

**LAPORAN PRAKTIKUM  
STRUKTUR DATA**

**MODUL 2  
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

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

int main() {
    const int JUMLAH_MHS = 5;
    const int JUMLAH_MK = 3;

    string nama[JUMLAH_MHS];
    float nilai[JUMLAH_MHS][JUMLAH_MK];
    float rata[JUMLAH_MHS];
    int indeksTerbaik = 0;

    cout << "=== Input Data Nilai Mahasiswa ===\n";
    for (int i = 0; i < JUMLAH_MHS; i++) {
        cout << "\nMasukkan nama mahasiswa ke-" << i + 1 << ": ";
        getline(cin, nama[i]);

        float total = 0;
        for (int j = 0; j < JUMLAH_MK; j++) {
            cout << " Nilai mata kuliah " << j + 1 << ": ";
            cin >> nilai[i][j];
            total += nilai[i][j];
        }
        cin.ignore();
        rata[i] = total / JUMLAH_MK;

        if (rata[i] > rata[indeksTerbaik]) {
            indeksTerbaik = i;
        }
    }

    cout << "\n\n=== Daftar Nilai Mahasiswa ===\n";
    cout << left << setw(15) << "Nama"
```

```

<< setw(10) << "MK1"
<< setw(10) << "MK2"
<< setw(10) << "MK3"
<< setw(10) << "Rata-rata"
<< "Keterangan" << endl;
cout << string(65, '-') << endl;

for (int i = 0; i < JUMLAH_MHS; i++) {
    cout << left << setw(15) << nama[i];
    for (int j = 0; j < JUMLAH_MK; j++) {
        cout << setw(10) << nilai[i][j];
    }
    cout << setw(10) << fixed << setprecision(2) << rata[i];
    if (i == indeksTerbaik)
        cout << "Terbaik";
    cout << endl;
}

cout << "\nMahasiswa dengan rata-rata tertinggi: " << nama[indeksTerbaik]
<< " (" << fixed << setprecision(2) << rata[indeksTerbaik] << ") \n";

return 0;
}

```

## Screenshots Output

```

PS C:\Users\ASUS\Documents\Struktur Data> cd "c:\Users\ASUS\Documents\Struktur Data\" ; if ($?) { g++ modul2s1.cpp -o
=== Input Data Nilai Mahasiswa ===

Masukkan nama mahasiswa ke-1: A
  Nilai mata kuliah 1: 90
  Nilai mata kuliah 2: 95
  Nilai mata kuliah 3: 80

Masukkan nama mahasiswa ke-2: B
  Nilai mata kuliah 1: 90
  Nilai mata kuliah 2: 85
  Nilai mata kuliah 3: 70

Masukkan nama mahasiswa ke-3: C
  Nilai mata kuliah 1: 80
  Nilai mata kuliah 2: 85
  Nilai mata kuliah 3: 85

Masukkan nama mahasiswa ke-4: D
  Nilai mata kuliah 1: 90
  Nilai mata kuliah 2: 75
  Nilai mata kuliah 3: 80

Masukkan nama mahasiswa ke-5: E
  Nilai mata kuliah 1: 95
  Nilai mata kuliah 2: 95
  Nilai mata kuliah 3: 95

=== Daftar Nilai Mahasiswa ===
Nama          MK1      MK2      MK3      Rata-rata Keterangan
-----
A              90       95       80       88.33
B            90.00      85.00      70.00      81.67
C            80.00      85.00      85.00      83.33
D            90.00      75.00      80.00      81.67
E            95.00      95.00      95.00      95.00      Terbaik

Mahasiswa dengan rata-rata tertinggi: E (95.00)

```

Deskripsi:

Program C++ tersebut digunakan untuk menginput data nilai beberapa mahasiswa, menghitung rata-rata nilai tiap mahasiswa, dan menentukan mahasiswa dengan rata-rata tertinggi.

Guided 2

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

int main() {
    int N;
    cout << "Masukkan jumlah elemen array (N): ";
    cin >> N;

    int *arr = new int[N];

    cout << "\n=== Input Elemen Array ===\n";
    for (int i = 0; i < N; i++) {
        cout << "Elemen ke-" << i + 1 << ": ";
        cin >> *(arr + i);
    }

    int jumlah = 0;
    int maks = *arr;
    int min = *arr;

    for (int *ptr = arr; ptr < arr + N; ptr++) {
        jumlah += *ptr;
        if (*ptr > maks) maks = *ptr;
        if (*ptr < min) min = *ptr;
    }

    cout << "\n=== Hasil Perhitungan ===\n";
    cout << "Jumlah seluruh bilangan : " << jumlah << endl;
    cout << "Nilai maksimum      : " << maks << endl;
    cout << "Nilai minimum      : " << min << endl;

    delete[] arr;

    return 0;
}
```

Screenshots Output

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

PS C:\Users\ASUS\Documents\Struktur Data> cd "c:\Users\ASUS\Documents\Struktur Data\" ; if ($?) { g++ modul2s2.cpp -
● Masukkan jumlah elemen array (N): 5

=== Input Elemen Array ===
Elemen ke-1: 10
Elemen ke-2: 25
Elemen ke-3: 20
Elemen ke-4: 15
Elemen ke-5: 50

=== Hasil Perhitungan ===
Jumlah seluruh bilangan : 120
Nilai maksimum          : 50
Nilai minimum           : 10
○ PS C:\Users\ASUS\Documents\Struktur Data> █
```

### Deskripsi:

Program C++ ini berfungsi untuk menginput sejumlah bilangan ke dalam array dinamis, kemudian menghitung jumlah keseluruhan elemen, mencari nilai maksimum, dan nilai minimum dengan memanfaatkan pointer untuk mengakses setiap elemen array. Pengguna diminta memasukkan jumlah elemen terlebih dahulu, lalu mengisi nilai-nilai array tersebut. Setelah semua nilai dimasukkan, program melakukan perhitungan menggunakan perulangan dan operasi pointer arithmetic, kemudian menampilkan hasil berupa jumlah total, nilai tertinggi, dan nilai terendah. Terakhir, memori yang digunakan untuk array dibebaskan dengan perintah `delete[]` agar tidak terjadi kebocoran memori

### Guided 3

```
#include <iostream>

using namespace std;

float hitungRataRata(int nilai[], int n) {
    int total = 0;
    for (int i = 0; i < n; i++) {
        total += nilai[i];
    }
    return static_cast<float>(total) / n;
}

void cariMaksMin(int nilai[], int n, int &maks, int &min) {
    maks = nilai[0];
    min = nilai[0];
```

```

        for (int i = 1; i < n; i++) {
            if (nilai[i] > maks)
                maks = nilai[i];
            if (nilai[i] < min)
                min = nilai[i];
        }
    }

int main() {
    int N;

    cout << "Masukkan jumlah siswa: ";
    cin >> N;

    int nilai[N];

    cout << "\n=== Input Nilai Ujian Siswa ===\n";
    for (int i = 0; i < N; i++) {
        cout << "Nilai siswa ke-" << i + 1 << ": ";
        cin >> nilai[i];
    }

    float rata = hitungRataRata(nilai, N);
    int maks, min;
    cariMaksMin(nilai, N, maks, min);

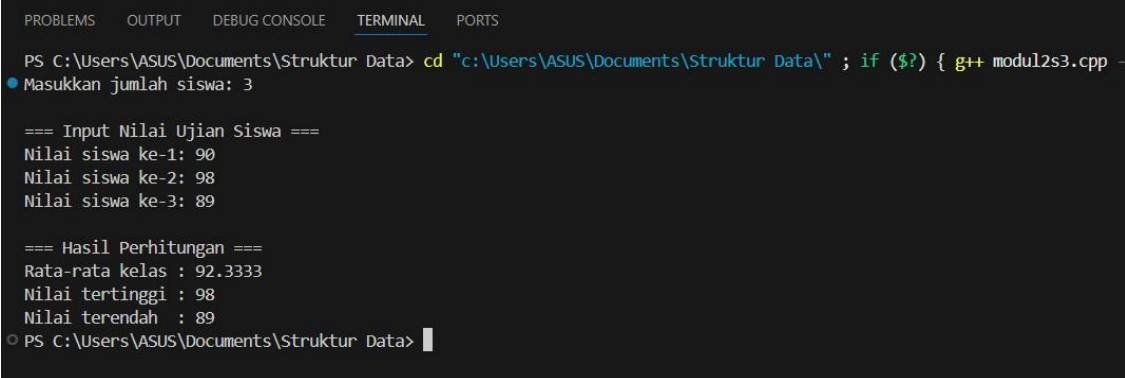
    cout << "\n=== Hasil Perhitungan ===\n";
    cout << "Rata-rata kelas : " << rata << endl;
    cout << "Nilai tertinggi : " << maks << endl;
    cout << "Nilai terendah : " << min << endl;

    return 0;
}

```

```
}
```

## Screenshots Output



```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS
PS C:\Users\ASUS\Documents\Struktur Data> cd "c:\Users\ASUS\Documents\Struktur Data\" ; if ($?) { g++ modul2s3.cpp -
• Masukkan jumlah siswa: 3

=== Input Nilai Ujian Siswa ===
Nilai siswa ke-1: 90
Nilai siswa ke-2: 98
Nilai siswa ke-3: 89

=== Hasil Perhitungan ===
Rata-rata kelas : 92.3333
Nilai tertinggi : 98
Nilai terendah : 89
PS C:\Users\ASUS\Documents\Struktur Data> |
```

## Deskripsi:

Program C++ ini digunakan untuk menghitung **rata-rata nilai ujian siswa**, serta menentukan **nilai tertinggi dan nilai terendah** di antara seluruh siswa. Pengguna diminta memasukkan jumlah siswa, kemudian nilai ujian masing-masing siswa disimpan dalam array. Program memiliki dua fungsi: `hitungRataRata()` yang menjumlahkan seluruh nilai dan menghitung rata-ratanya, serta `cariMaksMin()` yang mencari nilai maksimum dan minimum dengan membandingkan setiap elemen array. Setelah perhitungan selesai, program menampilkan hasil berupa rata-rata kelas, nilai tertinggi, dan nilai terendah secara rapi di layar.

## Guided 4

```
#include <iostream>

using namespace std;

void tampilkanPola(int n) {
    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= i; j++) {
            cout << j << " ";
        }
        cout << endl;
    }
}
```

```

}

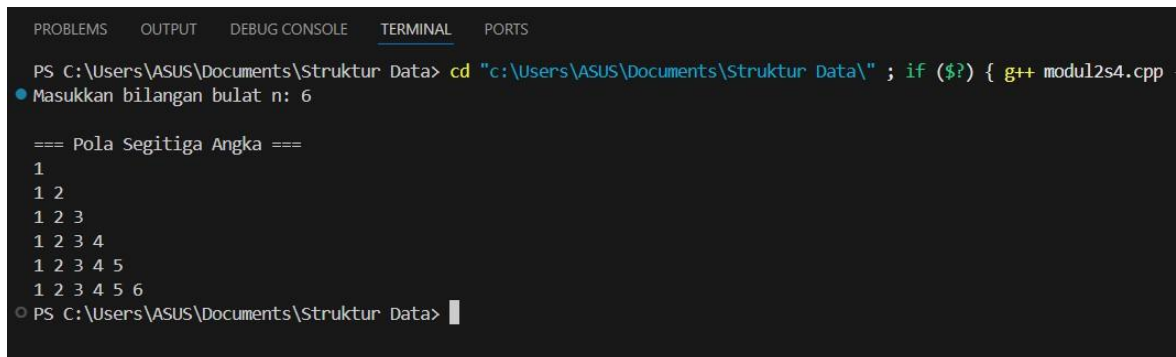
int main() {
    int n;
    cout << "Masukkan bilangan bulat n: ";
    cin >> n;

    cout << "\n=== Pola Segitiga Angka ===\n";
    tampilkanPola(n);

    return 0;
}

```

### Screenshots Output



```

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS
PS C:\Users\ASUS\Documents\Struktur Data> cd "c:\Users\ASUS\Documents\Struktur Data\" ; if ($?) { g++ modul2s4.cpp
Masukkan bilangan bulat n: 6

=== Pola Segitiga Angka ===
1
1 2
1 2 3
1 2 3 4
1 2 3 4 5
1 2 3 4 5 6
PS C:\Users\ASUS\Documents\Struktur Data>

```

### Deskripsi:

Program C++ ini berfungsi untuk **menampilkan pola segitiga angka** berdasarkan bilangan bulat n yang dimasukkan pengguna. Setelah pengguna memasukkan nilai n, program akan memanggil fungsi tampilkanPola() yang menggunakan dua perulangan bersarang (for). Perulangan luar menentukan jumlah baris, sedangkan perulangan dalam menampilkan deretan angka dari 1 hingga nilai baris saat itu. Hasilnya adalah pola segitiga angka yang dimulai dari angka 1 di baris pertama, kemudian bertambah satu angka di setiap baris berikutnya hingga mencapai nilai n

### 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.