

**Laporan Praktikum**

**Algoritma dan Struktur Data**

Ganjil 2023/2024

Program Studi Teknik Informatika

Institut Teknologi Sumatera

|  |  |
| --- | --- |
| **Modul :** | **Circular LinkedList** |
| **Nama :** | **Chikalyz Kayla Putri Maharani Mae** |
| **NIM :** | **122140161** |
| **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**

Tugas Circular Linked List

**Link repl.it Source Code**

<https://replit.com/@chikalyz1221401/Tugas-Minggu-3?v=1>

**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. #include <iostream>
2. using namespace std;
4. struct Node {
5. int value;
6. Node\* next;
7. };
9. Node\* head = nullptr;
11. void insertFirst(int value) {
12. Node\* newNode = new Node;
13. newNode->value = value;
15. if (head == nullptr) {
16. head = newNode;
17. newNode->next = head;
18. } else {
19. Node\* temp = head;
21. while (temp->next != head) {
22. temp = temp->next;
23. }
25. temp->next = newNode;
26. newNode->next = head;
27. head = newNode;
28. }
29. }
31. void insertLast(int value) {
32. Node\* newNode = new Node;
33. newNode->value = value;
35. if (head == nullptr) {
36. head = newNode;
37. newNode->next = head;
38. } else {
39. Node\* temp = head;
41. while (temp->next != head) {
42. temp = temp->next;
43. }
45. temp->next = newNode;
46. newNode->next = head;
47. }
48. }
50. void deleteLast() {
51. if (head == nullptr) {
52. cout << "Linked List kosong." << endl;
53. return;
54. }
56. Node\* temp = head;
58. while (temp->next->next != head) {
59. temp = temp->next;
60. }
62. Node\* toDelete = temp->next;
63. temp->next = head;
64. delete toDelete;
65. }
67. void deleteFirst() {
68. if (head == nullptr) {
69. cout << "Linked List kosong." << endl;
70. return;
71. }
73. Node\* temp = head;
75. while (temp->next != head) {
76. temp = temp->next;
77. }
79. Node\* toDelete = head;
80. temp->next = head->next;
81. head = head->next;
82. delete toDelete;
83. }
85. void displayLinkedList() {
86. if (head == nullptr) {
87. cout << "Linked List kosong." << endl;
88. return;
89. }
91. cout << "**\n**Isi circular linked list saat ini:" << endl << endl;
93. Node\* current = head;
95. do {
96. cout << "(" << current->value << ", " << current->next << ") ";
98. current = current->next;
99. } while (current != head);
101. cout << endl;
102. }
104. int main() {
105. int n, uk\_cll;
107. cout << "Tentukan panjang circular linked list yang anda mau: ";
108. cin >> uk\_cll;
110. for (int i = 0; i < uk\_cll; i++) {
111. cout << "Masukkan angka: ";
112. cin >> n;
114. if (n % 2 == 0) {
115. insertFirst(n);
116. } else {
117. insertLast(n);
118. }
119. }
121. displayLinkedList();
123. int choice;
125. do {
126. cout << "**\n**Pilih operasi yang ingin anda lakukan: " << endl;
127. cout << "1. Hapus data di awal" << endl;
128. cout << "2. Hapus data di akhir" << endl;
129. cout << "3. Keluar Program" << endl;
131. cout << "Masukkan pilihan anda: ";
132. cin >> choice;
134. switch (choice) {
135. case 1:
136. deleteFirst();
137. break;
138. case 2:
139. deleteLast();
140. break;
141. case 3:
142. cout << "Terima kasih" << endl;
143. break;
144. default:
145. cout << "Pilihan anda tidak valid" << endl;
146. break;
147. }
149. if (choice != 3) {
150. displayLinkedList();
151. }
152. } while (choice != 3);
154. return 0;
155. }

**Dokumentasi Hasil Running**

|  |
| --- |
|  |
| **Gambar 1.** Output Code LinkedList Tugas 1 Minggu 3.  Penjelasan singkat mengenai kode tersebut:  1. Struktur `Node` digunakan untuk mewakili simpul (node) dalam Circular Linked List. Setiap simpul memiliki dua atribut: `value` (nilai yang disimpan dalam simpul) dan `next` (pointer ke simpul berikutnya dalam linked list).  2. Kode dimulai dengan inisialisasi `head` (kepala) linked list ke `nullptr`, menandakan bahwa awalnya linked list kosong.  3. Fungsi `insertFirst` digunakan untuk menyisipkan simpul dengan nilai tertentu di awal linked list. Jika linked list masih kosong, maka simpul baru akan menjadi satu-satunya simpul dalam linked list. Jika tidak, simpul baru akan ditempatkan di depan, dan pointer terakhir akan diperbarui untuk menunjuk ke simpul baru.  4. Fungsi `insertLast` melakukan hal yang serupa dengan `insertFirst`, tetapi menyisipkan simpul baru di akhir linked list.  5. Fungsi `deleteFirst` menghapus simpul pertama dari linked list. Jika linked list kosong, pesan kesalahan akan ditampilkan. Jika tidak, kepala linked list akan diperbarui untuk menunjuk ke simpul kedua, dan simpul pertama akan dihapus.  6. Fungsi `deleteLast` menghapus simpul terakhir dari linked list. Jika linked list kosong, pesan kesalahan akan ditampilkan. Jika tidak, fungsi akan mencari simpul kedua terakhir dalam linked list dan mengubah pointer "next" pada simpul tersebut sehingga menunjuk kembali ke kepala linked list, sehingga simpul terakhir akan dihapus.  7. Fungsi `displayLinkedList` digunakan untuk mencetak isi dari linked list. Ini melakukan perulangan dari kepala linked list dan mencetak nilai dan pointer "next" setiap simpul.  8. Dalam `main`, program meminta pengguna untuk menentukan panjang circular linked list dan menginputkan angka. Angka-angka tersebut akan dimasukkan ke dalam linked list sesuai dengan aturan, yaitu angka genap dimasukkan di awal dan angka ganjil dimasukkan di akhir. Setelah itu, program akan menampilkan isi linked list.  9. Selanjutnya, pengguna diminta untuk memilih operasi yang ingin dilakukan: menghapus data di awal, menghapus data di akhir, atau keluar dari program. Sesuai dengan pilihan pengguna, program akan menjalankan operasi tersebut dan menampilkan isi linked list setelah operasi dilakukan. |

|  |
| --- |
|  |
|  |