

**Laporan Praktikum**

**Algoritma dan Struktur Data**

Ganjil 2023/2024

Program Studi Teknik Informatika

Institut Teknologi Sumatera

|  |  |
| --- | --- |
| **Modul :** | **Stack Queue** |
| **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**

Stack Queue

**Link repl.it Source Code**

<https://replit.com/@muhammad1221401/Muhammad-Yusuf-Prak-ASD-RA122140193#minggu-4/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. const int MAX\_SIZE = 5; // Maksimum ukuran stack
9. struct Stack {
10. int stackArray[MAX\_SIZE];
11. int top;
13. **// Create an empty stack**
14. Stack() {
15. top = -1;
16. }
18. **bool isEmpty() {**
19. return top == -1;
20. }
22. bool isFull() {
23. **return top == MAX\_SIZE - 1;**
24. }
26. void push(int value) {
27. if (isFull()) {
28. **cout << "Stack penuh. Tidak dapat menambahkan elemen lagi." << endl;**
29. return;
30. }
32. stackArray[++top] = value;
33. **}**
35. void pop() {
36. if (isEmpty()) {
37. cout << "Stack kosong. Tidak dapat menghapus elemen." << endl;
38. **return;**
39. }
41. --top;
42. }
44. void printStack() {
45. if (isEmpty()) {
46. cout << "Stack kosong." << endl;
47. return;
48. **}**
50. cout << "Isi stack:" << endl;
52. for (int i = top; i >= 0; i--) {
53. **cout << stackArray[i] << " -> index ke- " << i << endl;**
54. }
56. cout << endl;
57. }
59. int peek() {
60. if (isEmpty()) {
61. cout << "Stack kosong." << endl;
62. return -1; // Nilai default jika stack kosong
63. **}**
65. return stackArray[top];
66. }
68. **int countStack() {**
69. return top + 1;
70. }
71. };
73. **int main() {**
74. int size, numFromUser;
76. Stack;
78. **cout << "Tentukan berapa banyak angka yang ingin Anda masukkan ke stack: ";**
79. cin >> size;
81. if (size > MAX\_SIZE) {
82. cout << "Banyak angka melebihi kapasitas maksimum." << endl;
83. **return 1;**
84. }
86. for (int i = 0; i < size; i++) {
87. cout << "Masukkan angka ke-" << i + 1 << ": ";
88. **cin >> numFromUser;**
90. stack.push(numFromUser);
91. }
93. **cout << "Elemen teratas: " << stack.peek() << endl;**
95. stack.printStack();
97. cout << "Ukuran stack: " << stack.countStack() << endl;
99. return 0;
100. }

**Dokumentasi Hasil Running**

|  |
| --- |
|  |
| **Gambar 1.** Output Code Stack Latihan Minggu 4.  Code di atas adalah implementasi dari struktur data stack dalam bahasa pemrograman C++. Stack diwakili oleh struktur Stack yang memiliki array stackArray sebagai penyimpanan elemen-elemen stack. Struktur ini memiliki beberapa fungsi, seperti isEmpty dan isFull untuk memeriksa apakah stack kosong atau penuh, push untuk menambahkan elemen ke stack, pop untuk menghapus elemen dari stack, printStack untuk menampilkan isi stack, peek untuk melihat elemen teratas tanpa menghapusnya, dan countStack untuk menghitung jumlah elemen dalam stack. Program utama menggunakan objek dari struktur Stack untuk menginteraksi dengan stack, meminta pengguna untuk memasukkan angka ke dalam stack, dan kemudian menampilkan informasi seperti elemen teratas, isi stack, dan ukuran stack. |

|  |
| --- |
|  |
|  |