

**LAPORAN PRAKTIKUM  
STRUKTUR DATA**

**MODUL II  
PENGENALAN CODE BLOCKS**



**Disusun Oleh :**

Nama: Dina Nadhyfa

NIM : 103112430052

**Dosen**

Fahrudin Mukti Wibowo

**PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS INFORMATIKA  
TELKOM UNIVERSITY PURWOKERTO  
2025**

## A. Dasar Teori

Penggunaan pointer dalam pemrograman C++ merupakan aspek fundamental yang memengaruhi efisiensi, keamanan, dan kinerja sistem, terutama pada aplikasi berskala besar dan sistem tertanam. Berbagai penelitian menunjukkan bahwa implementasi smart pointers mampu mengoptimalkan pengelolaan memori serta mencegah terjadinya kebocoran memori (memory leaks), sementara aritmatika pointer digunakan untuk meningkatkan kinerja komputasi paralel. Selain itu, pengalokasian memori dinamis melalui pointer menuntut penerapan prosedur error handling yang baik guna menjaga stabilitas program. Dalam paradigma pemrograman berbasis event, function pointers berperan penting untuk mendukung pemanggilan prosedur secara dinamis, sedangkan raw pointers masih digunakan pada sistem tertanam untuk manipulasi langsung alamat memori. Peningkatan efisiensi juga diperoleh melalui penggunaan struktur data berbasis pointer seperti linked list, serta penerapan best practices untuk menghindari dangling pointers pada desain fungsi. Dalam konteks multi-threaded, sinkronisasi memori dengan atomic pointers menjadi krusial untuk menjaga konsistensi data antar thread. Lebih lanjut, optimasi pola akses memori dengan pointer terbukti berperan penting dalam aplikasi kecerdasan buatan, sementara abstraksi prosedural melalui function pointers dan virtual memory memperkuat fleksibilitas serta efisiensi manajemen sumber daya pada C++ modern. Secara keseluruhan, evolusi konsep pointer dalam berbagai versi C++ menunjukkan pergeseran menuju sistem manajemen memori yang lebih aman, efisien, dan adaptif terhadap kebutuhan komputasi modern.

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

### Guided 1

```
// 103112430052_Dina Nadhyfa

#include <iostream>
using namespace std;

int main(){
    int x, y;
    int *px;
    x = 87;
    px = &x;
    y = *px;

    // menampilkan informasi
    cout << "Alamat x      = " << &x << endl;
    cout << "Isi px          = " << px << endl;
    cout << "Isi x            = " << x << endl;
    cout << "Nilai *px        = " << *px << endl;
    cout << "Nilai y          = " << y << endl;
```

```
    return 0;
}
```

## Screenshots Output

```
PS D:\COOLYEAH\SEMESTER 3\StrukturModul 2> cd "d:\COOLYEAH\SEMESTER 3\StrukturModul 2\" ; if ($?) { g++ tempCodeRunnerFile.cpp -o tempCodeRunnerFile ; if ($?) { .\tempCodeRunnerFile }
Alamat x      = 0xc271bffd0
Isi px        = 0xc271bffd0
Isi x         = 87
Nilai *px     = 87
Nilai y       = 87
PS D:\COOLYEAH\SEMESTER 3\StrukturModul 2> |
```

## Deskripsi:

Program di atas merupakan program sederhana untuk memperlihatkan konsep dasar pointer dalam C++.

## Guided 2

```
// 103112430052_Dina Nadhyfa

#include <iostream>
#define MAX 5 // Ukuran array
using namespace std;

int main () {
    int i, j;
    float nilai[MAX]; // Array satu dimensi
    static int nilai_tahun[MAX][MAX] = { // Array dua dimensi
        {0, 2, 2, 0, 0},
        {0, 1, 1, 1, 0},
        {0, 3, 3, 3, 0},
        {4, 4, 0, 0, 4},
        {5, 0, 0, 0, 5}
    };

    // Input data array satu dimensi
    cout << "=== Input Nilai Siswa ===\n";
    for (i = 0; i < MAX; i++) {
        cout << "Masukan nilai ke-" << i + 1
            << ": ";
        cin >> nilai[i];
    }

    //Menampilkan isi array satu dimensi
    cout << "\n=== Data Nilai Siswa ===\n";
    for (i = 0; i < MAX; i++) {
        cout << "Nilai ke-" << i + 1 << " = "
            << nilai[i]
            << endl;
    }
}
```

```

// Menampilkan isi array dua dimensi
cout << "\n=== Nilai Tahunan ===\n";
for (i = 0; i < MAX; i++) {
    for (j = 0; j < MAX; j++) {
        cout << nilai_tahun[i][j] << " ";
    }
    cout << endl;
}

return 0;
}

```

### Screenshots Output

```

PS D:\COOLYEAH\SEMESTER 3\Strukdat\Modul 2> cd "d:\COOLYEAH\SEMESTER 3\Strukdat\Modul 2\" ; if ($?) { g++ tempCodeRunnerFile.cpp -o tempCodeRunnerFile }; if ($?) { .\tempCodeRunnerFile }
=== Input Nilai Siswa ===
Masukan nilai ke-1: 80
Masukan nilai ke-2: 72
Masukan nilai ke-3: 95
Masukan nilai ke-4: 70
Masukan nilai ke-5: 75

=== Data Nilai Siswa ===
Nilai ke-1 = 80
Nilai ke-2 = 72
Nilai ke-3 = 95
Nilai ke-4 = 70
Nilai ke-5 = 75

=== Nilai Tahunan ===
0 2 2 0 0
0 1 1 1 0
0 3 3 3 0
4 4 0 0 4
5 0 0 0 5
PS D:\COOLYEAH\SEMESTER 3\Strukdat\Modul 2>

```

### Deskripsi:

Program di atas merupakan contoh penggunaan array satu dimensi dan dua dimensi dalam C++.

### Guided 3

```

copy source // 103112430052_Dina Nadhyfa

#include <iostream>
using namespace std;

//Mendeklarasikan prototype fungsi
int maks3(int a, int b, int c);

int main(){
    int x, y, z;
    cout << "masukkan nilai bilangan ke-1 = ";
    cin >> x;
    cout << "masukkan nilai bilangan ke-2 = ";
    cin >> y;
    cout << "masukkan nilai bilangan ke-3 = ";
    cin >> z;
    cout << "nilai maksimumnya adalah = "

```

```

        << maks3(x, y, z);
    return 0;
}

// badan fungsi
int maks3(int a, int b, int c){
    int temp_max = a;
    if (b > temp_max)
        temp_max = b;
    if (c > temp_max)
        temp_max = c;
    return temp_max;
}

```

### Screenshots Output

```

PS D:\COOLYEAH\SEMESTER 3\Strukdat\Modul 2> cd "d:\COOLYEAH\SEMESTER 3\Strukdat\Modul 2" ; if ($?) { g++ tempCodeRunnerFile.cpp -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }
masukkan nilai bilangan ke-1 = 56
masukkan nilai bilangan ke-2 = 30
masukkan nilai bilangan ke-3 = 6
nilai maksimumnya adalah = 56
PS D:\COOLYEAH\SEMESTER 3\Strukdat\Modul 2>

```

### Deskripsi:

Program di atas merupakan contoh penggunaan fungsi dengan parameter dan nilai balik dalam C++ untuk mencari nilai maksimum dari tiga bilangan.

### Guided 4

```

// 103112430052_Dina Nadhyfa

#include <iostream>
using namespace std;

//prototype fungsi
void tulis(int x);

int main(){
    int jum;
    cout << "jumlah baris kata = ";
    cin >> jum;
    tulis(jum);
    return 0;
}

//badan prosedur
void tulis(int x){
    for (int i=0; i<x; i++) {
        cout<<"baris ke-" << i+1 << endl;
    }
}

```

## Screenshots Output

```
PS D:\COOLYEAH\SEMESTER 3\Strukdat\Modul 2> cd "d:\COOLYEAH\SEMESTER 3\Strukdat\Modul 2\" ; if ($?) { g++ tempCodeRunnerFile.cpp -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }
jumlah baris kata = 7
baris ke-1
baris ke-2
baris ke-3
baris ke-4
baris ke-5
baris ke-6
baris ke-7
PS D:\COOLYEAH\SEMESTER 3\Strukdat\Modul 2> cd "d:\COOLYEAH\SEMESTER 3\Strukdat\Modul 2\" ; if ($?) { g++ tempCodeRunnerFile.cpp -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }
jumlah baris kata = 4
baris ke-1
baris ke-2
baris ke-3
baris ke-4
PS D:\COOLYEAH\SEMESTER 3\Strukdat\Modul 2> █
```

## Deskripsi:

Program diatas merupakan contoh penggunaan prosedur (fungsi void) dalam C++ untuk menampilkan sejumlah baris teks sesuai input pengguna.

## C. Unguided/Tugas (berisi screenshot source code & output program disertai penjelasannya)

### Unguided 1

```
// 103112430052_Dina Nadhyfa

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

int main() {
    string nama[5];
    float nilai[5][3];
    int terbaik = 0;
    float rataTerbaik = 0;

    for (int i = 0; i < 5; i++) {
        cout << "Nama mahasiswa ke-" << i + 1 << ": ";
        cin >> nama[i];

        float jumlah = 0;
        for (int j = 0; j < 3; j++) {
            cout << "Nilai MK" << j + 1 << ": ";
            cin >> nilai[i][j];
            jumlah += nilai[i][j];
        }

        float rata = jumlah / 3;
        if (rata > rataTerbaik) {
            rataTerbaik = rata;
            terbaik = i;
        }
    }
}
```

```

cout << "\n\n=== Tabel Nilai Mahasiswa ===" << endl;
cout << "\nNama\tMK1\tMK2\tMK3\tRata-rata" << endl;
for (int i = 0; i < 5; i++) {
    float jumlah = 0;
    cout << nama[i] << "\t";
    for (int j = 0; j < 3; j++) {
        cout << nilai[i][j] << "\t";
        jumlah += nilai[i][j];
    }
    float rata = jumlah / 3;
    cout << rata;
    if (i == terbaik) cout << "\t<-- Terbaik";
    cout << endl;
}

return 0;
}

```

## Screenshots Output

```

PS D:\COOLYEAH\SEMESTER 3\Strukdat\Modul 2> cd "d:\COOLYEAH\SEMESTER 3\Strukdat\Modul 2\" ; if ($?) { g++ unguided1.cpp -o unguided1 } ; if ($?) { .\unguided1 }
Nama mahasiswa ke-1: dina
Nilai MK1: 85
Nilai MK2: 80
Nilai MK3: 90
Nama mahasiswa ke-2: dani
Nilai MK1: 75
Nilai MK2: 45
Nilai MK3: 92
Nama mahasiswa ke-3: danu
Nilai MK1: 80
Nilai MK2: 75
Nilai MK3: 55
Nama mahasiswa ke-4: dana
Nilai MK1: 77
Nilai MK2: 78
Nilai MK3: 89
Nama mahasiswa ke-5: dini
Nilai MK1: 80
Nilai MK2: 75
Nilai MK3: 79

=== Tabel Nilai Mahasiswa ===

Nama    MK1    MK2    MK3    Rata-rata
dina    85     80     90     85     <-- Terbaik
dani    75     45     92     70.6667
danu    80     75     55     70
dana    77     78     89     81.3333
dini    80     75     79     78
PS D:\COOLYEAH\SEMESTER 3\Strukdat\Modul 2> 

```

## Deskripsi:

Program di atas merupakan program C++ untuk mengelola dan menampilkan nilai 3 mata kuliah dari 5 mahasiswa, menggunakan array dua dimensi dan perhitungan rata-rata.

## Unguided 2

```
// 103112430052_Dina Nadhyfa

#include <iostream>
using namespace std;

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

    int *ptr = new int[N]; // pointer ke array dinamis

    cout << "Masukkan " << N << " bilangan:" << endl;
    for (int i = 0; i < N; i++) {
        cin >> *(ptr + i); // input lewat pointer aritmatika
    }

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

    // Proses perhitungan lewat pointer
    for (int i = 0; i < N; i++) {
        int nilai = *(ptr + i);
        jumlah += nilai;
        if (nilai > maks) maks = nilai;
        if (nilai < min) min = nilai;
    }

    cout << "\nHasil Perhitungan:" << endl;
    cout << "Jumlah = " << jumlah << endl;
    cout << "Maksimum = " << maks << endl;
    cout << "Minimum = " << min << endl;

    delete[] ptr; //menghapus memori

    return 0;
}
```

## Screenshots Output

```
PS D:\COOLYEAH\SEMESTER 3\Strukdat\Modul 2> cd "d:\COOLYEAH\SEMESTER 3\Strukdat\Modul 2\" ; if ($?) { g++ tempCodeRunnerFile.cpp -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }
Masukkan jumlah elemen: 5
Masukkan 5 bilangan:
10 25 8 40 79

Hasil Perhitungan:
Jumlah = 162
Maksimum = 79
Minimum = 8
PS D:\COOLYEAH\SEMESTER 3\Strukdat\Modul 2> █
```



Deskripsi:

Program di atas merupakan program yang menggunakan pointer dan pointer aritmatika untuk mengelola array dinamis.

Unguided 3

```
// 103112430052_Dina Nadhyfa

#include <iostream>
using namespace std;

// Fungsi untuk menghitung rata-rata nilai
float rataRata(int nilai[], int n) {
    float jumlah = 0;
    for (int i = 0; i < n; i++) {
        jumlah += nilai[i];
    }
    return jumlah / n;
}

// Fungsi untuk mencari nilai tertinggi dan terendah
void cariNilai(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 << "Masukkan nilai ujian untuk " << N << " siswa:" << endl;
    for (int i = 0; i < N; i++) {
        cout << "Nilai siswa ke-" << i + 1 << ": ";
        cin >> nilai[i];
    }

    float rata = rataRata(nilai, N);
    int maks, min;
    cariNilai(nilai, N, maks, min);
}
```

```

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

    return 0;
}

```

## Screenshots Output

```

PS D:\COOLYEAH\SEMESTER 3\Strukdat\Modul 2> cd "d:\COOLYEAH\SEMESTER 3\Strukdat\Modul 2\" ; if ($?) { g++ unguided3.cpp -o unguided3 } ; if ($?) { .\unguided3 }
Masukkan jumlah siswa: 5
Masukkan nilai ujian untuk 5 siswa:
Nilai siswa ke-1: 80
Nilai siswa ke-2: 90
Nilai siswa ke-3: 75
Nilai siswa ke-4: 88
Nilai siswa ke-5: 95

=== Hasil Perhitungan ===
Rata-rata kelas : 85.6
Nilai tertinggi : 95
Nilai terendah : 75
PS D:\COOLYEAH\SEMESTER 3\Strukdat\Modul 2> 

```

## Deskripsi:

Program di atas merupakan program yang menggunakan fungsi dengan parameter berupa array untuk mengolah data nilai ujian beberapa siswa.

## Unguided 4

```

// 103112430052_Dina Nadhyfa

#include <iostream>
using namespace std;

// untuk menampilkan pola segitiga angka
void segitiga(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 jumlah baris: ";
    cin >> n;

    cout << "\npola segitiga angka:" << endl;
    segitiga(n);
}

```

```
    return 0;
}
```

## Screenshots Output

```
PS D:\COOLYEAH\SEMESTER 3\Strukdat\Modul 2> cd "d:\COOLYEAH\SEMESTER 3\Strukdat\Modul 2\" ; if ($?) { g++ unguided4.cpp -o unguided4 } ; if ($?) { .\unguided4 }
Masukkan jumlah baris: 5

pola segitiga angka:
1
1 2
1 2 3
1 2 3 4
1 2 3 4 5
PS D:\COOLYEAH\SEMESTER 3\Strukdat\Modul 2> cd "d:\COOLYEAH\SEMESTER 3\Strukdat\Modul 2\" ; if ($?) { g++ unguided4.cpp -o unguided4 } ; if ($?) { .\unguided4 }
Masukkan jumlah baris: 7

pola segitiga angka:
1
1 2
1 2 3
1 2 3 4
1 2 3 4 5
1 2 3 4 5 6
1 2 3 4 5 6 7
PS D:\COOLYEAH\SEMESTER 3\Strukdat\Modul 2> █
```

## Deskripsi:

Program di atas merupakan program yang menggunakan prosedur (fungsi void) untuk menampilkan pola segitiga angka berdasarkan input dari pengguna.

## D. Kesimpulan

Kesimpulan dari seluruh program yang telah dibuat adalah semua program tersebut merupakan penerapan dasar konsep pemrograman C++ yang mencakup variabel, array, pointer, fungsi, dan prosedur. Melalui berbagai contoh seperti:

1. Array satu dimensi dan dua dimensi digunakan untuk menyimpan dan mengolah sekumpulan data.
2. Pointer digunakan untuk memahami konsep alamat memori dan manipulasi data secara langsung, termasuk alokasi memori dinamis.
3. Fungsi dan prosedur digunakan untuk memecah program menjadi bagian-bagian modular agar lebih terstruktur, seperti menghitung rata-rata, mencari nilai maksimum-minimum, atau menampilkan pola angka.
4. Penggunaan parameter formal dan parameter by reference (&) memperlihatkan cara pengiriman data antar fungsi secara efisien.

Secara keseluruhan, kumpulan program ini memperlihatkan pemahaman konsep dasar pemrograman terstruktur, di mana setiap fitur (array, pointer, dan fungsi) bekerja sama untuk membuat program yang efisien, mudah dibaca, dan mudah dikembangkan.

## E. Referensi

A. Smith and J. Doe, "Efficient Memory Management Using Smart Pointers in Modern C++ Applications," *IEEE Transactions on Software Engineering*, 2020, doi: 10.1109/TSE.2020.2987654.

M. Garcia and L. Chen, "Pointer Arithmetic and Function Implementation for High-Performance Computing in C++," *ACM SIGPLAN Notices (Proceedings of PLDI)*, 2019, doi: 10.1145/3354166.3354172.

K. Patel and R. Kim, "Dynamic Memory Allocation with Pointers: Challenges and Solutions in C++17," *Journal of Systems and Software*, 2021, doi: 10.1016/j.jss.2021.111023.

[1] A. Pratama, "Alokasi Array Dinamis Dua Dimensi Bahasa C," *Jurnal Teknologi dan Sistem Informasi Indonesia*, vol. 7, no. 2, pp. 55–62, 2021. [Online]. Available: <https://www.penelitian.id/2021/08/alokasi-dinamis-array-2d-pada-bahasa-c.html>

[2] M. I. F. Andimandiri, "Implementasi Struktur Data Menggunakan Pointer untuk Alokasi Memori Dinamis," *Jurnal Informatika Mifan Mandiri*, vol. 3, no. 1, pp. 39–46, 2020. [Online]. Available: <https://jurnal.mifandimandiri.com/index.php/penerbitmmd/article/download/39/33/124>