



doublelinkedlist.cpp × main.cpp × doublelinkedlist.h ×

```
1  #ifndef DOUBLELINKEDLIST_H_INCLUDED
2  #define DOUBLELINKEDLIST_H_INCLUDED
3  #include <iostream>
4  #define first(L) L.first
5  #define last(L) L.last
6  #define info(P) P->info
7  #define next(P) P->next
8  #define prev(P) P->prev
9
10 /* NAMA: FAHMI RAZAN RAMDANI
11    NIM: 1301194054 */
12 using namespace std;
13 typedef int infotype;
14 typedef struct elmlist *address;
15
16 struct elmlist
17 {
18     infotype info;
19     address next, prev;
20 };
21
22 struct List
23 {
24     address first, last;
25 };
26
27 bool isEmpty(List L);
28 void createList(List &L);
29 void createNewElmt(infotype X, address &P);
30 void insertFirst(List &L, address P);
31 void insertAfter(List &L, address Prec, address P);
32 void insertLast(List &L, address P);
33 void deleteFirst(List &L, address &P);
34 void deleteAfter(List &L, address Prec, address &P);
35 void deleteLast(List &L, address &P);
36 void concat(List L1, List L2, List &L3);
37 double median(List L, infotype jumlah);
38 void showList(List L);
39 address cariElm(List L, address &P, infotype X);
40 #endif // DOUBLELINKEDLIST_H_INCLUDED
41
```

main.cpp [TP4] - Code::Blocks 17.12

File Edit View Search Project Build Debug Fortran wxSmith Tools Tools+ Plugins DoxyBlocks Settings Help

doublelinkedlist.cpp x main.cpp x doublelinkedlist.h x

```
1  #include "doublelinkedlist.h"
2  /* NAMA: FAHMI RAZAN RAMDANI
3   * NIM: 1301194054 */
4
5  int main()
6  {
7      infotype X, Y, jumlah, jumlah2;
8      float i;
9      address P, Prec;
10     bool cek;
11     List L, L1, L2, L3;
12     cek = false;
13     X = 0;
14     jumlah = 1;
15     Y = 0;
16
17     createList(L);
18     createList(L1);
19     createList(L2);
20     createList(L3);
21     cout<<"cek list, list kosong: ";
22     cek = isEmpty(L);
23     if(cek == true)
24     {
25         cout<<"ya\n";
26     }else{
27         cout<<"tidak\n";
28     }
29
30     cout<<"Insert first:\n";
31     cin>>X;
32     while(X != 0){
33         createNewElmt(X, P);
34         insertFirst(L, P);
35         jumlah++;
36         cin>>X;
37     }
38     showList(L);
39     cout<<endl;
40
41     cout<<"Insert after:\n";
42     cout<<"Input: ";
43     cin>>X;
44     createNewElmt(X, P);
45     cout<<"Setelah: ";
46     cin>>Y;
47     Prec = cariElm(L, Prec, Y);
48     insertAfter(L, Prec, P);
49     showList(L);
50     cout<<endl;
51
52     cout<<"Insert last:\n";
53     cin>>X;
54     while(X != 0){
55         createNewElmt(X, P);
56         insertLast(L, P);
57         jumlah++;
58         cin>>X;
59     }
60     showList(L);
61     cout<<endl;
62
63     cout<<"Delete first:\n";
64     deleteFirst(L, P);
65     showList(L);
66     jumlah = jumlah - 1;
67     cout<<endl;
68
69     cout<<"Delete after:\n";
70     cout<<"Hapus: ";
```

C/C++ Windows (CR+LF) WINDOWS-1252 Line 3, Col 14, Pos 74 Insert Read/W

```
main.cpp [TP4] - Code::Blocks 17.12
File Edit View Search Project Build Debug Fortran wxSmith Tools Tools+ Plugins DoxyBlocks Settings Help

doublelinkedlist.cpp x main.cpp x doublelinkedlist.h x
53     cin>>X;
54     while(X != 0){
55         createNewElmt(X, P);
56         insertLast(L, P);
57         jumlah++;
58         cin>>X;
59     }
60     showList(L);
61     cout<<endl;
62
63     cout<<"Delete first:\n";
64     deleteFirst(L, P);
65     showList(L);
66     jumlah = jumlah - 1;
67     cout<<endl;
68
69     cout<<"Delete after:\n";
70     cout<<"Hapus: ";
71     Y = 0;
72     cin>>X;
73     P = cariElm(L, P, X);
74     cout<<"Setelah: ";
75     cin>>Y;
76     Prec = cariElm(L, Prec, Y);
77     deleteAfter(L, Prec, P);
78     showList(L);
79     jumlah = jumlah - 1;
80     cout<<endl;
81
82     cout<<"Delete last:\n";
83     deleteLast(L, P);
84     showList(L);
85     jumlah = jumlah - 1;
86     cout<<endl;
87
88     cout<<"Jumlah elemen: "<<jumlah<<endl;
89
90     i = median(L, jumlah);
91     cout<<"Median: "<<i<<endl;
92
93     cout<<"\nconcat:\n";
94     jumlah = 0;
95     cout<<"L1:\n";
96     cin>>X;
97     while(X != 0){
98         createNewElmt(X, P);
99         insertLast(L1, P);
100         jumlah++;
101         cin>>X;
102     }
103     showList(L1);
104     cout<<"L2:\n";
105     jumlah2 = 0;
106     cin>>X;
107     while(X != 0){
108         createNewElmt(X, P);
109         insertLast(L2, P);
110         jumlah2++;
111         cin>>X;
112     }
113     showList(L2);
114     cout<<endl;
115     concat(L1, L2, L3);
116     cout<<"L3:\n";
117     showList(L3);
118     cout<<"\nNAMA: FAHMI RAZAN RAMDANI\n";
119     cout<<"NIM: 1301194054\n";
120     return 0;
121 }
122
```

doublelinkedlist.cpp × main.cpp × doublelinkedlist.h ×

```
1  #include "doublelinkedlist.h"
2  /* NAMA: FAHMI RAZAN RAMDANI
3     NIM: 1301194054 */
4  using namespace std;
5
6  bool isEmpty(List L)
7  {
8      bool found;
9      found = false;
10     if(first(L) == NULL){
11         found = true;
12     }
13     return found;
14 }
15 void createList(List &L)
16 {
17     first(L) = NULL;
18     last(L) = NULL;
19 }
20 void createNewElmt(infotype X, address &P)
21 {
22     P = new elmList;
23     info(P) = X;
24     next(P) = NULL;
25     prev(P) = NULL;
26 }
27 void insertFirst(List &L, address P)
28 {
29     if(first(L) == NULL){
30         first(L) = P;
31         last(L) = P;
32     }else{
33         next(P) = first(L);
34         prev(first(L)) = P;
35         first(L) = P;
36     }
37 }
38 void insertAfter(List &L, address Prec, address P)
39 {
40     next(P) = next(Prec);
41     next(Prec) = P;
42     prev(P) = Prec;
43     prev(next(P)) = P;
44 }
45 address cariElm(List L, address &P, infotype X)
46 {
47     bool found;
48     found = false;
49     P = first(L);
50     while(P != NULL && !found){
51         if(info(P) == X){
52             return P;
53             found = true;
54         }
55         P = next(P);
56     }
57     if(info(P) == X){
58         return P;
59     }
60 }
61 void insertLast(List &L, address P)
62 {
63     address Q;
64     if(last(L) == NULL){
65         first(L) = P;
66         last(L) = P;
67     }else{
68         Q = last(L);
69         next(Q) = P;
70         prev(P) = Q;
71         last(L) = P;
72     }
73 }
74 void deleteFirst(List &L, address &P)
75 {
76     if(first(L) == last(L)){
77         P = first(L);
78         first(L) = NULL;
79         last(L) = NULL;
```

```
doublelinkedlist.cpp × main.cpp × doublelinkedlist.h ×
79     last(L) = NULL;
80 }else{
81     P = first(L);
82     first(L) = next(P);
83     prev(first(L)) = NULL;
84     next(P) = NULL;
85 }
86 }
87 void deleteAfter(List &L, address Prec, address &P)
88 {
89     P = next(Prec);
90     next(Prec) = next(P);
91     prev(next(P)) = Prec;
92     prev(P) = NULL;
93     next(P) = NULL;
94 }
95 void deleteLast(List &L, address &P)
96 {
97     address Q;
98     if(first(L) == last(L)){
99         P = first(L);
100        first(L) = NULL;
101        last(L) = NULL;
102    }else{
103        P = last(L);
104        Q = prev(last(L));
105        last(L) = Q;
106        next(Q) = NULL;
107        prev(P) = NULL;
108    }
109 }
110 void concat(List L1, List L2, List &L3)
111 {
112     L3 = L1;
113     next(last(L3)) = first(L2);
114     prev(first(L2)) = last(L3);
115 }
116 double median(List L, infotype jumlah)
117 {
118     address q = first(L);
119     address p;
120     int a;
121     float b;
122     a = 0;
123     b = 0.0;
124     if (first(L) != NULL) {
125         int jml = jumlah;
126         if (jml%2!=0) {
127             for (int i=1; i<=(jml/2); i++) {
128                 q = next(q);
129             }
130             return info(q);
131         } else{
132             while(a < (jml/2)){
133                 q = next(q);
134                 a++;
135             }
136             p = prev(q);
137             b = (info(p)+info(q));
138             return b/2;
139         }
140     } else {
141         return 0;
142     }
143 }
144 void showList(List L)
145 {
146     address P;
147     P = first(L);
148     cout<<"List: ";
149     while(P != NULL)
150     {
151         cout<<info(P)<<" ";
152         P = next(P);
153     }
154     cout<<endl;
155 }
156
157
```


cek list, list kosong: ya

Insert first:

3 1 0

List: 1 3

Insert after:

Input: 2

Setelah: 1

List: 1 2 3

Insert last:

4 5 6 7 0

List: 1 2 3 4 5 6 7

Delete first:

List: 2 3 4 5 6 7

Delete after:

Hapus: 5

Setelah: 4

List: 2 3 4 6 7

Delete last:

List: 2 3 4 6

Jumlah elemen: 4

Median: 3.5

concat:

L1:

1 2 3 0

List: 1 2 3

L2:

4 5 6 0

List: 4 5 6

L3:

List: 1 2 3 4 5 6

NAMA: FAHMI RAZAN RAMDANI

NIM: 1301194054

Process returned 0 (0x0) execution time : 28.933 s



dw34.me