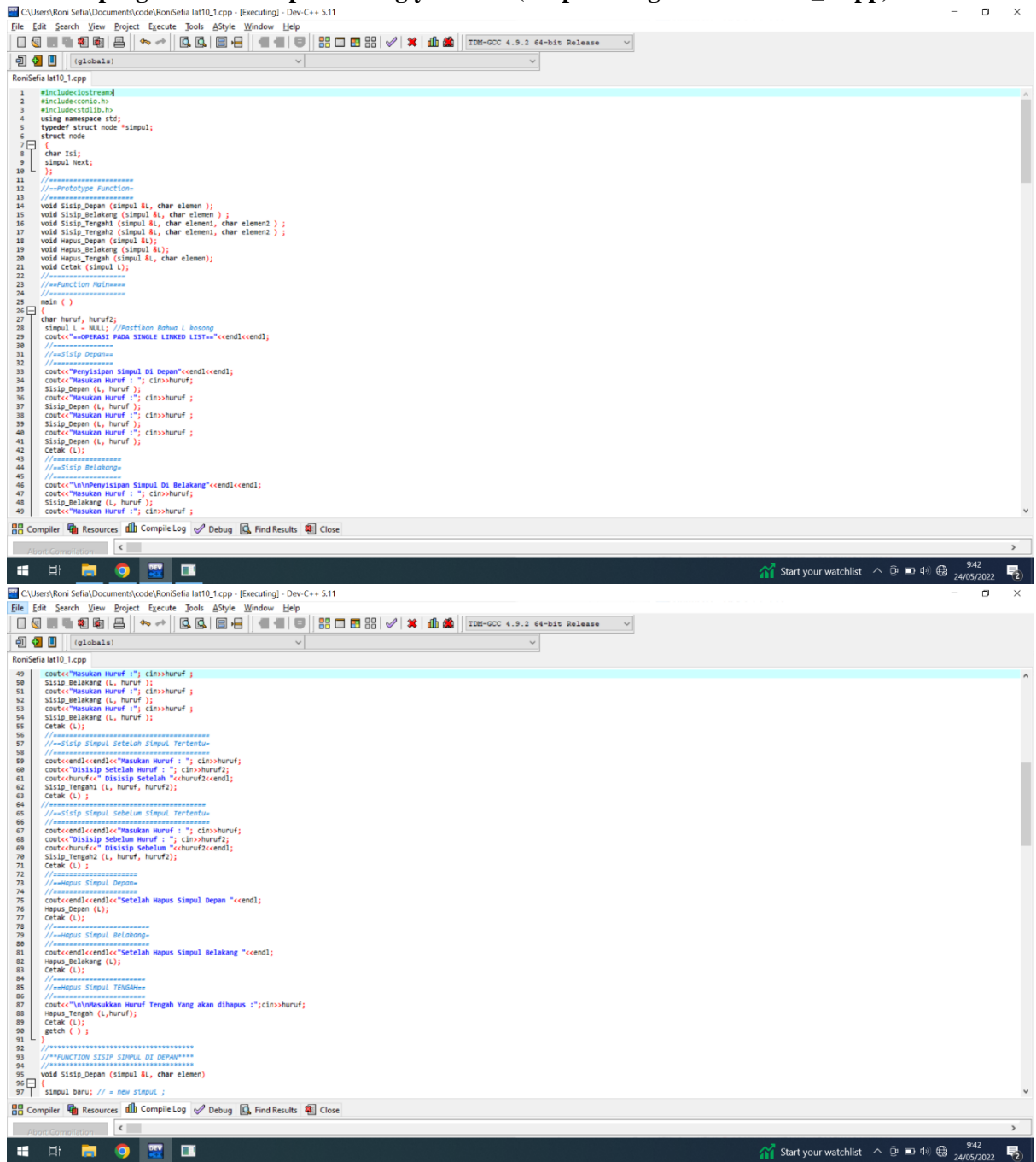
1. Stack atau tumpukan adalah kumpulan elemen yang hanya dapat di tambahatau dihapus dari satu ujung (gerbang) yang sama.
2. Tahapan-tahapan proses operasi PUSH:
  - Periksa apakah stack penuh (isfull), jika bernilai false/0 (tidak penuh) maka proses push dilaksanakan dan jika pemeriksaan ini bernilai true/1, maka proses push digagalkan.
  - Proses push-nya sendiri adalah dengan menambahkan field top dengan 1, kemudian elemen pada posisi top di isi dengan elemen data baru.
3. Tahapan-tahapan proses operasi POP:

Operasi ini biasanya dibuat dalam bentuk function yang me-return-kan nilai sesuai data yang ada di top.

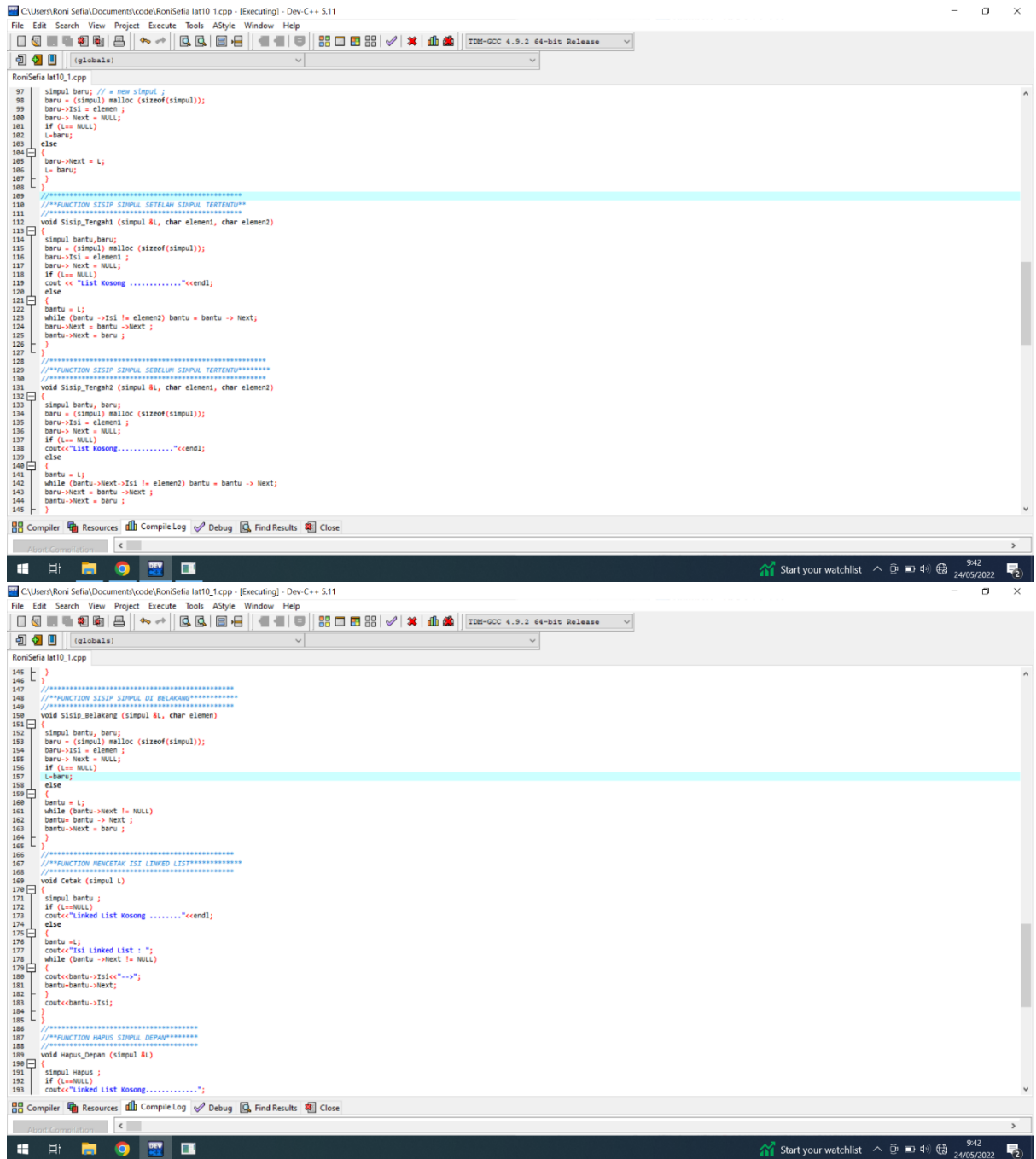
Operasi pop pada stack yang menggunakan array adalah terlebih dahulu memeriksa apakah stack sedang keadaan kosong, jika tidak kosong maka data diambil pada posisi yang ditunjuk oleh posisi top, kemudian disimpan dalam variabel baru dengan nama "data".
4. Karakteristik-karakteristik penting dari stack sebagai berikut :
  - Elemen stack yaitu item-item data di elemen stack.
  - Top (elemen puncak dari stack).
  - Jumlah elemen pada stack.
  - Status/kondisi stack.
  - Kondisi stack yang menjadi perhatian adalah :
    - Penuh.
    - Kosong.

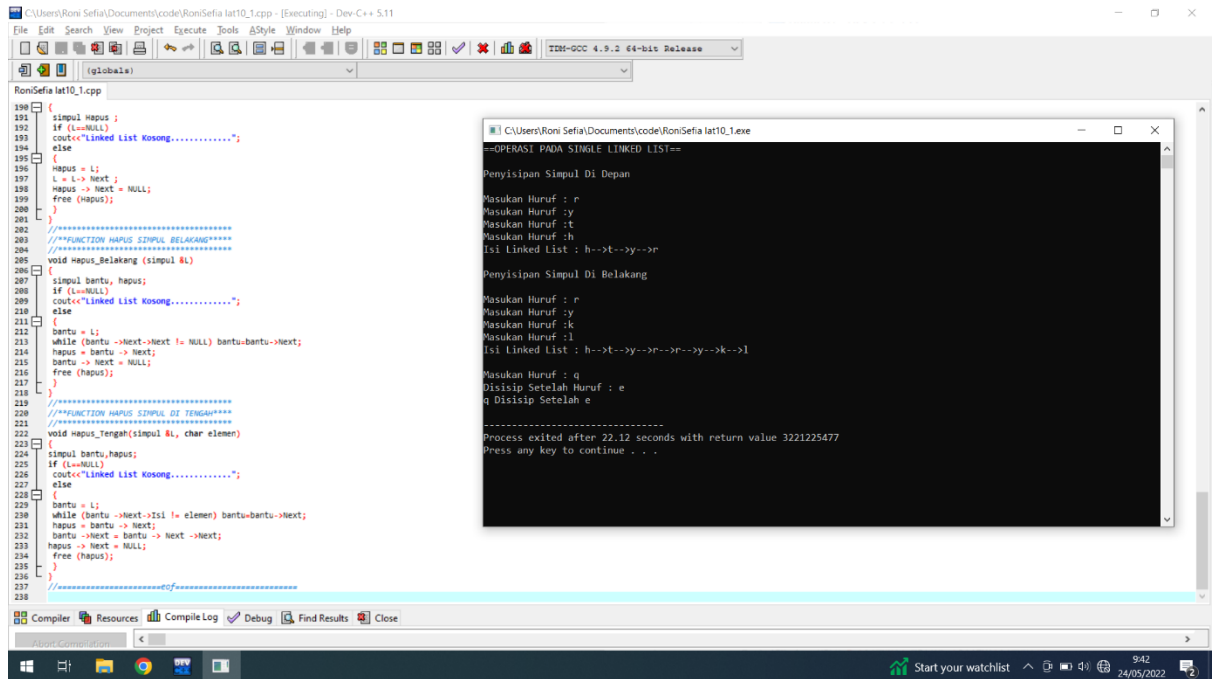
## TUGAS PRAKTIKUM 10

### a) Buatlah program untuk Operasi Singly Linked (simpan dengan nama lat10\_1.cpp)



```
1 #include<iostream>
2 #include<conio.h>
3 #include<stdlib.h>
4 using namespace std;
5 typedef struct node *simplu;
6 struct node
7 {
8     char isi;
9     simplu next;
10 };
11 //=====
12 //=====
13 //=====
14 void Sisip_Depan (simplu &L, char elemen );
15 void Sisip_Belakang (simplu &L, char elemen );
16 void Sisip_Tengah (simplu &L, char elemen1, char elemen2 );
17 void Sisip_Tengah (simplu &L, char elemen1, char elemen2 );
18 void Hapus_Depan (simplu &L);
19 void Hapus_Belakang (simplu &L);
20 void Hapus_Tengah (simplu &L, char elemen);
21 void Cetak (simplu L);
22 //=====
23 //=====
24 //=====
25 main ()
26 {
27     char huruf, huruf2;
28     simplu L = NULL; //Pontrik Bahwa i kosong
29     cout<<"=====OPERASI PADA SINGLY LINKED LIST====="<<endl;
30     //=====
31     //=====
32     //=====
33     cout<<"Penyisipan Simpul di Depan"<<endl;
34     cout<<"Masukan Huruf : "; cin>>huruf;
35     Sisip_Depan (L, huruf );
36     cout<<"Masukan Huruf : "; cin>>huruf ;
37     Sisip_Depan (L, huruf );
38     cout<<"Masukan Huruf : "; cin>>huruf ;
39     Sisip_Depan (L, huruf );
40     cout<<"Masukan Huruf : "; cin>>huruf ;
41     Sisip_Depan (L, huruf );
42     Cetak (L);
43     //=====
44     //=====
45     //=====
46     cout<<"\n\nPenyisipan Simpul Di Belakang"<<endl;
47     cout<<"Masukan Huruf : "; cin>>huruf;
48     Sisip_Belakang (L, huruf );
49     cout<<"Masukan Huruf : "; cin>>huruf ;
50     //=====
51     //=====
52     //=====
53     cout<<"Masukan Huruf : "; cin>>huruf ;
54     Sisip_Belakang (L, huruf );
55     Cetak (L);
56     //=====
57     //=====
58     //=====
59     cout<<endl<<endl<<"Masukan Huruf : "; cin>>huruf;
60     cout<<"Disisip Setelah Huruf : "; cin>>huruf2;
61     cout<<huruf<<" Disisip Setelah "<<huruf2<<endl;
62     Sisip_Tengah (L, huruf, huruf2);
63     Cetak (L );
64     //=====
65     //=====
66     //=====
67     cout<<endl<<endl<<"Masukan Huruf : "; cin>>huruf;
68     cout<<"Disisip Sebelum Huruf : "; cin>>huruf2;
69     cout<<huruf<<" Disisip Sebelum "<<huruf2<<endl;
70     Sisip_Tengah (L, huruf, huruf2);
71     Cetak (L );
72     //=====
73     //=====
74     //=====
75     cout<<endl<<endl<<"Setelah Hapus Simpul Depan "<<endl;
76     Hapus_Depan (L);
77     Cetak (L);
78     //=====
79     //=====
80     //=====
81     cout<<endl<<endl<<"Setelah Hapus Simpul Belakang "<<endl;
82     Hapus_Belakang (L);
83     Cetak (L);
84     //=====
85     //=====
86     //=====
87     cout<<"\n\nMasukkan Huruf Tengah Yang akan dihapus :";cin>>huruf;
88     Hapus_Tengah (L,huruf);
89     Cetak (L);
90     getch ( ) ;
91 }
92 //=====
93 //=====
94 //=====
95 void Sisip_Depan (simplu &L, char elemen)
96 {
97     simplu baru; // = new simplu ;
```





**b) Buatlah program untuk Operasi Doubly Linked (simpan dengan nama lat10\_2.cpp)**

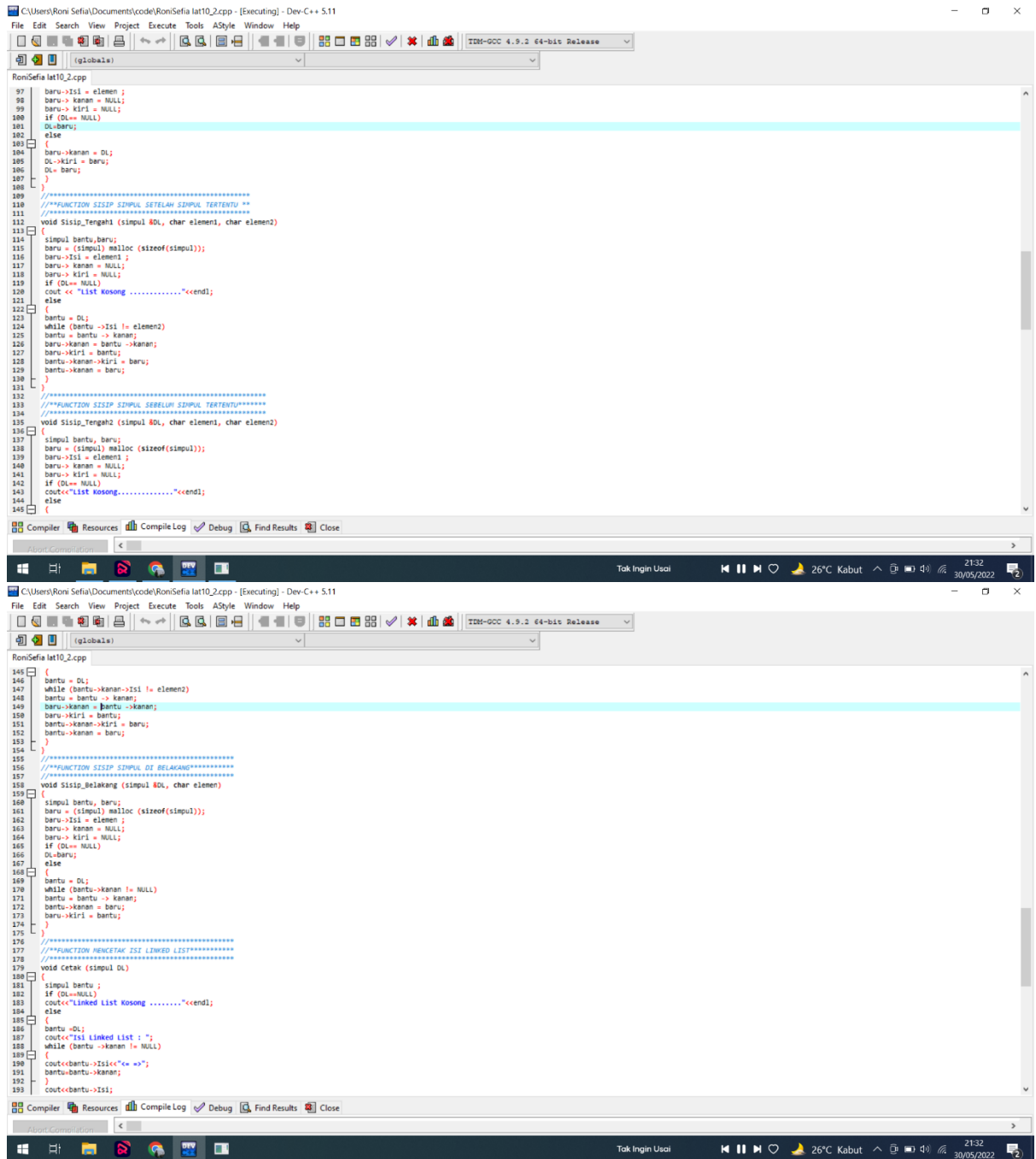
The image shows two screenshots of a C++ program for Doubly Linked List operations, running in Dev-C++ 5.11. The program is named `lat10_2.cpp` and is located at `C:\Users\Roni Seifa\Documents\code\Roni Seifa lat10_2.cpp`.

**Top Screenshot:** Shows the initial part of the program, including headers, struct definition, and function prototypes.

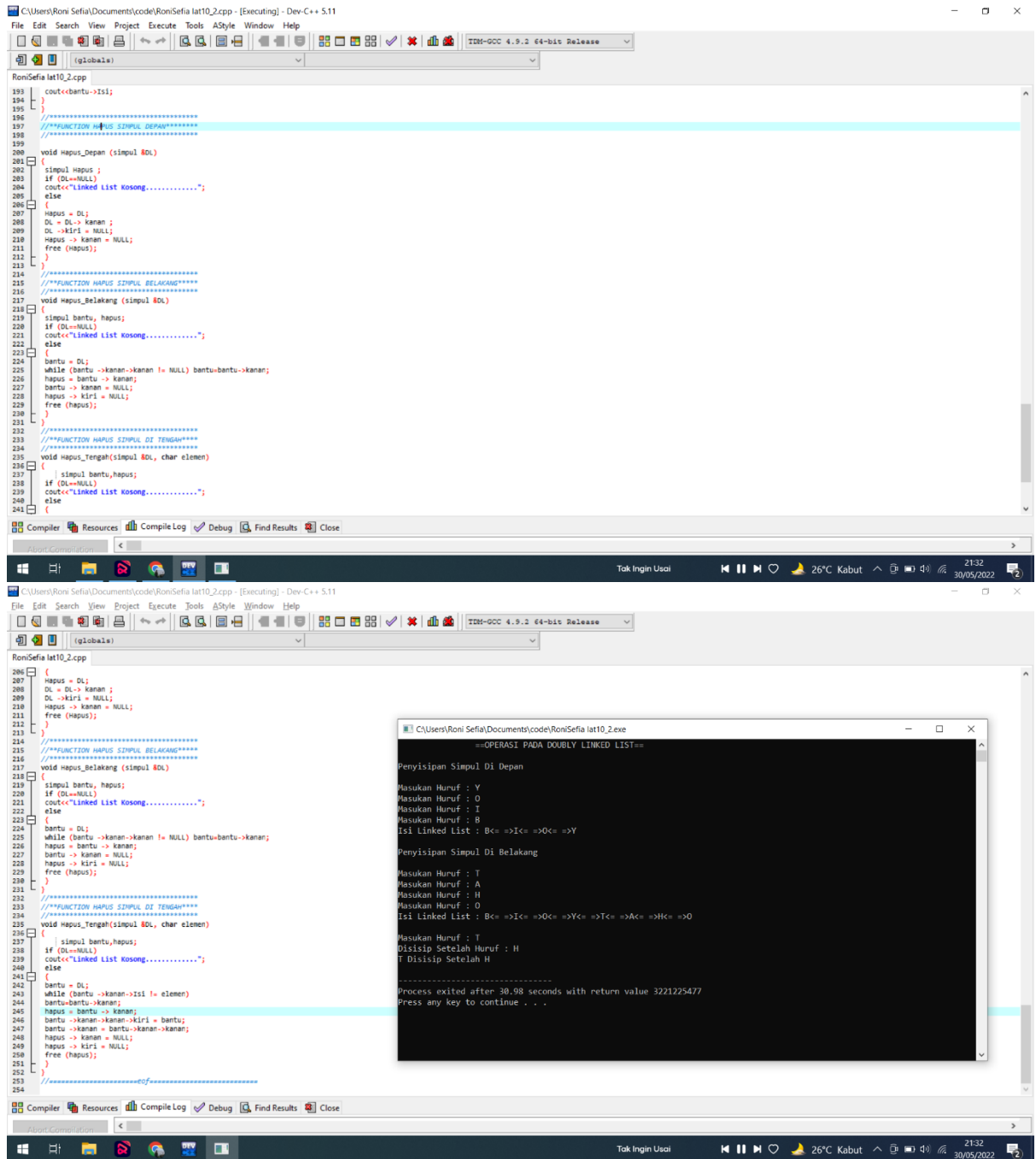
```
1 #include<iostream>
2 #include<conio.h>
3 #include<stdlib.h>
4 #define true 1
5 #define false 0
6 using namespace std;
7 typedef struct node *simpul;
8 struct node
9 {
10     char Isi;
11     simpul kanan;
12     simpul kiri;
13 };
14 //=====
15 //prototype function
16 //=====
17 void Sisip_Depan (simpul &DL, char elemen );
18 void Sisip_Belakang (simpul &DL, char elemen );
19 void Sisip_Tengah (simpul &DL, char elemen1, char elemen2 );
20 void Sisip_Tengah2 (simpul &DL, char elemen1, char elemen2 );
21 void Hapus_Depan (simpul &DL);
22 void Hapus_Belakang (simpul &DL);
23 void Hapus_Tengah (simpul &DL, char elemen);
24 void cetak (simpul DL);
25 //=====
26 //function main
27 //=====
28 main ()
29 {
30     char huruf, huruf2;
31     simpul DL = NULL; //Pastikan Bahwa DL Kosong
32     int i;
33     cout<<"\t\t==OPERASI PADA DOUBLY LINKED LIST==\n\n";
34     //=====
35     //==Sisip Depan==
36     cout<<"Penyisipan Simpul Di Depan<<endl;
37     for (i=1;i<=4;i++)
38     {
39         cout<<"Masukan Huruf : "; cin>>huruf;
40         Sisip_Depan (DL, huruf );
41     }
42     cetak (DL);
43     //=====
44     //==Sisip Belakang==
45     //=====
46     cout<<"\n\nPenyisipan Simpul Di Belakang<<endl;
47     for (i=1;i<=4;i++)
48     {
49         cout<<"Masukan Huruf : "; cin>>huruf;
50         Sisip_Belakang (DL, huruf );
51     }
52     cetak (DL);
53     //=====
54     //==Sisip Simpul Setelah Simpul Tertentu==
55     //=====
56     cout<<endl<<"Masukan Huruf : "; cin>>huruf;
57     cout<<"Disisip Setelah Huruf : "; cin>>huruf2;
58     cout<<huruf<<" Disisip Setelah "<<huruf2<<endl;
59     Sisip_Tengah (DL, huruf, huruf2);
60     cetak (DL );
61     //=====
62     //==Sisip Simpul Sebelum Simpul Tertentu==
63     //=====
64     cout<<endl<<"Masukan Huruf : "; cin>>huruf;
65     cout<<"Disisip Sebelum Huruf : "; cin>>huruf2;
66     cout<<huruf<<" Disisip Sebelum "<<huruf2<<endl;
67     Sisip_Tengah2 (DL, huruf, huruf2);
68     cetak (DL );
69     //=====
70     //==Hapus Simpul Depan==
71     //=====
72     cout<<endl<<"Setelah Hapus Simpul Depan "<<endl;
73     Hapus_Depan (DL);
74     cetak (DL);
75     //=====
76     //==Hapus Simpul Belakang==
77     //=====
78     cout<<endl<<"Setelah Hapus Simpul Belakang "<<endl;
79     Hapus_Belakang (DL);
80     cetak (DL);
81     //=====
82     //==Hapus Simpul TENGGAH==
83     //=====
84     cout<<"\n\nMasukkan Huruf Tengah Yang akan dihapus :";cin>>huruf;
85     hapus_Tengah (DL,huruf);
86     cetak (DL);
87     getch ( ) ;
88 }
89 //=====
90 //=====
91 //=====
92 //=====
93 void Sisip_Depan (simpul &DL, char elemen)
94 {
95     simpul baru;
96     baru = (simpul) malloc (sizeof(simpul));
97     baru->Isi = elemen ;
```

**Bottom Screenshot:** Shows the continuation of the program, including the `Sisip_Depan` function implementation and the `main` function's loop for inserting at the front.

```
98     simpul baru;
99     baru = (simpul) malloc (sizeof(simpul));
100    baru->Isi = elemen ;
101    if (DL == NULL)
102    {
103        DL = baru;
104        DL->kanan = NULL;
105        DL->kiri = NULL;
106    }
107    else
108    {
109        simpul temp = DL;
110        while (temp->kiri != NULL)
111            temp = temp->kiri;
112        temp->kiri = baru;
113        baru->kiri = temp;
114        baru->kanan = DL;
115        DL->kiri = baru;
116    }
117 }
```

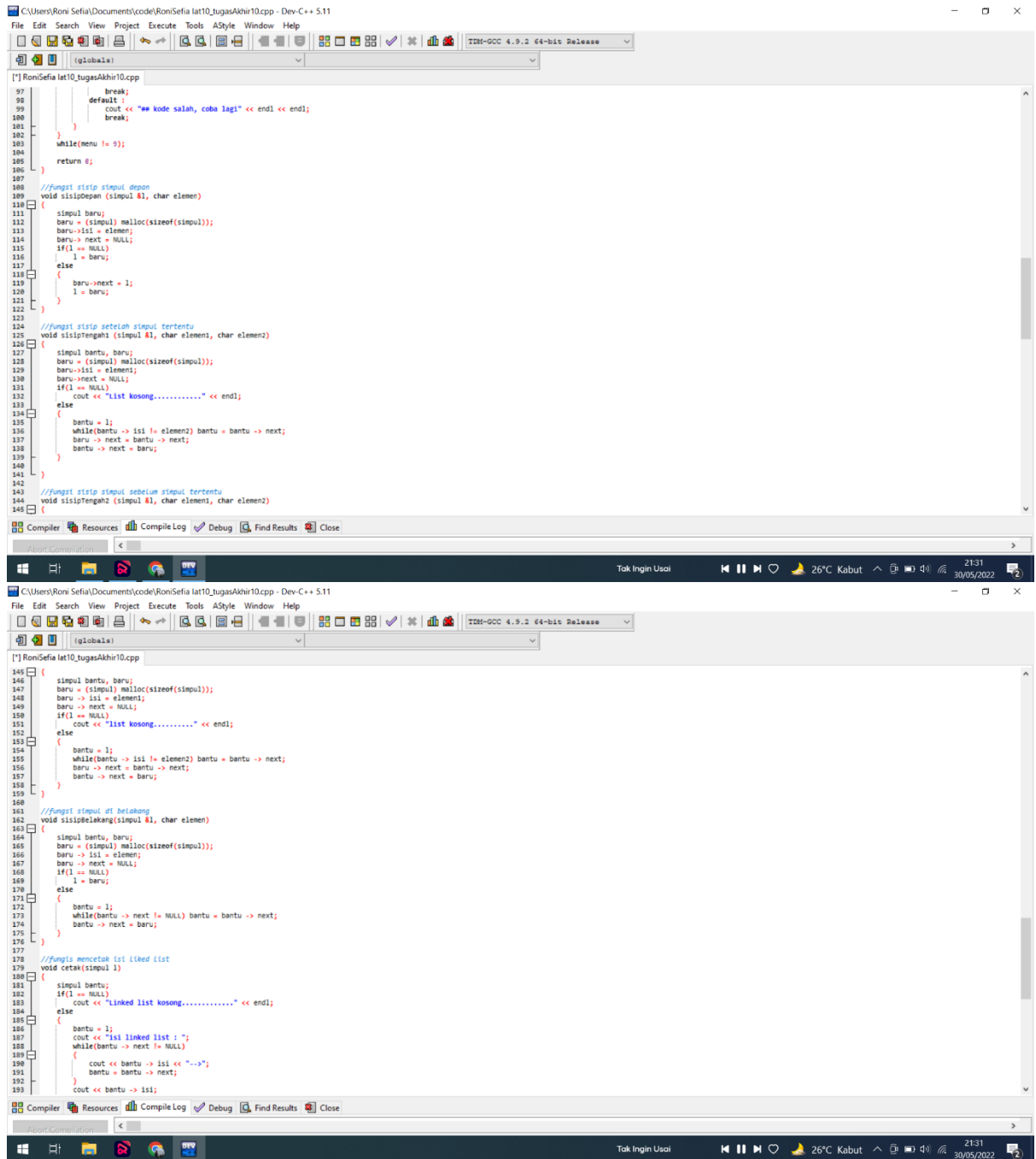


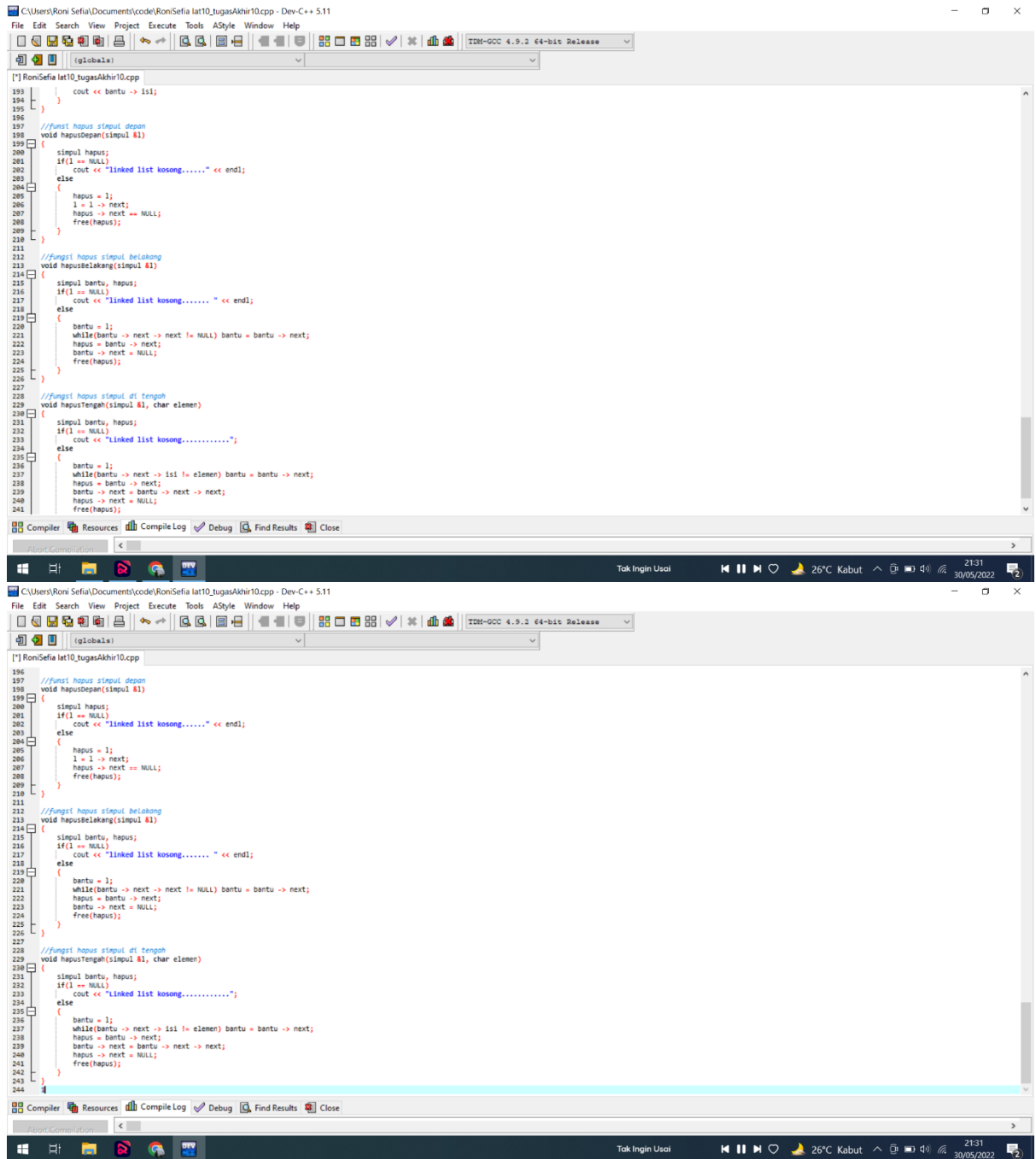




## TUGAS AKHIR 10

```
C:\Users\Roni Sefia\Documents\code\RoniSefia\lat10_tugasAkhir10.cpp - Dev-C++ 5.11
File Edit Search View Project Execute Tools AStyle Window Help
(globals)
[+] RoniSefia\lat10_tugasAkhir10.cpp
1 #include<iostream>
2 #include<cstdlib>
3
4 using namespace std;
5
6 typedef struct node *simpul;
7
8 struct node
9 {
10     char isi;
11     simpul next;
12 };
13
14 //prototype functions
15
16 void sisipdepan(simpul &l, char elemen);
17 void sisipbelakang(simpul &l, char elemen);
18 void sisiptengah1(simpul &l, char elemen1, char elemen2);
19 void sisiptengah2(simpul &l, char elemen1, char elemen2);
20 void hapusdepan(simpul &l);
21 void hapusbelakang(simpul &l);
22 void hapustengah(simpul &l, char elemen);
23 void cetak(simpul l);
24
25 int main()
26 {
27     char huruf, huruf2;
28     simpul l = NULL; //pastikan bahwa l kosong
29     int menu;
30     cout << "OPERASI PADA SINGLE LINKED LIST" << endl << endl;
31
32     do
33     {
34         cout << "Menu : " << endl;
35         cout << "1. sisip depan" << endl;
36         cout << "2. sisip belakang" << endl;
37         cout << "3. sisip tengah" << endl;
38         cout << "4. sisip tengah" << endl;
39         cout << "5. hapus depan" << endl;
40         cout << "6. hapus belakang" << endl;
41         cout << "7. hapus tengah" << endl;
42         cout << "8. cetak" << endl;
43         cout << "9. keluar" << endl;
44
45         cout << "Pilih menu : ";
46         cin >> menu;
47         cout << endl;
48
49         switch(menu)
50         {
51             case 1 :
52                 cout << "Masukkan huruf : ";
53                 cin >> huruf;
54                 sisipdepan(l, huruf);
55                 cout << endl;
56                 break;
57             case 2 :
58                 cout << "Masukkan huruf : ";
59                 cin >> huruf;
60                 sisipbelakang(l, huruf);
61                 cout << endl;
62                 break;
63             case 3 :
64                 cout << "Masukkan huruf : ";
65                 cin >> huruf; cout << endl;
66                 cout << "Masukkan sisip setelah huruf : ";
67                 cin >> huruf2; cout << endl;
68                 sisiptengah1(l, huruf, huruf2);
69                 break;
70             case 4 :
71                 cout << "Masukkan huruf : ";
72                 cin >> huruf; cout << endl;
73                 cout << "Masukkan sisip sebelum huruf : ";
74                 cin >> huruf2; cout << endl;
75                 sisiptengah2(l, huruf, huruf2);
76                 break;
77             case 5 :
78                 hapusdepan(l);
79                 cout << "Simpul depan dihapus" << endl << endl;
80                 break;
81             case 6 :
82                 hapusbelakang(l);
83                 cout << "Simpul belakang dihapus" << endl << endl;
84                 break;
85             case 7 :
86                 cout << "Masukkan huruf tengah yang akan dihapus : ";
87                 cin >> huruf;
88                 hapustengah(l, huruf);
89                 cout << endl;
90                 break;
91             case 8 :
92                 cetak(l);
93                 cout << endl << endl;
94                 break;
95             case 9 :
96                 cout << "Keluar program..." << endl;
97                 break;
98         }
99     } while (menu != 9);
100 }
```





C:\Users\Roni Sefia\Documents\code\RoniSefia\lat10\_tugasAkhir10.exe

OPERASI PADA SINGLE LINKED LIST

Menu :

1. sisip depan
2. sisip belakang
3. sisip tengah1
4. sisip tengah2
5. hapus depan
6. hapus belakang
7. hapus tengah
8. cetak
9. keluar

Pilih menu : 1

## Masukan huruf : Y

Menu :

1. sisip depan
2. sisip belakang
3. sisip tengah1
4. sisip tengah2
5. hapus depan
6. hapus belakang
7. hapus tengah
8. cetak
9. keluar

Pilih menu : 2

## Masukan huruf : K

Menu :

1. sisip depan
2. sisip belakang
3. sisip tengah1
4. sisip tengah2
5. hapus depan
6. hapus belakang
7. hapus tengah
8. cetak
9. keluar

Pilih menu : 3

## Masukan huruf : O

## Disisip setelah huruf : Y

Menu :

1. sisip depan
2. sisip belakang
3. sisip tengah1

Tak Ingin Usai 21:30 30/05/2022

C:\Users\Roni Sefia\Documents\code\RoniSefia\lat10\_tugasAkhir10.exe

## Masukan huruf : O

## Disisip setelah huruf : Y

Menu :

1. sisip depan
2. sisip belakang
3. sisip tengah1
4. sisip tengah2
5. hapus depan
6. hapus belakang
7. hapus tengah
8. cetak
9. keluar

Pilih menu : 4

## Masukan huruf : P

## Disisip sebelum huruf : Y

Menu :

1. sisip depan
2. sisip belakang
3. sisip tengah1
4. sisip tengah2
5. hapus depan
6. hapus belakang
7. hapus tengah
8. cetak
9. keluar

Pilih menu : 8

isi linked list : Y-->P-->O-->K

Menu :

1. sisip depan
2. sisip belakang
3. sisip tengah1
4. sisip tengah2
5. hapus depan
6. hapus belakang
7. hapus tengah
8. cetak
9. keluar

Pilih menu : 7

## Masukan huruf tengah yang akan dihapus : P

Menu :

1. sisip depan

Tak Ingin Usai 21:30 30/05/2022

C:\Users\Roni Sefia\Documents\code\RoniSefia\lat10\_tugasAkhir10.exe

```
Menu :
1. sisip depan
2. sisip belakang
3. sisip tengah1
4. sisip tengah2
5. hapus depan
6. hapus belakang
7. hapus tengah
8. cetak
9. keluar
Pilih menu : 5
```

## Simpul depan dihapus

```
Menu :
1. sisip depan
2. sisip belakang
3. sisip tengah1
4. sisip tengah2
5. hapus depan
6. hapus belakang
7. hapus tengah
8. cetak
9. keluar
Pilih menu : 6
```

## Simpul belakang dihapus

```
Menu :
1. sisip depan
2. sisip belakang
3. sisip tengah1
4. sisip tengah2
5. hapus depan
6. hapus belakang
7. hapus tengah
8. cetak
9. keluar
Pilih menu : 8
```

isi linked list : 0

```
Menu :
1. sisip depan
2. sisip belakang
3. sisip tengah1
4. sisip tengah2
5. hapus depan
6. hapus belakang
7. hapus tengah
```

Tak Ingin Usai 21:30 30/05/2022

C:\Users\Roni Sefia\Documents\code\RoniSefia\lat10\_tugasAkhir10.exe

```
1. sisip depan
2. sisip belakang
3. sisip tengah1
4. sisip tengah2
5. hapus depan
6. hapus belakang
7. hapus tengah
8. cetak
9. keluar
Pilih menu : 6
```

## Simpul belakang dihapus

```
Menu :
1. sisip depan
2. sisip belakang
3. sisip tengah1
4. sisip tengah2
5. hapus depan
6. hapus belakang
7. hapus tengah
8. cetak
9. keluar
Pilih menu : 8
```

isi linked list : 0

```
Menu :
1. sisip depan
2. sisip belakang
3. sisip tengah1
4. sisip tengah2
5. hapus depan
6. hapus belakang
7. hapus tengah
8. cetak
9. keluar
Pilih menu : 9
```

## Keluar program...

Process exited after 81.64 seconds with return value 0  
Press any key to continue . . .

Tak Ingin Usai 21:31 30/05/2022

# **LAPORAN AWAL**

## **PRAKTIKUM**

## **STRUKTUR DATA**

**LAPORAN KE - 10**



**Disusun Oleh :**

**Nama : Roni Sefia**  
**NIM : 201011401617**  
**Kelas : 04TPLP016**

**TEKNIK INFORMATIKA**  
**FAKULTAS TEKNIK**  
**UNIVERSITAS PAMULANG**

**Jl. Surya Kencana No.1, Pamulang Bar., Kec. Pamulang, Kota Tangerang**  
**Selatan, Banten 15417**

# **LAPORAN AKHIR**

## **PRAKTIKUM**

## **STRUKTUR DATA**

**LAPORAN KE - 9**



**Disusun Oleh :**

**Nama : Roni Sefia**  
**NIM : 201011401617**  
**Kelas : 04TPLP016**

**TEKNIK INFORMATIKA**  
**FAKULTAS TEKNIK**  
**UNIVERSITAS PAMULANG**

**Jl. Surya Kencana No.1, Pamulang Bar., Kec. Pamulang, Kota Tangerang**  
**Selatan, Banten 15417**