

TP Modul 4 - Single Linked List (part 1)

- list.h :

```
list.h x
1  #include <iostream>
2  #define first(L) L.first
3  #define next(P) P->next
4  #define info(P) P->info
5  using namespace std;
6  typedef int infotype;
7  typedef struct elmlist *address;
8
9  struct elmlist{
10     infotype info;
11     address next;
12 };
13 struct List{
14     address first;
15 };
16
17 void createList(List &L);
18 address allocate(infotype x);
19 void insertFirst(List &L, address P);
20 void printInfo(List L);
21
```

- list.cpp :

```
list.cpp x
1  #include <iostream>
2  #include "list.h"
3  using namespace std;
4
5  void createList(List &L){
6     first(L) = NULL;
7 }
8
9  address allocate(infotype x){
10     address P = new elmlist;
11     info(P) = x;
12     next(P) = NULL;
13
14     return P;
15 }
16
17 void insertFirst(List &L, address P){
18     next(P) = first(L);
19     first(L) = P;
20 }
21
22 void printInfo(List L){
23     address P = first(L);
24     while (P != NULL){
25         cout << info(P) << ", ";
26         P = next(P);
27     }
28     cout << endl;
29 }
```

- main.cpp :

```
main.cpp x
1  #include <iostream>
2  #include "list.h"
3  using namespace std;
4
5  int main()
6  {
7      List L;
8      address P;
9      int x;
10     createList(L);
11     cout << "Masukkan NIM perdigit" << endl;
12     cout << "Digit 1: ";
13     cin >> x;
14     P = allocate(x);
15     insertFirst(L, P);
16     cout << "Digit 2: ";
17     cin >> x;
18     P = allocate(x);
19     insertFirst(L, P);
20     cout << "Digit 3: ";
21     cin >> x;
22     P = allocate(x);
23     insertFirst(L, P);
24     printInfo(L);
25     return 0;
26 }
```

- output :

```
"D:\KuliahGena\SMT 3\Strukt  x  +  v
Masukkan NIM perdigit
Digit 1: 0
Digit 2: 9
Digit 3: 5
5, 9, 0,

Process returned 0 (0x0)   execution time : 3.267 s
Press any key to continue.
|
```

7. sesi have fun :

- list.h :

```
list.h x
1  #include <iostream>
2  #define first(L) L.first
3  #define next(P) P->next
4  #define info(P) P->info
5  #define next(Q) Q->next
6  #define info(Q) Q->info
7  #define next(prec) prec->next
8  #define info(prec) prec->info
9  using namespace std;
10 typedef int infotype;
11 typedef struct elmList *address;
12
13 struct elmList{
14     infotype info;
15     address next;
16 };
17 struct List{
18     address first;
19 };
20
21 void createList(List &L);
22 address allocate(infotype x);
23 void insertFirst(List &L, address P);
24 void printInfo(List L);
25 void insertLast(List &L, address P);
26 void insertAfter(List &L, address prec, address P);
27 address deleteFirst(List &L, address P);
28 address deleteLast(List &L, address P);
29 address deleteAfter(List &L, address prec, address P);
30
```

- list.cpp :

```
list.cpp x
1  #include <iostream>
2  #include "list.h"
3  using namespace std;
4
5  void createList(List &L){
6      first(L) = NULL;
7  }
8
9  address allocate(infotype x){
10     address P = new elmList;
11     info(P) = x;
12     next(P) = NULL;
13
14     return P;
15 }
16
17 void insertFirst(List &L, address P){
18     next(P) = first(L);
19     first(L) = P;
20 }
21
22 void printInfo(List L){
23     address P = first(L);
24     while (P != NULL){
25         cout << info(P);
26         P = next(P);
27     }
28     cout << endl;
29 }
```

```
list.cpp X
31 void insertLast(List &L, address P){
32     address Q;
33     Q = first(L);
34     while (next(Q) != NULL) {
35         Q = next(Q);
36     }
37     next(Q) = P;
38 }
39
40 void insertAfter(List &L, address prec, address P){
41     next(P) = next(prec);
42     next(prec) = P;
43 }
44
45 address deleteFirst(List &L, address P){
46     P = first(L);
47     first(L) = next(P);
48     next(P) = NULL;
49     return P;
50 }
```

```
*list.cpp X
52 address deleteLast(List &L, address P){
53     address Q, R;
54     Q = first(L);
55     while (next(Q) != NULL){
56         R = Q;
57         Q = next(Q);
58     }
59     P = next(R);
60     next(R) = NULL;
61     return P;
62 }
63
64 address deleteAfter(List &L, address prec, address P){
65     P = next(prec);
66     next(prec) = next(P);
67     next(P) = NULL;
68 }
```

- main.cpp :

```
main.cpp X
1  #include <iostream>
2  #include "list.h"
3  using namespace std;
4
5  int main()
6  {
7      List L;
8      address P;
9      int x, i;
10     createList(L);
11     cout << "Masukkan NIM perdigit" << endl;
12     for (i = 1; i <= 12; i++){
13         cout << "Digit " << i << ": ";
14         cin >> x;
15         P = allocate(x);
16         if (i == 1) {
17             insertFirst(L, P);
18         } else {
19             insertLast(L, P);
20         }
21     }
22     printInfo(L);
23     return 0;
24 }
```

- output :

```
"D:\KuliahGena\SMT 3\Strukt" × + v
Masukkan NIM perdigit
Digit 1: 1
Digit 2: 0
Digit 3: 3
Digit 4: 0
Digit 5: 3
Digit 6: 2
Digit 7: 3
Digit 8: 3
Digit 9: 0
Digit 10: 0
Digit 11: 9
Digit 12: 5
103032330095

Process returned 0 (0x0)   execution time : 9.528 s
Press any key to continue.
```