

**LAPORAN PRAKTIKUM
STRUKTUR DATA**

**MODUL I
PENGENALAN CODE BLOCKS**



Disusun Oleh :
FATIKHAH SUKMA ARTI
103122400019

Dosen
DIAH SEPTIANI S.Kom M.Cs

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

A. Dasar Teori

Struktur data merupakan cara menyimpan, pengorganisasian, dan pengaturan data agar dapat digunakan secara efisien.

Dalam C++, kita dapat membuat tipe bentukan (struct) untuk mengelompokkan beberapa data dalam satu wadah.

Selain itu, fungsi digunakan untuk membagi program menjadi bagian kecil yang mudah dipahami, diuji, dan digunakan kembali.

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

Guided 1

```
1  #include <iostream>
2  #include <conio.h>
3
4  using namespace std;
5
6  int main(){
7      int x,y;
8      int *px;
9      x = 87;
10     px = &x;
11     y = *px;
12
13     cout<<"Alamat x = "<< &x << endl;
14     cout<<"Isi px = "<< px << endl;
15     cout<<"Isi x = "<< x << endl;
16     cout<<"Nilai *px = "<< *px << endl;
17     cout<<"Nilai y = "<< y << endl;
18
19     getch();
20     return 0;
21 }
```

```

G+ contoh2.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         {0,1,1,1,0},
13         {0,3,3,3,0},
14         {4,4,0,0,4},
15         {5,0,0,0,5}
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 tahunan:\n";
29     for ( i = 0; i < MAX; i++){
30         for ( i = 0; i < MAX; i++){
31             cout << nilai_tahun[i][j] << " ";
32         }
33         cout << endl;
34     }
35
36     getch();
37     return 0;
38 }

```

Screenshots Output

```

Alamat x = 0x5ffec0
Isi px = 0x5ffec0
Isi x = 87
Nilai *px = 87
Nilai y = 87

```

```
Masukkan nilai ke-1: 4
Masukkan nilai ke-2: 3
Masukkan nilai ke-3: 2
Masukkan nilai ke-4: 5
Masukkan nilai ke-5: 5
```

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

Deskripsi:

Mereview materi pemograman algoritma

D. Unguied/Tugas (berisi screenshot source code & output program disertai penjelasannya)

Unguied 1

```
review.cpp x
review.cpp > main()
1  #include <iostream>
2  #include <string>
3  using namespace std;
4
5  struct nilaiSTD {
6      float clo1, clo2, clo3, clo4;
7      float nilaiAkhir;
8      string indeks;
9  };
10
11 float hitungNilaiAkhir(float c1, float c2, float c3, float c4){
12     return (0.3 * c1) + (0.3 * c2) + (0.2 * c3) + (0.2 * c4);
13 }
14
15 string tentukanIndeks(float nilai){
16     if (nilai >= 85) return "A";
17     else if (nilai >= 70) return "AB";
18     else if (nilai >= 65) return "B";
19     else if (nilai >= 65) return "BC";
20     else if (nilai >= 55) return "C";
21     else if (nilai >= 45) return "D";
22     else return "E";
23 }
24
25 int main(){
26     nilaiSTD mhs;
27
28     cout << "Masukkan nilai CLO-1: "; cin >> mhs.clo1;
29     cout << "Masukkan nilai CLO-2: "; cin >> mhs.clo2;
30     cout << "Masukkan nilai CLO-3: "; cin >> mhs.clo3;
31     cout << "Masukkan nilai CLO-4: "; cin >> mhs.clo4;
32
33     mhs.nilaiAkhir = hitungNilaiAkhir(mhs.clo1, mhs.clo2, mhs.clo3, mhs.clo4);
34     mhs.indeks = tentukanIndeks(mhs.nilaiAkhir);
35 }
```

```

36     cout << "\n HASIL NILAI" << endl;
37     cout << "Nilai Akhir: " << mhs.nilaiAkhir << endl;
38     cout << "Indeks Nilai: " << mhs.indeks << endl;
39
40     return 0;
41 }
42

```

Screenshots Output

```

Masukkan nilai CLO-1: 90
Masukkan nilai CLO-2: 80
Masukkan nilai CLO-3: 85
Masukkan nilai CLO-4: 79

Masukkan nilai CLO-2: 80
Masukkan nilai CLO-3: 85
Masukkan nilai CLO-4: 79

Masukkan nilai CLO-4: 79

HASIL NILAI
Nilai Akhir: 83.8
Indeks Nilai: AB

```

Deskripsi:

Program menyimpan nilai CLO dalam array, menghitung nilai akhir dengan fungsi `hitungAkhir`, lalu menentukan indeks dan menampilkannya dengan format 2 desimal.

E. Kesimpulan

Dari praktikum ini, dapat disimpulkan bahwa penggunaan **struct** mempermudah penyimpanan data yang terdiri dari beberapa komponen. Dengan adanya **function**, program menjadi modular dan lebih mudah dipahami. Percabangan `if-else` digunakan untuk menentukan kategori atau indeks berdasarkan nilai yang dihitung.

F. Referensi

Kurniawan, A. (2019). *Pemrograman C++ untuk Pemula*. Andi Publisher.