

TP MODUL 5

NAMA : Ahmad Naufal Ramadhan

NIM : 103012300239

KELAS : IF-47-02

Header.h

```
main.cpp x header.h x header.cpp x
1  #ifndef HEADER_H_INCLUDED
2  #define HEADER_H_INCLUDED
3
4  #include <iostream>
5  #define first(L) L.first
6  #define next(P) P->next
7  #define info(P) P->info
8
9  using namespace std;
10
11  /*103012300239, 0 adalah digit ke-9 NIM. 0 mod 3 adalah 0,|
12  maka data yang disimpan adalah int*/
13
14  typedef int infotype;
15  typedef struct elmList *address;
16
17  struct elmList {
18      infotype info;
19      address next;
20  };
21
22  struct List{
23      address first;
24  };
25
26  void createList_103012300239(List &L);
27  address alokasi_103012300239(infotype x);
28  void insertLast_103012300239(List &L, address p);
29  void show_103012300239(List &L);
30  address findMin_103012300239(List L);
31  void insertMid_103012300239(List &L, int value);
32  int menu_103012300239();
33
34  #endif // HEADER_H_INCLUDED
35
```

Header.cpp

```
main.cpp x header.h x header.cpp x
1  #include <iostream>
2  #include "header.h"
3
4  using namespace std;
5
6  void createList_103012300239(List &L){
7      first(L) = NULL;
8  }
9
10 address alokasi_103012300239(infotype x){
11     address P = new elmList;
12     info(P) = x;
13     next(P) = NULL;
14     return P;
15 }
16
17 void insertLast_103012300239(List &L, address p){
18     address q;
19     if (first(L) == NULL){
20         first(L) = p;
21     } else {
22         q = first(L);
23         while (next(q) != NULL) {
24             q = next(q);
25         }
26         next(q) = p;
27     }
28 }
29
30 void show_103012300239(List &L){
31     address p = first(L);
32     while (p != NULL){
33         cout << info(p);
34         if (next(p) != NULL) cout << ", ";
35         p = next(p);
36     }
37     cout << endl;
38 }
39
40 address findMin_103012300239(List L){
41     address minim, q;
42     minim = first(L);
43     q = first(L);
44     while (next(q) != NULL){
45         q = next(q);
46         if (info(q) < info(minim)){
47             minim = q;
48         }
49     }
50     return minim;
51 }
```

```

53 void insertMid_103012300239(List &L, int value){
54     int n, i;
55     address q = alokasi_103012300239(value);
56     address p = first(L);
57     n = 1;
58     while (next(p) != NULL){
59         p = next(p);
60         n++;
61     }
62     n = (n+1) / 2;
63     p = first(L);
64     for (i = 1; i < n; i++){
65         p = next(p);
66     }
67     next(q) = next(p);
68     next(p) = q;
69 }
70
71 int menu_103012300239(){
72     cout << "=====MENU===== " << endl;
73     cout << "1. Menambahkan N data baru" << endl;
74     cout << "2. Menampilkan semua data" << endl;
75     cout << "3. Menampilkan data terkecil" << endl;
76     cout << "4. Menyisipkan data di tengah" << endl;
77     cout << "0. EXIT" << endl;
78     cout << "Masukkan menu: ";
79
80     int input;
81     cin >> input;
82     return input;
83 }

```

Main.cpp

```
main.cpp x header.h x header.cpp x
1  #include <iostream>
2  #include "header.h"
3
4  using namespace std;
5
6  int main(){
7      List L;
8      address p;
9      int input, x, n, i;
10
11     createList_103012300239(L);
12
13     input = menu_103012300239();
14     while (input != 0){
15         switch(input){
16             case 1:
17                 cout << "Jumlah data yang akan ditambahkan: ";
18                 cin >> n;
19                 i = 1;
20                 while (i <= n){
21                     cout << "Masukkan data baru: ";
22                     cin >> x;
23                     p = alokasi_103012300239(x);
24                     insertLast_103012300239(L, p);
25                     i++;
26                 }
27                 cout << endl;
28                 break;
29
30             case 2:
31                 show_103012300239(L);
32                 cout << endl;
33                 break;
34
35             case 3:
36                 p = findMin_103012300239(L);
37                 cout << "Data terkecil: " << info(p) << endl;
38                 cout << endl;
39                 break;
40
41             case 4:
42                 cout << "Masukkan data yang ingin disisipkan: ";
43                 cin >> x;
44                 insertMid_103012300239(L, x);
45                 cout << endl;
46                 break;
47         }
48         input = menu_103012300239();
49     }
50     cout << endl;
51     cout << "ANDA TELAH KELUAR DARI PROGRAM";
52     cout << endl;
53
54     return 0;
55 }
```

running

```
"C:\Users\LENOVO\Desktop\S" × + v
=====MENU=====
1. Menambahkan N data baru
2. Menampilkan semua data
3. Menampilkan data terkecil
4. Menyisipkan data di tengah
0. EXIT
Masukkan menu: 1
Jumlah data yang akan ditambahkan: 5
Masukkan data baru: 1
Masukkan data baru: 2
Masukkan data baru: 3
Masukkan data baru: 4
Masukkan data baru: 5

=====MENU=====
1. Menambahkan N data baru
2. Menampilkan semua data
3. Menampilkan data terkecil
4. Menyisipkan data di tengah
0. EXIT
Masukkan menu: 2
1, 2, 3, 4, 5

=====MENU=====
1. Menambahkan N data baru
2. Menampilkan semua data
3. Menampilkan data terkecil
4. Menyisipkan data di tengah
0. EXIT
Masukkan menu: 3
Data terkecil: 1

=====MENU=====
1. Menambahkan N data baru
2. Menampilkan semua data
3. Menampilkan data terkecil
4. Menyisipkan data di tengah
0. EXIT
Masukkan menu: 4
Masukkan data yang ingin disisipkan: 8

=====MENU=====
1. Menambahkan N data baru
2. Menampilkan semua data
3. Menampilkan data terkecil
4. Menyisipkan data di tengah
0. EXIT
Masukkan menu: 2
1, 2, 3, 8, 4, 5

=====MENU=====
1. Menambahkan N data baru
2. Menampilkan semua data
3. Menampilkan data terkecil
4. Menyisipkan data di tengah
0. EXIT
Masukkan menu: 0

ANDA TELAH KELUAR DARI PROGRAM

Process returned 0 (0x0)   execution time : 30.319 s
Press any key to continue.
```