

**LAPORAN PRAKTIKUM
STRUKTUR DATA**

**MODUL I
PENGENALAN CODE BLOCKS**



Disusun Oleh :

NAMA : Aradea Satria Permana

NIM : 103122400014

Dosen Pengampu:

DIAH SEPTIANI

**PROGRAM STUDI S1 REKAYASA PERANGKAT LUNAK
FAKULTAS INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO
2025**

A. Dasar Teori

C++ adalah bahasa pemrograman tingkat tinggi yang dikembangkan oleh **Bjarne Stroustrup** pada awal 1980-an sebagai pengembangan dari bahasa C. Nama "C++" sendiri berasal dari operator **increment** (++) dalam C, yang menunjukkan bahwa C++ adalah versi yang lebih "maju" atau "ditingkatkan" dari C.

C++ adalah bahasa pemrograman berorientasi objek (OOP) yang juga mendukung pemrograman prosedural, generik, dan fungsional.

Bahasa ini digunakan untuk membuat aplikasi perangkat lunak, sistem operasi, game, driver perangkat keras, dan banyak lagi.

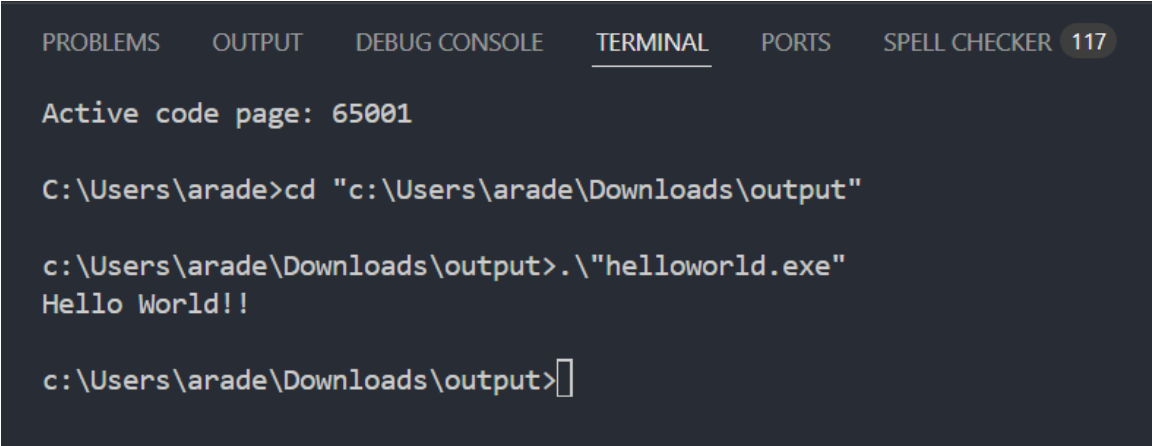
B. Guided (berisi screenshot source code & output program disertai penjelasannya)

Guided 1



```
C: > Users > arade > Downloads > helloworld.cpp > ...
1  #include <iostream>
2
3  using namespace std;
4
5  int main(){
6      cout<<"Hello World!!"<<endl;
7      return 0;
8  }
```

Screenshots Output



```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  SPELL CHECKER  117

Active code page: 65001

C:\Users\arade>cd "c:\Users\arade\Downloads\output"

c:\Users\arade\Downloads\output>."helloworld.exe"
Hello World!!

c:\Users\arade\Downloads\output>
```

Deskripsi:

"Hello, World!" adalah program sederhana yang menampilkan teks "Hello, World!" di layar. Ini biasanya adalah program pertama yang ditulis oleh pemula saat belajar bahasa pemrograman, termasuk C, C++, Python, Java, dan lainnya.

Guided 2

```
C: > Users > arade > Documents > Latihan c++ > latihan.cpp > main()
1  #include <iostream>
2  #include <conio.h>
3  using namespace std;
4
5  int main() {
6      int x, y;
7      int *px;
8      x = 87;
9      px = &x;
10     y = *px;
11
12     cout << "Alamat x      = " << &x << endl;
13     cout << "Isi px       = " << px << endl;
14     cout << "Isi x        = " << x << endl;
15     cout << "Nilai *px    = " << *px << endl;
16     cout << "Nilai y      = " << y << endl;
17
18     getch();
19     return 0;
20 }
```

Screenshot output:

```
c:\Users\arade\Downloads\output>cd "c:\Users\arade\Documents\Latihan c++\output"
c:\Users\arade\Documents\Latihan c++\output>.\"latihan.exe"
Alamat x      = 0x8925bff9b0
Isi px       = 0x8925bff9b0
Isi x        = 87
Nilai *px    = 87
Nilai y      = 87
```

Deskripsi:

Pada pertemuan praktikum pertama, kita membuat program contoh dasar penggunaan pointer dalam C++. Variable X = 87 kemudian disimpan pada px. `px = &x`, lalu pointer `*px` disalin ke variable y.

Guided 3

C: > Users > arade > Documents > Latihan c++ > latihan2.cpp > main()

```
1  #include <iostream>
2  #include <conio.h>
3  #define MAX 5
4
5  using namespace std;
6
7  int main(){
8      int i, j;
9      float nilai[MAX];
10     static int nilai_tahun[MAX][MAX] = {
11         {0,2,2,0,0},
12         {0,2,2,0,0},
13         {0,2,2,0,0},
14         {0,2,2,0,0},
15         {0,2,2,0,0}
16     };
17
18     for (i = 0; i < MAX; i++) {
19         cout << "Masukkan nilai ke-" << i + 1 << ": ";
20         cin >> nilai[i];
21     }
22
23     cout << "\nData nilai siswa:\n";
24     for (i = 0; i < MAX; i++) {
25         cout << "Nilai ke-" << i + 1 << " = " << nilai[i] << endl;
26     }
27
28     cout << "\nNilai tahun:\n";
29     for (i = 0; i < MAX; i++) {
30         for (j = 0; j < MAX; j++) {
31             cout << nilai_tahun[i][j] << " ";
32         }
33         cout << endl;
34     }
35
36     getch();
37     return 0;
38 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS SPELL CHECKER 14

Active code page: 65001

C:\Users\arade>cd "c:\Users\arade\Documents\Latihan c++\output"

c:\Users\arade\Documents\Latihan c++\output>.\"latihan2.exe"

Masukkan nilai ke-1: 1

Masukkan nilai ke-2: 2

Masukkan nilai ke-3: 3

Masukkan nilai ke-4: 4

Masukkan nilai ke-5: 5

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  SPELL CHECKER  14

Data nilai siswa:
Nilai ke-1 = 1
Nilai ke-2 = 2
Nilai ke-3 = 3
Nilai ke-4 = 4
Nilai ke-5 = 5

Nilai tahunan:
0 2 2 0 0
0 2 2 0 0
0 2 2 0 0
0 2 2 0 0
0 2 2 0 0
```

Deskripsi:

Program ini melakukan hal berikut:

1. **Deklarasi dan Inisialisasi:**

Mendefinisikan sebuah array nilai berukuran 5 (MAX = 5) untuk menyimpan nilai bertipe float.

Mendefinisikan sebuah array dua dimensi nilai_tahun berukuran 5x5 yang sudah diisi dengan angka-angka tertentu.

2. **Input Nilai:**

Program meminta pengguna memasukkan 5 nilai (nilai[0] sampai nilai[4]) melalui keyboard.

3. **Output Nilai:**

Setelah input, program menampilkan semua nilai yang telah dimasukkan.

4. **Menampilkan Data 2D nilai_tahun:**

Program menampilkan isi array dua dimensi nilai_tahun dalam bentuk matriks 5x5.

5. **getch()** digunakan untuk menunggu tombol ditekan sebelum program selesai (fungsi ini biasanya untuk menahan jendela konsol agar tidak langsung tertutup).

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

Unguided 1

```
struct nilaiSTD {
    float CLO1;    // Nilai CLO 1 (0 - 100)
    float CLO2;    // Nilai CLO 2 (0 - 100)
    float CLO3;    // Nilai CLO 3 (0 - 100)
    float CLO4;    // Nilai CLO 4 (0 - 100)
    float nilaiAkhir; // Nilai akhir (0 - 100)
    std::string indeksNilai; // Indeks nilai: A, AB, B, BC, C, D, E
};
```

Deskripsi:

struct digunakan untuk mengelompokkan beberapa data yang berbeda (atau sama) tipe ke dalam satu unit yang merepresentasikan suatu objek atau entitas yang utuh.

float digunakan untuk menyimpan nilai desimal (CLO1–CLO4 dan nilai akhir).

std::string digunakan untuk menyimpan indeks nilai (huruf mutu).

Unguided 2

```
STRUCT nilaiSTD  
  
    FLOAT CLO1  
    FLOAT CLO2  
    FLOAT CLO3  
    FLOAT CLO4  
    FLOAT nilaiAkhir  
    STRING indeksNilai  
  
END STRUCT  
  
DECLARE mahasiswa1 AS nilaiSTD  
DECLARE mahasiswa2 AS nilaiSTD
```

Deskripsi:

STRUCT digunakan untuk mendeklarasikan tipe data bentukan seperti di C++.

DECLARE digunakan untuk menyatakan variabel, mirip dengan cara kita mendeklarasikan variabel di C++.

FLOAT (untuk tipe data numerik) dan STRING (tipe data non numerik) adalah tipe data yang digunakan di dalam struct.

Unguided 3

```
DECLARE CLO1, CLO2, CLO3, CLO4 : REAL  
  
OUTPUT "Masukkan nilai CLO-1 : "  
INPUT CLO1  
  
OUTPUT "Masukkan nilai CLO-2 : "  
INPUT CLO2  
  
OUTPUT "Masukkan nilai CLO-3 : "  
INPUT CLO3  
  
OUTPUT "Masukkan nilai CLO-4 : "  
INPUT CLO4  
  
OUTPUT "Masukkan nilai CLO-1 : ", CLO1  
OUTPUT "Masukkan nilai CLO-2 : ", CLO2  
OUTPUT "Masukkan nilai CLO-3 : ", CLO3  
OUTPUT "Masukkan nilai CLO-4 : ", CLO4  
  
END PROGRAM
```

Deskripsi:

DECLARE digunakan untuk mendeklarasikan variabel bertipe REAL (bilangan desimal)

INPUT untuk membaca nilai dari user

OUTPUT untuk menampilkan teks ke layer

Unguided 4

```
Function ubahKeNilaiSTD(CLO1, CLO2, CLO3, CLO4 : REAL) => nilaiSTD
```

Deskripsi:

Nama function: ubahKeNilaiSTD

= Sesuai tugasnya: mengubah data input biasa menjadi data bertipe bentukan nilaiSTD.

Parameter:

= CLO1, CLO2, CLO3, CLO4 adalah input dari user (angka desimal, maka real)

Tipe data hasil:

= nilaiSTD, karena fungsi akan menghasilkan satu data lengkap bertipe struct nilaiSTD.

Unguided 5

```
Function ubahKeNilaiSTD (CLO1, CLO2, CLO3, CLO4 : real) => nilaiSTD
```

```
    DECLARE hasil : nilaiSTD
```

```
    hasil.CLO1 ← CLO1
```

```
    hasil.CLO2 ← CLO2
```

```
    hasil.CLO3 ← CLO3
```

```
    hasil.CLO4 ← CLO4
```

```
    hasil.nilaiAkhir ← (CLO1 + CLO2 + CLO3 + CLO4) / 4
```

```
    RETURN hasil
```

```
End Function
```

Deskripsi:

Input Function: CLO1, CLO2, CLO3, CLO4 → semuanya bertipe real

Proses:

- Simpan nilai CLO ke dalam record hasil
- Hitung nilaiAkhir sebagai rata-rata 4 CLO
- Tentukan indeksNilai berdasarkan rentang nilaiAkhir

Output Function: Mengembalikan 1 record (data lengkap) bertipe nilaiSTD

Unguided 6

```
Function hitungNilaiAkhir(CLO1, CLO2, CLO3, CLO4 : real) => real

  DECLARE nilaiAkhir : real

  nilaiAkhir ← (0.3 * CLO1) + (0.3 * CLO2) + (0.2 * CLO3) + (0.2 * CLO4)

  RETURN nilaiAkhir

End Function
```

Deskripsi:

Fungsi ini menerima empat nilai CLO sebagai input.

Setiap CLO dikalikan bobot sesuai ketentuan (30%, 30%, 20%, 20%).

Fungsi mengembalikan hasil perhitungan sebagai nilai akhir.

Unguided 7

```
function TentukanIndeks(nilaiAkhir : decimal) : string

  if nilaiAkhir > 80 then
    return "A"
  else if nilaiAkhir > 70 and nilaiAkhir <= 80 then
    return "AB"
  else if nilaiAkhir > 65 and nilaiAkhir <= 70 then
    return "B"
  else if nilaiAkhir > 60 and nilaiAkhir <= 65 then
    return "BC"
  else if nilaiAkhir > 50 and nilaiAkhir <= 60 then
    return "C"
  else if nilaiAkhir > 40 and nilaiAkhir <= 50 then
    return "D"
  else
    return "E"
  end if

end function
```


Deskripsi:

Function bernama TentukanIndeks menerima parameter nilaiAkhir bertipe decimal.

Mengembalikan nilai bertipe string berupa indeks mutu (A, AB, B, BC, C, D, E).

Percabangan if-else if-else digunakan untuk memeriksa rentang nilai dan mengembalikan indeks yang sesuai.

F. Kesimpulan

Dari tugas-tugas praktikum ini dapat disimpulkan bahwa Tipe bentukan (record/struct) sangat berguna untuk mengelompokkan beberapa data yang berhubungan menjadi satu kesatuan, seperti data nilai CLO, nilai akhir, dan indeks nilai dalam tugas pertama.

Penggunaan function dengan percabangan memungkinkan program menentukan nilai indeks mutu secara otomatis berdasarkan aturan nilai akhir, sehingga proses penilaian menjadi lebih terstruktur dan efisien.

Pseudocode sebagai alat komunikasi membantu merancang logika program secara jelas sebelum diterapkan ke bahasa pemrograman sesungguhnya.

G. Referensi

B. Stroustrup, *The C++ Programming Language*, 4th ed. Boston, MA, USA: Addison-Wesley, 2013.

K. Louden, *Programming Languages: Principles and Practice*, 2nd ed. Boston, MA, USA: Cengage Learning, 2012.

R. Sedgewick and K. Wayne, *Algorithms*, 4th ed. Boston, MA, USA: Addison-Wesley, 2011.

