

**Laporan Praktikum**

**Algoritma dan Struktur Data**

Ganjil 2023/2024

Program Studi Teknik Informatika

Institut Teknologi Sumatera

|  |  |
| --- | --- |
| **Modul :** | **Double Linked List** |
| **Kelompok:** | **Muhammad Yusuf(122140193)**  **Muhammad Rizqullah Bimo Primatama(122140220)**  **Kevin Naufal Dany(122140222)**  **Ihya Razky Hidayat(122140167)** |
|  |  |
| **Kelas:** | **ASD RA** |

Instruksi sederhana :

* Disarankan kepada **Praktikan Algoritma Struktur Data** untuk mengeditnya menggunakan Google Docs agar tidak berantakan dan rapi,
* Silahkan mengganti **Nama Modul** baik yang ada pada **Cover** dan **Header** sesuai dengan materi praktikum,
* Gunakan text styling seperti **Heading 1**, **Normal Text** yang telah terformat / Text Style lainnya yang digunakan untuk menjaga estetika laporan,
* Gunakan **[Syntax Highlighter](https://highlight.hohli.com/index.php)** untuk merapikan kode yang sudah Praktikan buat ke dalam Laporan Praktikum.

**Materi Praktikum**

Double Linked List

**Link repl.it Source Code**

<https://replit.com/@muhammad1221401/Muhammad-Yusuf-Prak-ASD-RA122140193#minggu-2/latihan.cpp>

**Source Code**

Gunakan [**Syntax Highlighter**](https://highlight.hohli.com/index.php)untuk merapikan Source Code yang dipindahkan dari text editor anda ke dokumen ini.

1. // Muhammad Yusuf
2. // 122140193
4. #include <iostream>
5. using namespace std;
7. struct Node {
8. **int value;**
9. Node\* next;
10. Node\* prev;
11. };
13. **struct List {**
14. Node\* head=nullptr;
15. Node\* tail=nullptr;
16. };
18. **Node \*createNode(int value);**
19. void insertSorted(List& list, int value);
20. void printDoubleLinkedList(Node\* head);
22. int main() {
23. **List;**
24. int banyakData, inputFromUser;
26. cout << "Tentukan banyak data yang ingin diinputkan (Angka): ";
27. cin >> banyakData;
29. for (int i = 0; i < banyakData; i++) {
30. cout << "Masukkan angka ke-" << i + 1 << ": ";
31. cin >> inputFromUser;
33. **insertSorted(list, inputFromUser);**
34. }
36. printDoubleLinkedList(list.head);
38. **// Delete all nodes to prevent memory leak**
39. while (list.head != nullptr) {
40. Node\* temp = list.head;
41. list.head = list.head->next;
42. delete temp;
43. **}**
45. return 0;
46. }
48. **Node\* createNode(int value) {**
49. Node\* newNode = new Node;
51. newNode->value = value;
52. newNode->next = nullptr;
53. **newNode->prev = nullptr;**
55. return newNode;
56. }
58. **void insertSorted(List& list, int value) {**
59. Node\* newNode = createNode(value);
61. if (!list.head) {
62. list.head = newNode;
63. **list.tail = newNode;**
64. }else if (value <= list.head->value) {
65. newNode->next = list.head;
66. list.head->prev = newNode;
67. list.head = newNode;
68. **}else if (value >= list.tail->value) {**
69. newNode->prev = list.tail;
70. list.tail->next = newNode;
71. list.tail = newNode;
72. }else {
73. **Node\* current = list.head;**
75. while (current->next && current->next->value < value) {
76. current = current->next;
77. }
79. newNode->next = current->next;
80. current->next = newNode;
81. newNode->prev = current;
82. }
83. **}**
85. void printDoubleLinkedList(Node\* head) {
86. Node\* current = head;
87. while (current) {
88. **cout << "[" << current->prev << " " << current->value << " " << current->next << "] ";**
89. current = current->next;
90. }
91. cout << endl;
92. }

**Dokumentasi Hasil Running**

|  |
| --- |
|  |
| **Gambar 1.** Output Code Double Linked List Latihan Minggu 2.  Pada gambar tersbut kita diminta untuk menginputlan 5 angka, yang nantinya tiap angka akan langsung diurutkan menggunakan fungsi insertSorted, lalu code akan mencetak semua double linked list menggunakan fungsi printDoubleLinkedList dan akan menampilkan value, pointer next, dan prev secara terurut. |

|  |
| --- |
|  |
|  |