

**LAPORAN PRAKTIKUM
STRUKTUR DATA**

**MODUL 7
PENGENALAN CODE BLOCKS**



Disusun Oleh :
NAMA : MUHAMMAD ATHA RIZQI
RADITYA
NIM : 103112400184

Dosen
FAHRUDIN MUKTI WIBOWO

**PROGRAM STUDI STRUKTUR DATA
FAKULTAS INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO
2025**

A. Dasar Teori

C++ adalah pengembangan dari bahasa C yang dibuat oleh Bjarne Stroustrup sekitar tahun 1980-an. C++ disebut bahasa multi-paradigma, artinya bisa dipakai dengan gaya prosedural (pakai fungsi biasa), berorientasi objek (pakai class dan object), atau bahkan gabungan keduanya. C++ punya dasar-dasar seperti variabel, operator percabangan (if, switch), perulangan (for, while), dan bisa memakai class untuk membuat objek.

B. Guided (berisi screenshot source code & output program disertai penjelasannya)

Guided 1

H

```
#ifndef STACK_H
#define STACK_H
#include <iostream>
using namespace std;

typedef int infotype;

struct Stack {
    infotype info[20];
    int top;
};

void createStack(Stack &S);
void push(Stack &S, infotype x);
infotype pop(Stack &S);
void printInfo(Stack S);
void balikStack(Stack &S);
void pushAscending(Stack &S, infotype x); // NO.2
void getInputStream(Stack &S); // NO.3

#endif
```

Cpp

```
#include "stack.h"

void createStack(Stack &S){
    S.top = -1;
}

void push(Stack &S, infotype x){
    if(S.top < 19){
        S.top++;
        S.info[S.top] = x;
    }
}

infotype pop(Stack &S){
    infotype x = -1;
    if(S.top >= 0){
        x = S.info[S.top];
        S.top--;
    }
    return x;
}

void printInfo(Stack S){
    cout << "[TOP] ";
    for(int i = S.top; i >= 0; i--){
        cout << S.info[i] << " ";
    }
    cout << endl;
}
```

```
void balikStack(Stack &S){  
    Stack temp;  
    createStack(temp);  
  
    while(S.top >= 0){  
        push(temp, pop(S));  
    }  
    S = temp;  
}  
  
void pushAscending(Stack &S, infotype x){  
    Stack temp;  
    createStack(temp);  
  
    while(S.top >= 0 && S.info[S.top] < x){  
        push(temp, pop(S));  
    }  
  
    push(S, x);  
  
    while(temp.top >= 0){  
        push(S, pop(temp));  
    }  
}  
  
void getInputStream(Stack &S){  
    char c;  
    cout << "Masukkan angka tanpa spasi, kemudian ENTER: ";  
    cin.ignore();
```

```
while(cin.get(c) && c != '\n'){

    if(isdigit(c)){

        push(S, c - '0');

    }

}

}
```

Main

```
#include "stack.h"

int main(){

    cout << "Hello world!" << endl;

    Stack S;

    createStack(S);

    push(S,3);

    push(S,4);

    push(S,8);

    pop(S);

    push(S,2);

    push(S,3);

    pop(S);

    push(S,9);

    printInfo(S);

    cout << "balik stack" << endl;

    balikStack(S);

    printInfo(S);

    return 0;

}
```

Screenshots Output

```
PS C:\Users\ASUS\Documents\Struktur Data\Laporan\modul7>
● >>
PS C:\Users\ASUS\Documents\Struktur Data\Laporan\modul7>
● >>
Hello world!
[TOP] 9 2 4 3
balik stack
[TOP] 3 4 2 9
```

Deskripsi:

Program ini merupakan implementasi struktur data Doubly Linked List yang digunakan untuk menyimpan dan mengelola data secara dinamis. Setiap elemen list memiliki pointer next dan prev sehingga memungkinkan proses traversal baik dari depan maupun belakang. Program menyediakan fitur untuk menambah elemen, mencari data, menampilkan seluruh isi list, serta menghapus elemen di posisi awal, akhir, maupun setelah node tertentu sesuai kebutuhan pengguna melalui menu interaktif.

Guided 2

Main

```
#include "stack.h"

int main(){
    cout << "Hello world!" << endl;
    Stack S;
    createStack(S);
    pushAscending(S,3);
    pushAscending(S,4);
    pushAscending(S,8);
    pushAscending(S,2);
    pushAscending(S,3);
    pushAscending(S,9);
```

```

printInfo(S);

cout << "balik stack" << endl;

balikStack(S);

printInfo(S);

return 0;

}

```

Screenshots Output

```

PROBLEMS    OUTPUT    DEBUG CONSOLE    TERMINAL    PORTS

PS C:\Users\ASUS\Documents\Struktur Data\Laporan\modul7>
● >>
PS C:\Users\ASUS\Documents\Struktur Data\Laporan\modul7>
● >>
Hello world!
[TOP] 2 3 3 4 8 9
balik stack
[TOP] 9 8 4 3 3 2

```

Deskripsi:

Program ini merupakan implementasi struktur data Doubly Linked List yang digunakan untuk menyimpan dan mengelola data secara dinamis. Setiap elemen list memiliki pointer next dan prev sehingga memungkinkan proses traversal baik dari depan maupun belakang. Program menyediakan fitur untuk menambah elemen, mencari data, menampilkan seluruh isi list, serta menghapus elemen di posisi awal, akhir, maupun setelah node tertentu sesuai kebutuhan pengguna melalui menu interaktif.

Guided 3

Main

```

#include "stack.h"

int main(){

    cout << "Hello world!" << endl;

    Stack S;

    createStack(S);
}

```

```
getInputStream(S);

printInfo(S);

cout << "balik stack" << endl;

balikStack(S);

printInfo(S);

return 0;

}
```

Screenshots Output

```
PROBLEMS    OUTPUT    DEBUG CONSOLE    TERMINAL    PORTS

Laporan\modul7> g++ main.cpp stack.cpp -o stackProgram
>>
>>
Hello world!
Masukkan angka tanpa spasi, kemudian ENTER: 4729601
[TOP] 1 0 6 9 2 7
balik stack
[TOP] 7 2 9 6 0 1
```

Deskripsi:

Program ini merupakan implementasi struktur data Doubly Linked List yang digunakan untuk menyimpan dan mengelola data secara dinamis. Setiap elemen list memiliki pointer next dan prev sehingga memungkinkan proses traversal baik dari depan maupun belakang. Program menyediakan fitur untuk menambah elemen, mencari data, menampilkan seluruh isi list, serta menghapus elemen di posisi awal, akhir, maupun setelah node tertentu sesuai kebutuhan pengguna melalui menu interaktif.

D. Kesimpulan

Kesimpulan dari keempat program C++ tersebut adalah bahwa masing-masing program mengajarkan konsep dasar pemrograman dan logika perhitungan menggunakan array, pointer, fungsi, serta struktur perulangan. Dari program pertama, dipelajari cara mengolah data mahasiswa, menghitung rata-rata, dan menampilkan hasil dalam format tabel. Program kedua memperkenalkan array dinamis dan penggunaan pointer untuk mengakses serta memanipulasi data. Program ketiga menekankan penggunaan fungsi terpisah (modularisasi) untuk menghitung rata-rata, nilai maksimum, dan minimum dari sekumpulan data. Sementara program keempat mengajarkan konsep perulangan bersarang untuk mencetak pola angka berbentuk segitiga. Secara keseluruhan, keempat

program ini membantu memahami logika dasar, manipulasi data, penggunaan fungsi, array, pointer, serta struktur kontrol dalam bahasa pemrograman C++.

E. Referensi

- Alhazmi, A.,& Huang, W.(2020). Teaching loops concept through visualization construction. International Journal of Instruction. 13(4),399-114.
- Sáez-López, J. M., Román-González, M., & Vázquez-Cano, E. (2020). Dampak penggunaan bahasa pemrograman terhadap pemahaman konsep loop pada siswa K-12. Pendidikan dan Teknologi Informasi, 25(1), 987-1000.
- Stroustrup, B. (1999). Tinjauan umum bahasa pemrograman C++. ACM SIGPLAN Notices, 34(4), 7-18.
- Stroustrup, B. (1999). Tinjauan umum bahasa pemrograman C++. ACM SIGPLAN Notices, 34(4), 7-18.