

**Laporan Praktikum**

**Algoritma dan Struktur Data**

Ganjil 2023/2024

Program Studi Teknik Informatika

Institut Teknologi Sumatera

|  |  |
| --- | --- |
| **Modul :** | **Linked List** |
| **Nama :** | **Muhammad Yusuf** |
| **NIM :** | **122140193** |
| **Kelas (Kelas Asal) :** | **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**

Linked List

**Link repl.it Source Code**

<https://replit.com/@muhammad1221401/Muhammad-Yusuf-Prak-ASD-RA122140193#minggu-1/class/tugas1.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. class Node {
8. **public:**
9. int value;
10. Node\* next;
12. Node(int val) : value(val), next(nullptr) {}
13. **};**
15. class LinkedList {
16. public:
17. Node\* head;
19. LinkedList() : head(nullptr) {}
21. void insertSorted(int value) {
22. Node\* newNode = new Node(value);
24. if (head == nullptr || value <= head->value) {
25. newNode->next = head;
26. head = newNode;
27. return;
28. **}**
30. Node\* current = head;
32. while (current->next != nullptr && current->next->value < value) {
33. **current = current->next;**
34. }
36. newNode->next = current->next;
37. current->next = newNode;
38. **}**
40. void printLinkedList() {
41. Node\* current = head;
43. **while (current != nullptr) {**
44. cout << " [" << current->value << " | (" << current->next << ")] ";
46. current = current->next;
47. }
48. **cout << endl;**
49. }
51. void deleteNode(int n) {
52. Node\* current = head;
53. **Node\* prev = nullptr;**
55. while (current != nullptr && current->value != n) {
56. prev = current;
57. current = current->next;
58. **}**
60. if (current == nullptr) {
61. cout << "Maaf, angka yang Anda inputkan tidak ada dalam linked list." << endl;
62. return;
63. **}**
65. if (prev == nullptr) {
66. head = current->next;
67. }
68. **else {**
69. prev->next = current->next;
70. }
72. cout << "Angka " << n << " telah dihapus dari linked list." << endl;
73. **delete current;**
74. }
76. ~LinkedList() {
77. while (head != nullptr) {
78. **Node\* temp = head;**
79. head = head->next;
80. delete temp;
81. }
82. }
83. **};**
85. int main() {
86. LinkedList list;
87. int inputFromUser;
89. cout << "Masukkan 5 angka untuk diurutkan: " << endl;
91. for (int i = 0; i < 5; i++) {
92. cin >> inputFromUser;
93. **list.insertSorted(inputFromUser);**
94. }
96. list.printLinkedList();
98. **cout << "Input angka yang ingin dihapus: " << endl;**
99. cin >> inputFromUser;
101. list.deleteNode(inputFromUser);
103. **cout << "Linked list setelah dihapus: " << endl;**
104. list.printLinkedList();
106. return 0;
107. }

**Dokumentasi Hasil Running**

|  |
| --- |
|  |
| **Gambar 1.** Output Code LinkedList Tugas 1 Minggu 1.  Pada gambar tersbut kita diminta untuk menginputlan 5 angka, yang nantinya tiap angka akan langsung diurutkan menggunakan fungsi insertSorted,lalu code akan mencetak semua linkedlist menggunakan fungsi printLinkedList dan akan menampilkan value serta pointer nextnya secara terurut.  Setelah itu user diminta menginputkan angka yang ingin dihapus lalu akan dijalankan fungsi deleteNode dan passing parameternya. Setelah itu akan printlist lagi. |

|  |
| --- |
|  |
|  |