

LAPORAN PRAKTIKUM
STRUKTUR DATA
STACK



Nama :

Muhammad Mahrus Ali (2311104006)

Dosen :

Yudha Islami Sulistya,S.Kom.,M.Cs.

PROGRAM STUDI S1 REKAYASA PERANGKAT LUNAK
FAKULTAS INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO
2024

A. Soal Tugas Pendahuluan

1. Buat file stack.h

```
#ifndef STACK_H
#define STACK_H

typedef char infotype;

struct stack {
    infotype info[15];
    int Top;
};

void createStack(stack &S);
bool isEmpty(stack S);
bool isFull(stack S);
void push(stack &S, infotype x);
infotype pop(stack &S);
void printInfo(stack S);

#endif
```

2. Membuat file stack.cpp

```
#include "stack.h"
#include <iostream>
using namespace std;

void createStack(stack &S) {
    S.Top = 0;
}

bool isEmpty(stack S) {
    return S.Top == 0;
}

bool isFull(stack S) {
    return S.Top == 15;
}

void push(stack &S, infotype x) {
    if (!isFull(S)) {
        S.Top++;
        S.info[S.Top] = x;
    } else {
        cout << "Stack penuh, tidak bisa menambahkan elemen." << endl;
    }
}

infotype pop(stack &S) {
    if (!isEmpty(S)) {
        infotype x = S.info[S.Top];
        S.Top--;
        return x;
    } else {
        cout << "Stack kosong, tidak ada elemen yang bisa diambil." << endl;
        return '\0';
    }
}

void printInfo(stack S) {
    if (isEmpty(S)) {
        cout << "Stack kosong." << endl;
    } else {
        for (int i = S.Top; i > 0; i--) {
            cout << S.info[i] << " ";
        }
        cout << endl;
    }
}
```

3. Setelah file stack.h dan stack.cpp sudah ada, selanjutnya buat file main.cpp

```
#include "stack.h"
#include <iostream>
using namespace std;

int main() {
    stack S;
    createStack(S);

    char frasaAwal[] = {'P', 'E', 'R', 'C', 'A', 'Y', 'A', 'D', 'I', 'R', 'I'};
    char frasaPop[] = {'D', 'I', 'R', 'I'};

    for (char c : frasaAwal) {
        push(S, c);
    }

    cout << "\nIsi stack awal: ";
    printInfo(S);

    cout << "Isi stack setelah pop: ";
    for (char c : frasaPop) {
        pop(S);
    }
    printInfo(S);

    return 0;
}
```

Output kode diatas:

```
PS D:\Kuliah\struktur data\07_Stack\TP\Soal_TP.cpp> g++ stack.cpp main.cpp -o TP
PS D:\Kuliah\struktur data\07_Stack\TP\Soal_TP.cpp> ./TP

Isi stack awal: I R I D A Y A C R E P
Isi stack setelah pop: A Y A C R E P
PS D:\Kuliah\struktur data\07_Stack\TP\Soal_TP.cpp>
```

B. Latihan Unguided

1. Buat program untuk menentukan apakah kalimat tersebut yang di inputkan dalam program stack adalah palindrom atau tidak. Palindrom kalimat yang dibaca dari depan dan belakang sama. Jelaskan cara kerjanya.

Input:

```

#include <iostream>
#include <stack>
#include <string>
using namespace std;

bool isiPalindrom(string kalimat) {
    stack<char> s;
    string cleanedKalimat = "";

    for (char c : kalimat) {
        if (isalnum(c)){
            cleanedKalimat += tolower(c);
        }
    }

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

    for (char c : kalimat){
        if (c != s.top()) {
            return false;
        }
        s.pop();
    }
    return true;
}

int main(){
    string kalimat;

    cout << "Masukan Kalimat: " ;
    getline(cin, kalimat);

    if(isiPalindrom(kalimat)){
        cout << "kalimat tersebut palindrom\n" << endl;
    } else {
        cout << "kalimat tersebut not palindrom\n" << endl;
    }

    return 0;
}

```

Fungsi isiPalindrom terlebih dahulu membersihkan kalimat dari karakter non-alfanumerik dan mengubah semua huruf menjadi huruf kecil untuk memudahkan perbandingan. Setiap karakter hasil pembersihan dimasukkan ke dalam stack. Kemudian, program membandingkan setiap karakter dari awal kalimat dengan karakter yang diambil dari stack, yang tersusun terbalik. Jika semua pasangan karakter cocok, fungsi mengembalikan true, menandakan bahwa kalimat tersebut adalah palindrom. Program utama meminta input dari pengguna, menjalankan fungsi isiPalindrom, dan menampilkan hasilnya sesuai dengan apakah kalimat itu palindrom atau bukan.

Outputnya:

```

PS D:\Kuliah\struktur data\07_Stack> cd "d:\Kuliah\struktur data\07_Stack\Unguided\" ; if ($?) { g++
unguided_01.cpp -o unguided_01 } ; if ($?) { .\unguided_01 }
Masukan Kalimat: ini
kalimat tersebut palindrom

PS D:\Kuliah\struktur data\07_Stack\Unguided> cd "d:\Kuliah\struktur data\07_Stack\Unguided\" ; if ($?)
{ g++ unguided_01.cpp -o unguided_01 } ; if ($?) { .\unguided_01 }
Masukan Kalimat: TELU
kalimat tersebut bukan palindrom

PS D:\Kuliah\struktur data\07_Stack\Unguided>

```

2. Buat program untuk melakukan pembalikan terhadap kalimat menggunakan stack dengan minimal 3 kata. Jelaskan output program dan sourcecodenya beserta operasi atau fungsi yang dibuat.

Input:

```

#include <iostream>
#include <stack>
#include <sstream>
using namespace std;

void kalimatMundur(string kalimat) {
    stack<string> s;
    stringstream ss(kalimat);
    string kata;

    while (ss >> kata){
        s.push(kata);
    }

    cout << "Kalimat Kebalik: ";
    while (!s.empty()) {
        cout << s.top() << " ";
        s.pop();
    }
}

int main(){
    string kalimat;

    cout << "\nMasukan tiga Kalimat (contoh : aku suka kamu): " ;
    getline(cin, kalimat);
    kalimatMundur(kalimat);

    return 0 ;
}

```

Pertama, program mengimpor pustaka yang diperlukan, termasuk iostream, stack, dan sstream. Di dalam fungsi kalimatMundur, sebuah stack digunakan untuk menyimpan kata-kata dari kalimat yang dimasukkan. Kalimat dibaca menggunakan stringstream, yang memungkinkan pemisahan kata-kata berdasarkan spasi. Selanjutnya, setiap kata yang diperoleh dari stringstream didorong ke dalam stack. Setelah semua kata

dimasukkan, program mencetak kalimat yang dibalik dengan mengeluarkan kata-kata dari stack satu per satu hingga stack kosong. Dalam fungsi main, program meminta pengguna untuk memasukkan kalimat, yang kemudian diteruskan ke fungsi kalimatMundur untuk dibalik dan ditampilkan hasilnya.

Outputnya:



```
PS D:\Kuliah\struktur data\07_Stack> cd "d:\Kuliah\struktur data\07_Stack\Unguided\" ; if ($?) { g++  
unguided_02.cpp -o unguided_02 } ; if ($?) { .\unguided_02 }  
  
Masukan tiga Kalimat (contoh : aku suka kamu): aku dan kamu  
Kalimat Kebalik: kamu dan aku  
PS D:\Kuliah\struktur data\07_Stack\Unguided>
```