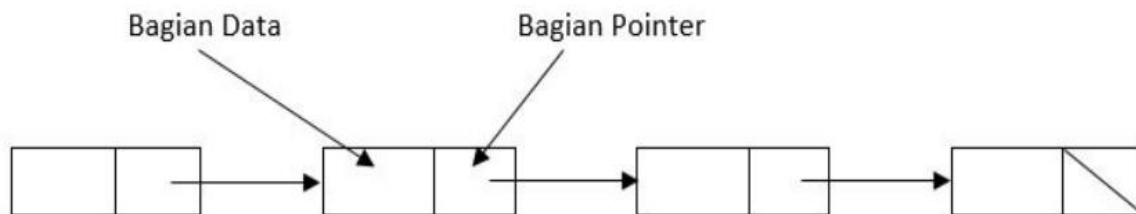


## BAB 4. Linked List

### 4.1 Definisi Linked List

Linked list adalah struktur data yang terdiri dari sejumlah simpul (node) yang terhubung satu sama lain melalui suatu tautan (pointer atau referensi). Setiap simpul pada linked list menyimpan data dan tautan ke simpul berikutnya dalam linked list.



Dari gambar diatas kita lihat bahwa masing-masing simpul terbagi atas dua bagian, yaitu bagian Isi dan bagian Pointer, maka dengan demikian dapat dibuat dalam struct yang terdiri dari 2 field, seperti berikut ini.

```
typedef struct
struct node
{
    type_data Isi;
    simpul->next;
}
```

### 4.2 Single Linked List

Single linked list adalah struktur data linier yang terdiri dari serangkaian simpul (node) yang terhubung satu sama lain dengan satu tautan (pointer atau referensi) ke simpul berikutnya pada linked list. Setiap simpul pada single linked list menyimpan data dan tautan ke simpul berikutnya. Single linked list umumnya memiliki satu tautan yang menghubungkan setiap simpul, sehingga kita hanya bisa mengakses simpul-simpul pada linked list secara sekuensial dari head sampai ke simpul terakhir.

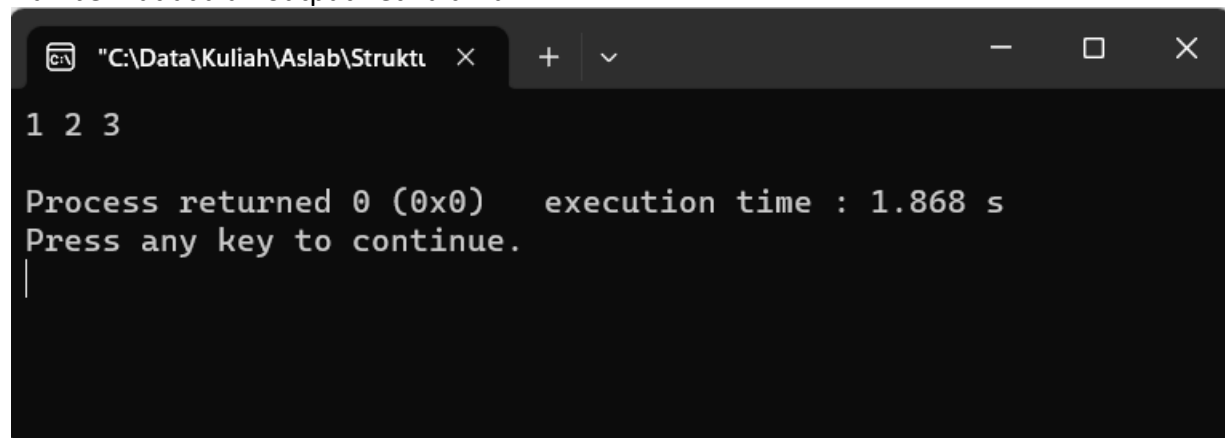
Berikut adalah contoh source code dari single linked list

```

2   using namespace std;
3
4   class Node {
5   public:
6       int data;
7       Node* next;
8   };
9
10  int main() {
11      // Buat simpul pertama dan isinya
12      Node* head = new Node();
13      head->data = 1;
14      head->next = NULL;
15
16      // Buat simpul kedua dan isinya
17      Node* second = new Node();
18      second->data = 2;
19      second->next = NULL;
20
21      // Sambungkan simpul kedua ke simpul pertama
22      head->next = second;
23
24      // Buat simpul ketiga dan isinya
25      Node* third = new Node();
26      third->data = 3;
27      third->next = NULL;
28
29      // Sambungkan simpul ketiga ke simpul kedua
30      second->next = third;
31
32      // Cetak isi setiap simpul pada linked list
33      Node* node = head;
34      while (node != NULL) {
35          cout << node->data << " ";
36          node = node->next;
37      }
38      cout << endl;
39
40      return 0;
41  }

```

Dan berikut adalah output Ketika di run



```

C:\Data\Kuliah\Aslab\Strukt...
1 2 3

Process returned 0 (0x0)   execution time : 1.868 s
Press any key to continue.

```

### 3.3 Double Linked List

Double linked list adalah struktur data linier yang terdiri dari serangkaian simpul (node) yang terhubung satu sama lain dengan dua tautan (pointer atau referensi), yaitu tautan ke simpul sebelumnya dan tautan ke simpul berikutnya pada linked list. Setiap simpul pada double linked list menyimpan data dan dua tautan ke simpul sebelumnya dan berikutnya. Hal ini memungkinkan kita untuk mengakses simpul-simpul pada linked list secara sekuensial dari head sampai ke simpul terakhir atau sebaliknya dari tail sampai ke simpul pertama.

Berikut adalah contoh source code dari double linked list

```
1  #include <iostream>
2  #include <conio.h>
3  #include <windows.h>
4  using namespace std;
5
6
7  struct dataKtp {
8      char nama[50];
9      char alamat[50];
10     char golongan_darah[2];
11     dataKtp *next, *prev;
12 };
13 dataKtp *awal=NULL,*akhir=NULL,*bantu,*baru;
14
15
16 void inputDatadepan() {
17     baru = new dataKtp;
18     baru->next = NULL;
19     baru->prev = NULL;
20     cout << "Nama\t\t: ";
21     cin >> baru->nama;
22     cout << "Alamat\t\t: ";
23     cin >> baru->alamat;
24     cout << "Golongan darah : ";
25     cin >> baru->golongan_darah;
26     if (awal == NULL) {
27         awal = akhir = baru;
28     }
29     else {
30         baru->next = awal;
31         awal->prev = baru;
32         awal = baru;
33     }
34     system ("cls");
35 }
36
```

```

38 void inputDatabelakang() {
39     baru = new dataKtp;
40     baru->next = NULL;
41     baru->prev = NULL;
42     cout << "Nama\t\t: ";
43     cin >> baru->nama;
44     cout << "Alamat\t\t: ";
45     cin >> baru->alamat;
46     cout << "Golongan darah : ";
47     cin >> baru->golongandarah;
48     if (awal == NULL) {
49         awal = akhir = baru;
50     }
51
52     else {
53         akhir->next = baru;
54         baru->prev = akhir;
55         akhir = baru;
56     }
57     system ("cls");
58 }
59
60 void outputData () {
61     bantu=awal;
62     while (bantu != NULL) {
63         cout << "Nama\t\t: "<<bantu->nama<<endl;
64         cout << "Alamat\t\t: "<<bantu->alamat<<endl;
65         cout << "Golongan darah : "<<bantu->golongandarah<<endl;
66         bantu = bantu->next;
67     }
68 }
69
70
71 int main()
72 {
73     int pilihan;
74     do {
75         cout << "-----DOUBLE LINKED LIST-----\n";
76         cout << "Pilih menu : \n";
77         cout << "1. Input data di depan\n";
78         cout << "2. Input data di belakang\n";
79         cout << "3. Output data\n";
80         cout << "4. EXIT\n";
81         cout << "Pilihan : ";
82         cin >> pilihan;
83         switch (pilihan){
84             case 1 :
85                 inputDatadepan();
86                 break;
87             case 2 :
88                 inputDatabelakang();
89                 break;
90             case 3 :
91                 if (baru == NULL){
92                     cout << "Data masih kosong !\n";

```

```

93         }
94     else {
95         outputData();
96     }
97     break;
98     default :
99         cout << "-----EXIT-----";
100        break;
101    }
102    }while(pilihan <= 3);
103    getch();
104    }

```

Dan ini adalah output Ketika di run

```

-----DOUBLE LINKED LIST-----
Pilih menu :
1. Input data di depan
2. Input data di belakang
3. Output data
4. EXIT
Pilihan : |

```

### LATIHAN:

Buat program linked list yang dapat membalikkan data (reverse) seperti gambar dibawah

```
Original Linked List: 5 4 3 2 1  
Reversed Linked List: 1 2 3 4 5
```