

PRAKTIKUM STRUKTUR DATA

TUGAS PENDAHULUAN 06

Double Linked List bagian 01



Nama :

Raihan Sastra Wibyanto (2311104020)

Dosen :

Yudha Islami Sulistya,S.Kom,M.Kom.

PROGRAM STUDI S1 REKAYASA PERANGKAT LUNAK

FAKULTAS INFORMATIKA

TELKOM UNIVERSITY PURWOKERTO

2024

SOAL TP

Soal 1: Menambahkan Elemen di Awal dan Akhir DLL

Contoh Program:

```

1  #include <iostream>
2  using namespace std;
3
4  class Node {
5  public:
6      int data;
7      Node* next;
8      Node* prev;
9
10     Node(int value) {
11         data = value;
12         next = nullptr;
13         prev = nullptr;
14     }
15 };
16
17 class DoubleLinkedList {
18 private:
19     Node* head;
20
21 public:
22     DoubleLinkedList() {
23         head = nullptr;
24     }
25
26     void insertFirst_2311104020(int value) {
27         Node* newNode = new Node(value);
28         if (head == nullptr) {
29             head = newNode;
30         } else {
31             newNode->next = head;
32             head->prev = newNode;
33             head = newNode;
34         }
35     }
36
37     void insertLast_2311104020(int value) {
38         Node* newNode = new Node(value);
39         if (head == nullptr) {
40             head = newNode;
41         } else {
42             Node* temp = head;
43             while (temp->next != nullptr) {
44                 temp = temp->next;
45             }
46             temp->next = newNode;
47             newNode->prev = temp;
48         }
49     }
50
51     void display_2311104020() {
52         Node* temp = head;
53         while (temp != nullptr) {
54             cout << temp->data;
55             if (temp->next != nullptr) cout << " <-> ";
56             temp = temp->next;
57         }
58         cout << endl;
59     }
60 };
61
62 int main() {
63     DoubleLinkedList dll;
64
65     dll.insertFirst_2311104020(10);
66     dll.insertFirst_2311104020(5);
67     dll.insertLast_2311104020(20);
68
69     cout << "DAFTAR ANGGOTA LIST: ";
70     dll.display_2311104020();
71
72     return 0;
73 }
74

```

Output:

```
Raihan Sastra W@MSI MINGW64 ~/Documents/Institut Teknologi Telkom Purwokerto/Semester 3/Struktur Data/STD_Raihan_Sastra_Wibyanto_2311104020/06_Double_Linked_List_Bagian1/TP/output (main)
$ ./"DLL.exe"
DAFTAR ANGGOTA LIST: 5 <-> 10 <-> 20
```

Soal 2: Menghapus Elemen di Awal dan Akhir DLL

Contoh Program:

```

1 #include <iostream>
2 using namespace std;
3
4 class Node {
5 public:
6     int data;
7     Node* next;
8     Node* prev;
9
10    Node(int value) {
11        data = value;
12        next = nullptr;
13        prev = nullptr;
14    }
15 };
16
17 class DoubleLinkedList {
18 private:
19     Node* head;
20
21 public:
22     DoubleLinkedList() {
23         head = nullptr;
24     }
25
26     void insertFirst_2311104020(int value) {
27         Node* newNode = new Node(value);
28         if (head == nullptr) {
29             head = newNode;
30         } else {
31             newNode->next = head;
32             head->prev = newNode;
33             head = newNode;
34         }
35     }
36
37     void insertLast_2311104020(int value) {
38         Node* newNode = new Node(value);
39         if (head == nullptr) {
40             head = newNode;
41         } else {
42             Node* temp = head;
43             while (temp->next != nullptr) {
44                 temp = temp->next;
45             }
46             temp->next = newNode;
47             newNode->prev = temp;
48         }
49     }
50
51     void deleteFirst_2311104020() {
52         if (head == nullptr) {
53             cout << "List is empty, nothing to delete." << endl;
54             return;
55         }
56         Node* temp = head;
57         if (head->next == nullptr) {
58             head = nullptr;
59         } else {
60             head = head->next;
61             head->prev = nullptr;
62         }
63         delete temp;
64     }
65
66     void deleteLast_2311104020() {
67         if (head == nullptr) {
68             cout << "List is empty, nothing to delete." << endl;
69             return;
70         }
71         Node* temp = head;
72         if (head->next == nullptr) {
73             head = nullptr;
74         } else {
75             while (temp->next != nullptr) {
76                 temp = temp->next;
77             }
78             temp->prev->next = nullptr;
79         }
80         delete temp;
81     }
82
83     void display_2311104020() {
84         Node* temp = head;
85         if (temp == nullptr) {
86             cout << "List is empty" << endl;
87             return;
88         }
89         while (temp != nullptr) {
90             cout << temp->data;
91             if (temp->next != nullptr) cout << " <-> ";
92             temp = temp->next;
93         }
94         cout << endl;
95     }
96 };
97
98 int main() {
99     DoubleLinkedList dll;
100
101     dll.insertFirst_2311104020(15);
102     dll.insertFirst_2311104020(10);
103     dll.insertLast_2311104020(20);
104
105     cout << "DAFTAR ANGGOTA LIST: ";
106     dll.display_2311104020();
107
108     dll.deleteFirst_2311104020();
109     dll.deleteLast_2311104020();
110
111     cout << "DAFTAR ANGGOTA LIST SETELAH PENGHAPUSAN: ";
112     dll.display_2311104020();
113
114     return 0;
115 }

```

Output:

```
Raihan Sastra W@MSI MINGW64 ~/Documents/Institut Teknologi Telkom Purwokerto/Semester 3/Struktur Data/STD_Raihan_Sastra_Wibyanto_2311104020/06_Double_Linked_List_Bagian1/TP/output (main)
$ ./"DLL2.exe"
DAFTAR ANGGOTA LIST: 10 <-> 15 <-> 20
DAFTAR ANGGOTA LIST SETELAH PENGHAPUSAN: 15
```

Soal 3: Menampilkan Elemen dari Depan ke Belakang dan sebaliknya

Contoh Program:

```

1  #include <iostream>
2  using namespace std;
3
4  class Node {
5  public:
6      int data;
7      Node* next;
8      Node* prev;
9
10     Node(int value) {
11         data = value;
12         next = nullptr;
13         prev = nullptr;
14     }
15 };
16
17 class DoubleLinkedList {
18 private:
19     Node* head;
20     Node* tail;
21
22 public:
23     DoubleLinkedList() {
24         head = nullptr;
25         tail = nullptr;
26     }
27
28     void insertLast_2311104020(int value) {
29         Node* newNode = new Node(value);
30         if (head == nullptr) {
31             head = tail = newNode;
32         } else {
33             tail->next = newNode;
34             newNode->prev = tail;
35             tail = newNode;
36         }
37     }
38
39     void displayForward_2311104020() {
40         Node* temp = head;
41         while (temp != nullptr) {
42             cout << temp->data;
43             if (temp->next != nullptr) cout << " <-> ";
44             temp = temp->next;
45         }
46         cout << endl;
47     }
48
49     void displayBackward_2311104020() {
50         Node* temp = tail;
51         while (temp != nullptr) {
52             cout << temp->data;
53             if (temp->prev != nullptr) cout << " <-> ";
54             temp = temp->prev;
55         }
56         cout << endl;
57     }
58 };
59
60 int main() {
61     DoubleLinkedList dll;
62
63     dll.insertLast_2311104020(1);
64     dll.insertLast_2311104020(2);
65     dll.insertLast_2311104020(3);
66     dll.insertLast_2311104020(4);
67
68     cout << "Daftar elemen dari depan ke belakang: ";
69     dll.displayForward_2311104020();
70
71     cout << "Daftar elemen dari belakang ke depan: ";
72     dll.displayBackward_2311104020();
73
74     return 0;
75 }
76

```

Output:

```
Raihan Sastra W@MSI MINGW64 ~/Documents/Institut Teknologi Telkom Purwokerto/Semester 3/Struktur Data/STD_Raihan_Sastra_Wibyanto_2311104020/06_Double_Linked_List_Bagian1/TP/output (main)
$ ./"DLL3.exe"
Daftar elemen dari depan ke belakang: 1 <-> 2 <-> 3 <-> 4
Daftar elemen dari belakang ke depan: 4 <-> 3 <-> 2 <-> 1
```