

# LEMBAR KERJA PRAKTIKUM ALGORITMA DAN STRUKTUR DATA

Dosen Pengampu:

1. Nurul Fathanah Mustamin, S.Pd., M.T
2. Andry Fajar Zulkarnain, S.ST., M.T

## Modul 4. Double Link List

### Tujuan Praktikum:

1. Mahasiswa dapat menjelaskan mengenai Double Link List
2. Mahasiswa mampu melakukan analisis pada algoritma Double Link List yang dibuat
3. Mahasiswa mampu mengimplementasikan algoritma Double Link List pada sebuah aplikasi secara tepat dan efisien

Soal Praktikum:

1. Lengkapi coding pada function tambahDepanH() agar bisa berjalan dengan lancar. running, simpan program, dan screenshot hasil running !

```
1  #include <conio.h>
2  #include <iostream>
3  #include <stdlib.h>
4
5  using namespace std;
6
7  typedef struct TNode {
8      string data;
9      TNode *next;
10     TNode *prev;
11 };
12
13 TNode *head, *tail;
14
15 int pil, menu;
16 char pilihan[1];
17 string dataBaru;
18
19 void inith();
20 void inithT();
21 int isEmptyH();
22 int isEmptyHT();
23
24 void tambahDepanH();
25 void tambahDepanHT();
26 void tambahBelakangH();
27 void tambahBelakangHT();
28 void hapusDepanH();
29 void hapusDepanHT();
30 void hapusBelakangH();
31 void hapusBelakangHT();
32 void tampilkanH();
33 void tampilkanHT();
34 void clearH();
35 void clearHT();
36
37 int main()
38 {
39     menu:
40     cout<<"Double Linked List Non Circular (DLLNC)"<<endl;
41     cout<<"===== "<<endl;
42     cout<<"Silahkan pilih program DLLNC yang ingin dijalankan!"<<endl;
43     cout<<"1. DLLNC dengan Head"<<endl;
44     cout<<"2. DLLNC dengan Head dan Tail"<<endl;
45     cout<<"3. Quit"<<endl;
46     cout<<"Pilihan : ";
47     cin>>menu;
48     system("cls");
```

```

49     if(menu==1){
50         do {
51             cout<<"Double Linked List Non Circular (DLLNC) (Head)"<<endl;
52             cout<<"===== "<<endl;
53             cout<<"1. Tambah Depan"<<endl;
54             cout<<"2. Tambah Belakang"<<endl;
55             cout<<"3. Tampilkan Data"<<endl;
56             cout<<"4. Hapus Depan"<<endl;
57             cout<<"5. Hapus Belakang"<<endl;
58             cout<<"6. Reset"<<endl;
59             cout<<"7. Kembali ke Menu"<<endl;
60             cout<<"Pilihan : ";
61             cin>>pilihan;
62             pil=atoi(pilihan);

```

```

63
64         switch(pil) {
65             case 1:
66                 tambahDepanH();
67                 break;
68             case 2:
69                 tambahBelakangH();
70                 break;
71             case 3:
72                 tampilkanH();
73                 break;
74             case 4:
75                 hapusDepanH();
76                 break;
77             case 5:
78                 hapusBelakangH();
79                 break;
80             case 6:
81                 clearH();
82                 break;
83             default:
84                 system("cls");
85                 goto menu;
86         }
87
88         cout<<"\npress any key to continue"<<endl;
89         getch();
90         system("cls");
91
92     } while (pil<7);
93 } else if(menu==2){
94     do {

```

```

95     cout<<"Double Linked List Non Circular (DLLNC) (Head dan Tail)"<<endl;
96     cout<<"===== "<<endl;
97     cout<<"1. Tambah Depan"<<endl;
98     cout<<"2. Tambah Belakang"<<endl;
99     cout<<"3. Tampilkan Data"<<endl;
100    cout<<"4. Hapus Depan"<<endl;
101    cout<<"5. Hapus Belakang"<<endl;
102    cout<<"6. Reset"<<endl;
103    cout<<"7. Kembali ke Menu"<<endl;
104    cout<<"Pilihan : ";
105    cin>>pilihan;
106    pil=atoi(pilihan);
107
108    switch(pil) {
109    case 1:
110        tambahDepanHT();
111        break;
112    case 2:
113        tambahBelakangHT();
114        break;
115    case 3:
116        tampilkanHT();
117        break;
118    case 4:
119        hapusDepanHT();
120        break;
121    case 5:
122        hapusBelakangHT();
123        break;
124    case 6:
125        clearHT();
126        break;
127    default:
128        system("cls");
129        goto menu;
130    }
131
132    cout<<"\npress any key to continue"<<endl;
133    getch();
134    system("cls");
135
136    } while (pil<7);
137 } else {
138     cout<<"\nTERIMA KASIH"<<endl;
139     cout<<"Program was made by Nama (NIM). "<<endl;
140 }
141 }
142

```

```

143 void initH(){
144     head = NULL;
145 }
146
147 void initHT(){
148     head = NULL;
149     tail = NULL;
150 }
151
152 int isEmptyH(){
153     if(head == NULL) return 1;
154     else return 0;
155 }
156
157 int isEmptyHT(){
158     if(tail == NULL) return 1;
159     else return 0;
160 }
161
162 void tambahDepanH() {
163     cout<<"Masukkan data : ";
164
165
166
167
168
169
170
171
172
173
174
175
176
177     cout << "Data \""<<dataBaru<<"\" berhasil dimasukkan di bagian depan.";
178 }
179
180 void tambahDepanHT() {
181     cout<<"Masukkan data : ";
182     cin>>dataBaru;
183     TNode *baru;
184     baru = new TNode;
185     baru->data = dataBaru;
186     baru->next = NULL;
187     baru->prev = NULL;
188     if(isEmptyHT() == 1) {
189         head = baru;
190         tail = baru;

```

```

191     } else {
192         baru->next = head;
193         head->prev = baru;
194         head = baru;
195     }
196     cout << "Data \'"<<dataBaru<<\'\" berhasil dimasukkan di bagian depan.";
197 }
198
199 void tambahBelakangH() {
200     cout<<"Masukkan data : ";
201     cin>>dataBaru;
202     TNode *baru, *bantu;
203     baru = new TNode;
204     baru->data = dataBaru;
205     baru->next = NULL;
206     baru->prev = NULL;
207     if(isEmptyH() == 1) {
208         head = baru;
209     } else {
210         bantu = head;
211         while(bantu->next != NULL){
212             bantu = bantu->next;
213         }
214         bantu->next = baru;
215         baru->prev = bantu;
216     }
217     cout << "Data \'"<<dataBaru<<\'\" berhasil dimasukkan di bagian belakang.";
218 }
219

```

```

220 void tambahBelakangHT() {
221     cout<<"Masukkan data : ";
222     cin>>dataBaru;
223     TNode *baru;
224     baru = new TNode;
225     baru->data = dataBaru;
226     baru->next = NULL;
227     baru->prev = NULL;
228     if(isEmptyHT() == 1) {
229         head = baru;
230         tail = baru;
231     } else {
232         tail->next = baru;
233         baru->prev = tail;
234         tail = baru;
235     }
236     cout << "Data \'"<<dataBaru<<\'\" berhasil dimasukkan di bagian belakang.";
237 }

```

```

238
239 void tampilkanH() {
240     TNode *bantu;
241     bantu = head;
242     if(isEmptyH() == 0) {
243         while(bantu != NULL) {
244             cout<<bantu->data<<' ';
245             bantu = bantu->next;
246         }
247         cout<<endl;
248     } else cout<<"Tidak terdapat data pada Linked List";
249 }
250
251 void tampilkanHT() {
252     TNode *bantu;
253     bantu = head;
254     if(isEmptyHT() == 0) {
255         while(bantu != tail->next) {
256             cout<<bantu->data<<' ';
257             bantu = bantu->next;
258         }
259         cout<<endl;
260     } else cout<<"Tidak terdapat data pada Linked List";
261 }
262
263 void hapusDepanH() {
264     TNode *hapus;
265     string data;
266     if(isEmptyH() == 0) {
267         hapus = head;
268         data = hapus->data;
269         if(head->next != NULL) {
270             head = head->next;
271             head->prev = NULL;
272         } else {
273             inith();
274         }
275         delete hapus;
276         cout<<"Data \"<<data<<\" yang berada di depan telah berhasil dihapus.";
277     } else cout<<"Tidak terdapat data pada Linked List";
278 }
279
280 void hapusDepanHT() {
281     TNode *hapus;
282     string data;
283     if(isEmptyHT() == 0) {
284         hapus = head;
285         data = hapus->data;

```

```

286         if(head->next != NULL) {
287             head = head->next;
288             head->prev = NULL;
289         } else {
290             initHT();
291         }
292         delete hapus;
293         cout<<"Data \""<<data<<"\" yang berada di depan telah berhasil dihapus.";
294     } else cout<<"Tidak terdapat data pada Linked List";
295 }
296
297 void hapusBelakangH() {
298     TNode *hapus;
299     string data;
300     if(isEmptyH() == 0) {
301         hapus = head;
302         while(hapus->next != NULL){
303             hapus = hapus->next;
304         }
305         data = hapus->data;
306         if(head->next != NULL) {
307             hapus->prev->next = NULL;
308         } else {
309             initH();
310         }
311         delete hapus;
312         cout<<"Data \""<<data<<"\" yang berada di belakang telah berhasil dihapus.";
313     } else cout<<"Tidak terdapat data pada Linked List";
314 }
315
316 void hapusBelakangHT() {
317     TNode *hapus;
318     string data;
319     if(isEmptyHT() == 0) {
320         hapus = tail;
321         data = hapus->data;
322         if(head->next != NULL) {
323             tail = tail->prev;
324             tail->next = NULL;
325         } else {
326             initHT();
327         }
328         delete hapus;
329         cout<<"Data \""<<data<<"\" yang berada di belakang telah berhasil dihapus.";
330     } else cout<<"Tidak terdapat data pada Linked List";
331 }
332
333 void clearH() {

```

```

334     TNode *bantu, *hapus;
335     bantu = head;
336     while(bantu != NULL) {
337         hapus = bantu;
338         bantu = bantu->next;
339         delete hapus;
340     }
341     initH();
342     cout<<"Seluruh data pada Linked List telah dibersihkan.";
343 }
344
345 void clearHT() {
346     TNode *bantu, *hapus;
347     bantu = head;
348     while(bantu != NULL) {
349         hapus = bantu;
350         bantu = bantu->next;
351         delete hapus;
352     }
353     initHT();
354     cout<<"Seluruh data pada Linked List telah dibersihkan.";
355 }

```

2. Apa fungsi next pada coding?
3. Apa fungsi prev pada coding?
4. Gantilah baris 244 dan 256 dari `cout<<bantu->data<<' '`; menjadi `cout<<head->data<<' '`; lalu jawab pertanyaan berikut :
  - A. Apa yang terjadi jika anda menambahkan beberapa data pada program lalu tampilkan datanya, dan screenshoot hasilnya.
  - B. Jelaskan mengapa hal tersebut bisa terjadi dan data apa yang ditampilkan oleh program?