

**Laporan Praktikum
MODUL 7 (STACK)**



Disusun Oleh:

Dwi Candra Pratama/2211104035

Asisten Praktikum:

Dosen Pengampu

Wahyu Andi Saputra

**Program Studi S1 SOFTWARE ENGINEERING
Fakultas Informatika Telkom University Purwokerto
2024**

A. TUJUAN PRAKTIKUM

- Mampu memahami konsep stack pada struktur data dan algoritma
- Mampu mengimplementasikan operasi-operasi pada stack
- Mampu memecahkan permasalahan dengan solusi stack

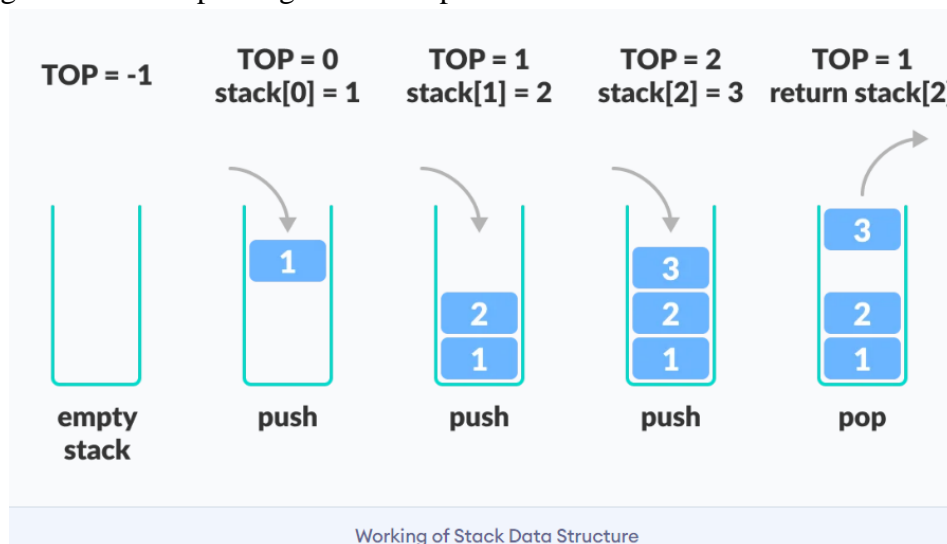
B. DASAR TEORI

Sebuah stack atau tumpukan merupakan struktur data yang berfungsi untuk menyimpan dan mengelola kumpulan data dengan prinsip Last-In, First-Out (LIFO). Analogi yang sering digunakan adalah tumpukan piring di kafetaria, di mana piring terakhir yang ditambahkan akan menjadi yang pertama diambil.

Dalam implementasinya, stack dapat direpresentasikan sebagai struktur data terurut yang memiliki dua operasi utama: push dan pop. Operasi push digunakan untuk menambahkan elemen baru ke dalam stack, sementara operasi pop digunakan untuk menghapus elemen teratas dari stack.

Prinsip LIFO yang menjadi dasar stack membuatnya sangat bermanfaat dalam berbagai aplikasi, termasuk manajemen memori komputer, evaluasi ekspresi aritmatika, dan manajemen panggilan fungsi dalam pemrograman. Sebagai contoh, dalam manajemen memori, stack digunakan untuk menyimpan alamat-alamat memori yang dialokasikan untuk variabel dan fungsi.

Dengan prinsip LIFO ini, stack memungkinkan akses data dengan efisiensi, di mana elemen yang terakhir dimasukkan akan menjadi yang pertama diambil. Hal ini menjadikannya salah satu struktur data yang sangat penting dalam pengembangan perangkat lunak dan pemrograman komputer..



Operasi pada stack melibatkan beberapa fungsi dasar yang dapat dilakukan pada struktur data ini. Berikut adalah beberapa operasi umum pada stack:

- Push (Masukkan):** Menambahkan elemen ke dalam tumpukan pada posisi paling atas atau ujung.
- Pop (Keluarkan):** Menghapus elemen dari posisi paling atas atau ujung tumpukan.

- c. Top (Atas): Mendapatkan nilai atau melihat elemen teratas pada tumpukan tanpa menghapusnya.
- d. IsEmpty (Kosong): Memeriksa apakah tumpukan kosong atau tidak.
- e. IsFull (Penuh): Memeriksa apakah tumpukan penuh atau tidak (terutama pada implementasi tumpukan dengan kapasitas terbatas).
- f. Size (Ukuran): Mengembalikan jumlah elemen yang ada dalam tumpukan.
- g. Peek (Lihat): Melihat nilai atau elemen pada posisi tertentu dalam tumpukan tanpa menghapusnya.
- h. Clear (Hapus Semua): Mengosongkan atau menghapus semua elemen dari tumpukan.
- i. Search (Cari): Mencari keberadaan elemen tertentu dalam tumpukan.

C. GUIDED 1

Sourcecode

```
#include <iostream>
#define MAX 100

using namespace std;

class Stack {
private:
    int top;
    int arr[MAX];

public:
    Stack() { top = -1; }

    bool isFull() { return top == MAX - 1; }
    bool isEmpty() { return top == -1; }

    void push(int x) {
        if (isFull()) {
            cout << "Stack Overflow\n";
            return;
        }
        arr[++top] = x;
    }

    void pop() {
        if (isEmpty()) {
            cout << "Stack Underflow\n";
            return;
        }
        top--;
    }
}
```

```

int peek() {
    if (!isEmpty()) {
        return arr[top];
    }
    cout << "Stack is empty\n";
    return -1; // Return a sentinel value
}

void display() {
    if (isEmpty()) {
        cout << "Stack is empty\n";
        return;
    }
    for (int i = top; i >= 0; i--) {
        cout << arr[i] << " ";
    }
    cout << "\n";
}
};

int main() {
    Stack s;
    s.push(10);
    s.push(20);
    s.push(30);

    cout << "Stack elements: ";
    s.display();

    cout << "Top element: " << s.peek() << "\n";

    s.pop();
    s.pop();
    cout << "After popping, stack elements: ";
    s.display();

    return 0;
}

```

```

PS D:\Semester5\StrukturData\PraktikumStrukturData\Pertemuan 7\GUIDED\output> & .\GUIDED7.exe'
Stack elements: 30 20 10
Top element: 30
After popping, stack elements: 10
PS D:\Semester5\StrukturData\PraktikumStrukturData\Pertemuan 7\GUIDED\output> █

```

GUIDED 2

```
#include <iostream>
using namespace std;

class Node {
public:
    int data;
    Node* next;

    Node(int value) {
        data = value;
        next = nullptr;
    }
};

class Stack {
private:
    Node* top;

public:
    Stack() { top = nullptr; }

    bool isEmpty() { return top == nullptr; }

    void push(int x) {
        Node* newNode = new Node(x);
        newNode->next = top;
        top = newNode;
    }

    void pop() {
        if (isEmpty()) {
            cout << "Stack Underflow\n";
            return;
        }
        Node* temp = top;
        top = top->next;
        delete temp;
    }

    int peek() {
        if (!isEmpty()) {
            return top->data;
        }
        cout << "Stack is empty\n";
        return -1; // Return a sentinel value
    }
}
```

```

void display() {
    if (isEmpty()) {
        cout << "Stack is empty\n";
        return;
    }
    Node* current = top;
    while (current) {
        cout << current->data << " ";
        current = current->next;
    }
    cout << "\n";
}

};

int main() {
    Stack s;
    s.push(10);
    s.push(20);
    s.push(30);

    cout << "Stack elements: ";
    s.display();

    cout << "Top element: " << s.peek() << "\n";

    s.pop();
    cout << "After popping, stack elements: ";
    s.display();

    return 0;
}

```

```

PS D:\Semester5\StrukturData\PraktikumStrukturData\Pertemuan 7> cd 'd:\Semester5\StrukturData\PraktikumStrukturData\Pertemuan 7\GUIDED\output' & .\Pertemuan7.exe
Stack elements: 30 20 10
Top element: 30
Stack after pop: 20 10
PS D:\Semester5\StrukturData\PraktikumStrukturData\Pertemuan 7\GUIDED\output>

```

D. UNGUIDED

1. Buatlah program untuk menentukan apakah kalimat tersebut yang diinputkan dalam program stack adalah palindrom/tidak. Palindrom kalimat yang dibaca dari depan dan belakang sama. Jelaskan bagaimana cara kerja programnya.

Contoh:

Kalimat : ini

Kalimat tersebut adalah polindrom

Kalimat : telkom

Kalimat tersebut adalah bukan polindrom

```
Masukan Kalimat : ini
Kalimat tersebut adalah : Palindrom
```

Source Codenya:

```
// Program untuk mengecek apakah sebuah kalimat adalah palindrom
menggunakan C++
#include <iostream>
#include <stack>
#include <algorithm> // untuk transform

// Fungsi untuk mengecek apakah kalimat adalah palindrom
bool is_palindrome(std::string kalimat) {
    std::stack<char> stack;

    // Menghilangkan spasi dan mengubah ke huruf kecil
    kalimat.erase(remove(kalimat.begin(), kalimat.end(), ' '), kalimat.end());
    std::transform(kalimat.begin(), kalimat.end(), kalimat.begin(), ::tolower);

    // Menyimpan karakter ke dalam stack
    for (char c : kalimat) {
        stack.push(c);
    }

    // Membandingkan karakter satu per satu
    for (char c : kalimat) {
        if (c != stack.top()) {
            return false;
        }
        stack.pop();
    }
    return true;
}

int main() {
    std::string kalimat;
    std::cout << "Masukkan Kalimat: ";
    std::getline(std::cin, kalimat);

    if (is_palindrome(kalimat)) {
```

```

        std::cout << "Kalimat tersebut adalah palindrom" << std::endl;
    } else {
        std::cout << "Kalimat tersebut bukan palindrom" << std::endl;
    }

    return 0;
}

```

```

PS D:\Semester5\StrukturData\PraktikumStrukturData\Pertemuan 7\UNGUIDED\output> cd 'd:\Semester5\StrukturData\PraktikumStrukturData\Pertemuan 7\UNGUIDED\output'
PS D:\Semester5\StrukturData\PraktikumStrukturData\Pertemuan 7\UNGUIDED\output> & .\unguided1.exe
Masukkan Kalimat: ini
Kalimat tersebut adalah palindrom
PS D:\Semester5\StrukturData\PraktikumStrukturData\Pertemuan 7\UNGUIDED\output> cd 'd:\Semester5\StrukturData\PraktikumStrukturData\Pertemuan 7\UNGUIDED\output'
PS D:\Semester5\StrukturData\PraktikumStrukturData\Pertemuan 7\UNGUIDED\output> & .\unguided1.exe
Masukkan Kalimat: telkom
Kalimat tersebut bukan palindrom
PS D:\Semester5\StrukturData\PraktikumStrukturData\Pertemuan 7\UNGUIDED\output>

```

2. Buatlah program untuk melakukan pembalikan terhadap kalimat menggunakan stack dengan minimal 3 kata. Jelaskan output program dan source codenya beserta operasi/fungsi yang dibuat?

Contoh

Kalimat : Telkom Purwokerto

Hasil : otrekowruP mokleT

```

Masukkan Kata Telkom Purwokerto
Datastack Array :
Data : otrekowruP mokleT

```

Source Code:

```

// Program untuk mengecek apakah sebuah kalimat adalah palindrom
menggunakan C++
#include <iostream>
#include <stack>
#include <algorithm> // untuk transform

// Fungsi untuk mengecek apakah kalimat adalah palindrom
bool is_palindrome(std::string kalimat) {
    std::stack<char> stack;

    // Menghilangkan spasi dan mengubah ke huruf kecil
    kalimat.erase(remove(kalimat.begin(), kalimat.end(), ' '), kalimat.end());
    std::transform(kalimat.begin(), kalimat.end(), kalimat.begin(), ::tolower);

    // Menyimpan karakter ke dalam stack

```



```

    for (char c : kalimat) {
        stack.push(c);
    }

    // Membandingkan karakter satu per satu
    for (char c : kalimat) {
        if (c != stack.top()) {
            return false;
        }
        stack.pop();
    }
    return true;
}

// Fungsi untuk membalikkan kalimat menggunakan stack
std::string reverse_sentence(std::string sentence) {
    std::stack<char> stack;
    std::string reversed_sentence;

    // Menyimpan setiap karakter dalam stack
    for (char c : sentence) {
        stack.push(c);
    }

    // Mengeluarkan karakter satu per satu dari stack untuk membalikkan kalimat
    while (!stack.empty()) {
        reversed_sentence += stack.top();
        stack.pop();
    }

    return reversed_sentence;
}

int main() {
    std::string kalimat;
    std::cout << "Masukkan Kata: ";
    std::getline(std::cin, kalimat);

    std::cout << "Datastack Array: " << std::endl;
    std::cout << "Data: " << reverse_sentence(kalimat) << std::endl;

    return 0;
}

```

```
PS D:\Semester5\StrukturData\PraktikumStrukturData\Pertemuan 7> cd 'd:\Semester5\StrukturData\PraktikumStrukturData\Pertemuan 7\UNGUIDED\output'
PS D:\Semester5\StrukturData\PraktikumStrukturData\Pertemuan 7\UNGUIDED\output> & .\'unguided2.exe'
Masukkan Kata: Telkom Purwokerto
Datastack Array:
Data: otrekowruP mokleT
PS D:\Semester5\StrukturData\PraktikumStrukturData\Pertemuan 7\UNGUIDED\output> |
```