

**LAPORAN PRAKTIKUM STRUKTUR  
DATA DAN ALGORITMA**

**MODUL VI  
STACK**



**Disusun Oleh :**

Muhammad Arsyad Zaidan (2311102058)

**Dosen**

Wahyu Andi Saputra, S.Pd., M.Eng.

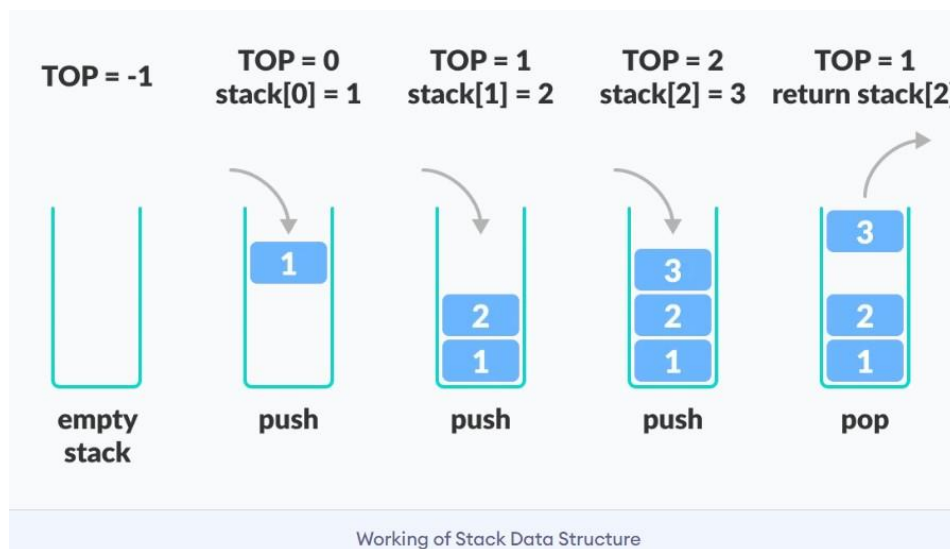
**PROGRAM STUDI S1 TEKNIK INFORMATIKA  
FAKULTAS INFORMATIKA  
INSTITUT TEKNOLOGI TELKOM PURWOKERTO  
2024**

## B. Dasar Teori

### a. Pengertian Hash Table

Stack adalah struktur data sederhana yang digunakan untuk menyimpan data (mirip dengan Linked Lists). Dalam tumpukan, urutan kedatangan data penting. Sebuah tumpukan piring di kafetaria adalah contoh bagus dari tumpukan. Piring ditambahkan ke tumpukan saat mereka dibersihkan dan ditempatkan di bagian atas. Ketika sebuah piring dibutuhkan, diambil dari bagian atas tumpukan. Piring pertama yang ditempatkan di tumpukan adalah yang terakhir digunakan.

Definisi: Sebuah tumpukan adalah daftar terurut di mana penyisipan dan penghapusan dilakukan di satu ujung, disebut atas. Elemen terakhir yang dimasukkan adalah yang pertama dihapus. Oleh karena itu, disebut daftar Last in First out (LIFO).



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.
- Top (Atas):** Mendapatkan nilai atau melihat elemen teratas pada tumpukan tanpa menghapusnya.
- 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.

## 1. Latihan Guided dan Unguided

### 1) Guided

#### Guided 1

Source Code

```
#include <iostream>
using namespace std;
string arrayBuku[5];
int maksimal = 5, top = 0;
bool isFull()
{
    return (top == maksimal);
}
bool isEmpty()
{
    return (top == 0);
}
void pushArrayBuku(string data)
{
    if (isFull())
    {
        cout << "Data telah penuh" << endl;
    }
    else
    {
        arrayBuku[top] = data;
        top++;
    }
}
void popArrayBuku()
{
    if (isEmpty())
```

```

    {
        cout << "Tidak ada data yang dihapus" << endl;
    }
    else
    {
        arrayBuku[top - 1] = "";
        top--;
    }
}

void peekArrayBuku(int posisi)
{
    if (isEmpty())
    {
        cout << "Tidak ada data yang bisa dilihat" << endl;
    }
    else
    {
        int index = top;
        for (int i = 1; i <= posisi; i++)
        {
            index--;
        }
        cout << "Posisi ke " << posisi << " adalah " << arrayBuku[index] << endl;
    }
}

int countStack()
{
    return top;
}

void changeArrayBuku(int posisi, string data)
{
    if (posisi > top)
    {
        cout << "Posisi melebihi data yang ada" << endl;
    }
    else
    {
        int index = top;
        for (int i = 1; i <= posisi; i++)
        {
            index--;
        }
        arrayBuku[index] = data;
    }
}

void destroyArraybuku()
{

```

```

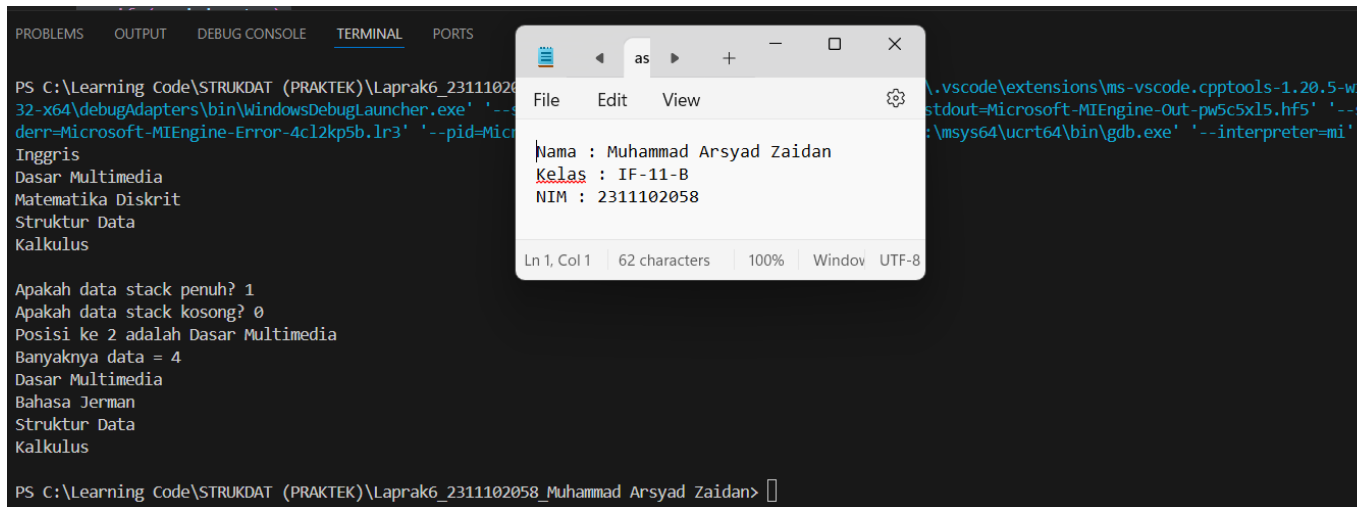
        for (int i = top; i >= 0; i--)
        {
            arrayBuku[i] = "";
        }
        top = 0;
    }
}

void cetakArrayBuku()
{
    if (isEmpty())
    {
        cout << "Tidak ada data yang dicetak" << endl;
    }
    else
    {
        for (int i = top - 1; i >= 0; i--)
        {
            cout << arrayBuku[i] << endl;
        }
    }
}

int main()
{
    pushArrayBuku("Kalkulus");
    pushArrayBuku("Struktur Data");
    pushArrayBuku("Matematika Diskrit");
    pushArrayBuku("Dasar Multimedia");
    pushArrayBuku("Inggris");
    cetakArrayBuku();
    cout << "\n";
    cout << "Apakah data stack penuh? " << isFull() << endl;
    cout << "Apakah data stack kosong? " << isEmpty() << endl;
    peekArrayBuku(2);
    popArrayBuku();
    cout << "Banyaknya data = " << countStack() << endl;
    changeArrayBuku(2, "Bahasa Jerman");
    cetakArrayBuku();
    cout << "\n";
}

```

Output Code



The screenshot shows a VS Code terminal window with the following output:

```
PS C:\Learning Code\STRUKNAT (PRAKTEK)\Laprak6_2311102058_Muhammad Arsyad Zaidan>
32-x64\debugAdapters\bin\WindowsDebugLauncher.exe' '--stdout=Microsoft-MIEngine-Out-pw5c5x15.hf5' '--pid=Microsoft-MIEngine-Error-4c12kp5b.lr3' '--pid=Microsoft-MIEngine-Error-4c12kp5b.lr3' '--interpreter=miengine'
Inggris
Dasar Multimedia
Matematika Diskrit
Struktur Data
Kalkulus

Apakah data stack penuh? 1
Apakah data stack kosong? 0
Posisi ke 2 adalah Dasar Multimedia
Banyaknya data = 4
Dasar Multimedia
Bahasa Jerman
Struktur Data
Kalkulus

PS C:\Learning Code\STRUKNAT (PRAKTEK)\Laprak6_2311102058_Muhammad Arsyad Zaidan>
```

A dialog box is overlaid on the terminal, displaying the following information:

```
Nama : Muhammad Arsyad Zaidan
Kelas : IF-11-B
NIM : 2311102058
```

The dialog box also shows a status bar at the bottom: Ln 1, Col 1 | 62 characters | 100% | Window | UTF-8.

## Deskripsi Code

Penerapan stack pada pemrograman tersebut berfungsi untuk untuk mengelola stack, termasuk menambah, menghapus, melihat, mengubah, dan mencetak elemen stack. Pada program tersebut juga memasang Operasi penyisipan, pencarian, penghapusan, dan pencetakan ditunjukkan dengan jelas dalam program ini. Dan code dibuat untuk mengelola data pada pelajaran mata kuliah.

## 2) Unguided

### Unguided 1

#### Source Code

```
#include <iostream>
#include <stack>
#include <cctype>
#include <string>

using namespace std;

bool isPalindrome(const string &str) {
    stack<char> s;
    string cleanedStr;

    for (char ch : str) {
        if (isalpha(ch)) {
            cleanedStr += tolower(ch);
        }
    }
}
```

```

    for (char ch : cleanedStr) {
        s.push(ch);
    }

    for (char ch : cleanedStr) {
        if (ch != s.top()) {
            return false;
        }
        s.pop();
    }

    return true;
}

int main() {
    string input;

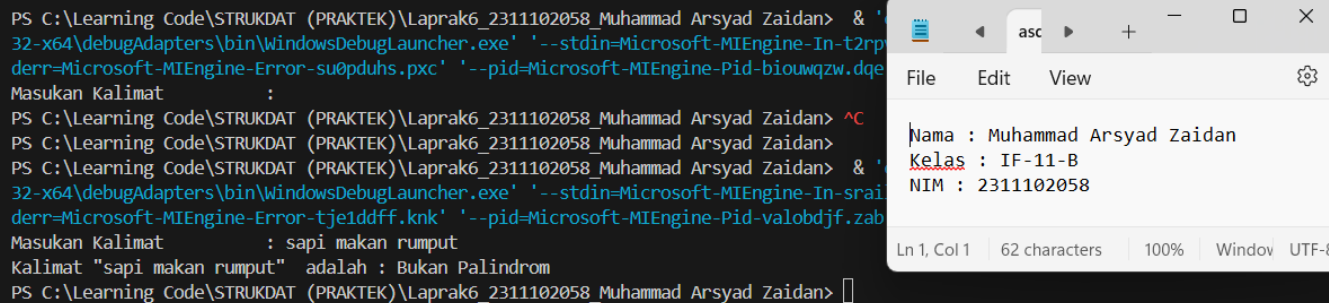
    cout << "Masukan Kalimat\t\t : ";
    getline(cin, input);

    if (isPalindrome(input)) {
        cout << "Kalimat \"" << input << "\" adalah\t : Palindrom" << endl;
    } else {
        cout << "Kalimat \"" << input << "\" adalah : Bukan Palindrom" << endl;
    }

    return 0;
}

```

## Output Code



The screenshot shows a Windows command prompt window with the following commands and output:

```

PS C:\Learning Code\STRUKDAT (PRAKTEK)\Laprak6_2311102058_Muhammad Arsyad Zaidan> & 'C:\Program Files\Microsoft Visual Studio\2019\Community\VC\Tools\MSI\x64\debugAdapters\bin\WindowsDebugLauncher.exe' '--stdin=Microsoft-MIEngine-In-t2rp...
derr=Microsoft-MIEngine-Error-su0pduhs.pxc' '--pid=Microsoft-MIEngine-Pid-biouwqzw.dqe
Masukan Kalimat :
PS C:\Learning Code\STRUKDAT (PRAKTEK)\Laprak6_2311102058_Muhammad Arsyad Zaidan> ^C
PS C:\Learning Code\STRUKDAT (PRAKTEK)\Laprak6_2311102058_Muhammad Arsyad Zaidan>
PS C:\Learning Code\STRUKDAT (PRAKTEK)\Laprak6_2311102058_Muhammad Arsyad Zaidan> & 'C:\Program Files\Microsoft Visual Studio\2019\Community\VC\Tools\MSI\x64\debugAdapters\bin\WindowsDebugLauncher.exe' '--stdin=Microsoft-MIEngine-In-srai...
derr=Microsoft-MIEngine-Error-tje1ddff.knk' '--pid=Microsoft-MIEngine-Pid-valobdjf.zab
Masukan Kalimat : sapi makan rumput
Kalimat "sapi makan rumput" adalah : Bukan Palindrom
PS C:\Learning Code\STRUKDAT (PRAKTEK)\Laprak6_2311102058_Muhammad Arsyad Zaidan>

```

Overlaid on the command prompt is a Notepad++ window titled 'asc' containing the following text:

```

Nama : Muhammad Arsyad Zaidan
Kelas : IF-11-B
NIM : 2311102058

```

The Notepad++ status bar at the bottom indicates: Ln 1, Col 1 | 62 characters | 100% | Window | UTF-8.

## Deskripsi Code

Code tersebut berisikan tentang penerapan stack. Di program juga terdapat fungsi push, top, dan pop digunakan pada code tersebut yang berfungsi untuk memasukkan karakter pada stack yang paling bawah dan karakter yang terakhir kali dimasukkan akan dipindahkan ke paling atas yang akhirnya akan mengambil karakter dari puncak stack satu per satu dan membandingkannya dengan karakter dalam string dan program tersebut dapat memeriksa apakah string tersebut adalah palindrom atau tidak. Program stack pada code ini menggunakan prinsip LIFO (Last In, First Out).

## Unguided 2

### Source Code

```
#include <iostream>
#include <stack>
#include <string>

using namespace std;

string reverseWords(const string &str) {
    stack<char> s;
    string result = "";

    //Membalikkan setiap karakter dalam string menggunakan stack
    for (char ch : str) {
        s.push(ch);
    }

    //Mengambil karakter dari stack untuk membentuk kalimat yang dibalik
    while (!s.empty()) {
        result += s.top();
        s.pop();
    }

    return result;
}

int main() {
    string input;

    cout << "Masukkan kalimat (minimal 3 kata): ";
    getline(cin, input);

    //Memastikan bahwa input memiliki minimal 3 kata
    int wordCount = 0;
    for (char ch : input) {
```



```

        if (ch == ' ') {
            wordCount++;
        }
    }
    wordCount++; //Menambah satu untuk kata terakhir

    if (wordCount < 3) {
        cout << "Error: Kalimat harus memiliki minimal 3 kata." << endl;
    } else {
        string reversed = reverseWords(input);
        cout << "Hasil: " << reversed << endl;
    }

    return 0;
}

```

## Output Code

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Learning Code\STRUKDAT (PRAKTEK)\Laprak6_2311102058_Muhammad Arsyad Zaidan> & '32-x64\debugAdapters\bin\WindowsDebugLauncher.exe' '--stdin=Microsoft-MIEngine-In-2er4derr=Microsoft-MIEngine-Error-cidjntj2.pn3' '--pid=Microsoft-MIEngine-Pid-w5mshldc.jf4
Masukkan kalimat (minimal 3 kata):
PS C:\Learning Code\STRUKDAT (PRAKTEK)\Laprak6_2311102058_Muhammad Arsyad Zaidan> ^C
PS C:\Learning Code\STRUKDAT (PRAKTEK)\Laprak6_2311102058_Muhammad Arsyad Zaidan>
PS C:\Learning Code\STRUKDAT (PRAKTEK)\Laprak6_2311102058_Muhammad Arsyad Zaidan> & '32-x64\debugAdapters\bin\WindowsDebugLauncher.exe' '--stdin=Microsoft-MIEngine-In-gmepderr=Microsoft-MIEngine-Error-ezlubadh.3nx' '--pid=Microsoft-MIEngine-Pid-j0xrxf3.gtx
Masukkan kalimat (minimal 3 kata): daffa minum air
Hasil: ria munim affad
PS C:\Learning Code\STRUKDAT (PRAKTEK)\Laprak6_2311102058_Muhammad Arsyad Zaidan>

```

asc

File Edit View

Nama : Muhammad Arsyad Zaidan  
 Kelas : IF-11-B  
 NIM : 2311102058

Ln 1, Col 1 | 62 characters | 100% | Window UTF-8

## Deskripsi Code

Code ini juga menggunakan fungsi push, top, dan pop, yang mana berfungsi untuk membalikkan urutan karakter pada string dan berprinsip LIFO (Last In, First Out), pemrograman tersebut berisikan aktivitas mengambil kata dari stack untuk membentuk kalimat yang dibalik, pengecekan jumlah kata, sehingga dengan semua itu dapat digunakan untuk membalikkan kata seperti yang ada di output di atas yaitu “daffa minum air” menjadi “ria munim affad”.

## 2. Kesimpulan

Dari pratikum ini, mahasiswa dapat mempelajari bagaimana konsep Pemrograman C++ pada stack. Stack adalah struktur data sederhana yang digunakan untuk menyimpan data. Cara kerja tersebut hampir sama dengan Linked List.

Diantara Stack dan Linked List, Struktur data Stack perlu melakukan penumpukan, yang daftar terurut di mana penyisipan dan penghapusan dilakukan di satu ujung, disebut atas. Elemen terakhir yang dimasukkan adalah yang pertama dihapus. Oleh karena itu, disebut daftar Last in First out (LIFO). Sedangkan Linked List, mengharuskan struktur data linier berbentuk rantai simpul di mana setiap simpul menyimpan 2 item, yaitu nilai data dan pointer ke simpul elemen berikutnya.

## 3. Referensi

- [1] Tim Asisten Asprak, "Stack", Learning Management System, 2024.
- [2] Sindar, Anita, 2018, "Struktur Data Dan Algoritma Dengan C++", Serang : Cv. Aa. Rizky.
- [3] Geeksforgeeks, "Stack in C++ STL", <https://www.geeksforgeeks.org/stack-in-cpp-stl/>. terakhir kali diakses pada tanggal 20 Mei 2024.