

**LAPORAN PRAKTIKUM
STRUKTUR DATA**

**MODUL I
PENGENALAN CODE BLOCKS**



Nama :

Chiara Calina Devi - 103122400016

Dosen :

Diah Septiani S.Kom M.Cs

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

A. Dasar Teori

Pengenalan IDE CodeBlocks & Bahasa C++

Modul ini memperkenalkan dasar-dasar pemrograman menggunakan bahasa C++ dan lingkungan pengembangan terpadu (IDE)

CodeBlocks. Materi ini mencakup:

- **Pengenalan CodeBlocks:** Modul ini menjelaskan cara menginstal dan menggunakan CodeBlocks, termasuk cara membuat proyek baru, menulis sintaks, dan melakukan kompilasi program. Kompilasi dapat dilakukan dengan perintah Build (Ctrl+F9), Run (Ctrl+F10), atau Build and Run (F9).
 - **Dasar-dasar Pemrograman C++:**
 - **Struktur Program:** Setiap program C++ harus memiliki sebuah fungsi main() sebagai titik awal eksekusi program.
 - **Pengenal (Identifier):** Nama yang digunakan untuk variabel, konstanta, fungsi, dll. Penamaan disarankan menggunakan kata benda tunggal, dengan huruf pertama setiap kata ditulis kapital (misalnya, NamaMahasiswa).
 - **Tipe Data dan Variabel:** Tipe data dasar seperti int, float, char, dan bool dijelaskan. Variabel adalah tempat penyimpanan data yang nilainya dapat berubah selama eksekusi program.
 - **Konstanta:** Data yang nilainya tetap dan tidak dapat diubah.
 - **Input/Output (I/O):**
 - **Output:** Menggunakan cout untuk menampilkan data ke layar. Dapat juga menggunakan printf dan puts.
 - **Input:** Menggunakan cin untuk membaca data dari pengguna.
 - **Operator:** Modul ini menjelaskan berbagai jenis operator, termasuk operator aritmatika (+, -, *, /, %),
 - relasi (==, !=, >, <, >=, <=),
 - logika (&&, ||, !),
 - penugasan (=, +=, -=),
 - sizeof,
 - serta increment dan decrement.
 - **Kondisional:** Struktur yang digunakan untuk membuat percabangan program berdasarkan suatu kondisi. Modul ini membahas tiga bentuk kondisional, kemungkinan besar merujuk pada if, if-else, dan if-else if-else atau switch-case.
 - **Perulangan (Looping):** Digunakan untuk mengeksekusi blok kode berulang kali. Tiga jenis perulangan yang dibahas adalah for, while, dan do while. Perulangan for dan while akan berhenti jika kondisi tidak terpenuhi, sedangkan do while akan menjalankan kode setidaknya satu kali sebelum memeriksa kondisi.
-

B. Guided

1) source code :

```
main.cpp  prak1.cpp  prak2.cpp  prak1.exe
prak1.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 << "\nNilai Semester dari array 1 dimensi:\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 (j = 0; j < MAX; j++) {
31             cout << nilai_tahun[i][j] << " ";
32         }
33         cout << endl;
34     }
35
36     getch();
37     return 0;
38 }
```

Ouput :

```
PROBLEMS  OUTPUT  TERMINAL  DEBUG CONSOLE  PORTS  SPELL-CHECKER 16
C:\Users\ACER\Documents\strukdat>prak1
Masukkan Nilai KE-1: 2
Masukkan Nilai KE-2: 3
Masukkan Nilai KE-3: 4
Masukkan Nilai KE-4: 5
Masukkan Nilai KE-5: 6

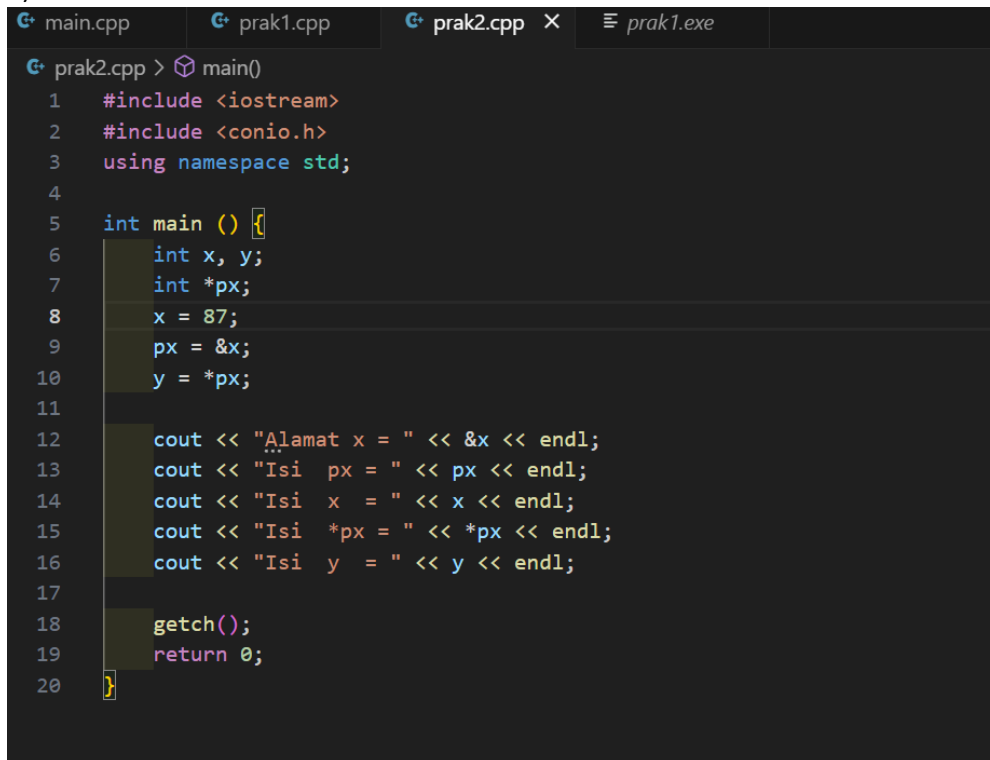
Nilai Semester dari array 1 dimensi:
Nilai ke-1 = 2
Nilai ke-2 = 3
Nilai ke-3 = 4
Nilai ke-4 = 5
Nilai ke-5 = 6

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
```

Deskripsi :

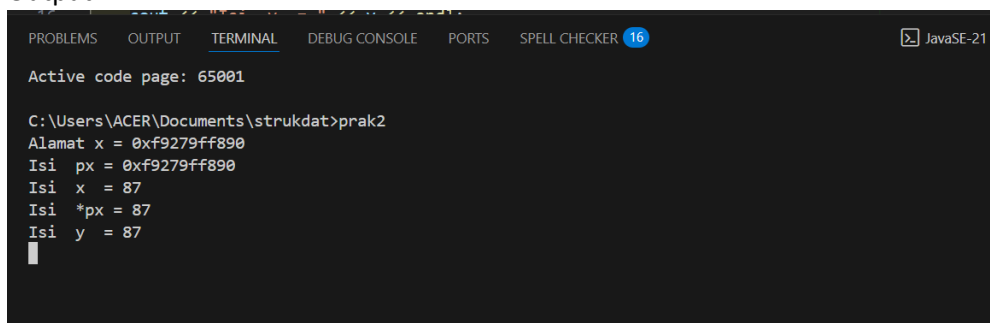
Source code ini menampilkan contoh penggunaan dasar bahasa C++ dengan menuliskan perintah *cout* untuk menampilkan teks ke layar. Program masih sederhana tanpa input pengguna, sehingga hanya fokus pada proses output. Dengan contoh ini, mahasiswa diperkenalkan bagaimana struktur *main()* bekerja dan cara mengeksekusi program pertama kali.

2) source code :



```
main.cpp  prak1.cpp  prak2.cpp  prak1.exe
prak2.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 << "Isi  *px = " << *px << endl;
16     cout << "Isi  y = " << y << endl;
17
18     getch();
19     return 0;
20 }
```

Output :



```
PROBLEMS  OUTPUT  TERMINAL  DEBUG CONSOLE  PORTS  SPELL CHECKER 16  JavaSE-21 LT
Active code page: 65001

C:\Users\ACER\Documents\strukdat>prak2
Alamat x = 0xf9279ff890
Isi  px = 0xf9279ff890
Isi  x = 87
Isi  *px = 87
Isi  y = 87
```

Deskripsi :

Pada source code ini, ditambahkan proses input menggunakan *cin* sehingga pengguna dapat memasukkan data. Data yang dimasukkan akan ditampilkan kembali dengan *cout*, sehingga mahasiswa memahami alur komunikasi antara program dan pengguna. Program ini mengajarkan dasar interaksi input-output dalam C++.

C. Unguided

1) source code:

```
soal1.cpp > nilaiSTD > CLO2
1  #include <iostream>
2  #include <string>
3  using namespace std;
4
5  struct nilaiSTD {
6      float CLO1;
7      float CLO2;
8      float CLO3;
9      float CLO4;
10     float nilaiAkhir;
11     string indeksNilai;
12 };
13
14 int main() {
15     cout << "Deklarasi struct nilaiSTD berhasil dibuat." << endl;
16     return 0;
17 }
18
```

Output :

```
Active code page: 65001

C:\Users\ACER\Documents\Tugas_ADT>soal1
Deklarasi struct nilaiSTD berhasil dibuat.

C:\Users\ACER\Documents\Tugas_ADT>g++ soal1.cpp -o soal1

C:\Users\ACER\Documents\Tugas_ADT>soal1
Deklarasi struct nilaiSTD berhasil dibuat.

C:\Users\ACER\Documents\Tugas_ADT>
```

Deskripsi :

pada soal ini dibuat sebuah *struct* bernama nilaiSTD yang berfungsi untuk menampung data nilai mahasiswa. Komponen yang disimpan meliputi nilai CLO1 sampai CLO4, nilai akhir, serta indeks nilai berbentuk huruf. Dengan adanya struct ini, data mahasiswa dapat diorganisir dengan lebih rapi karena menjadi satu kesatuan.

2) source code:

```
soal2.cpp > ...
1  include <iostream>
2  include <string>
3  using namespace std;
4
5  struct nilaiSTD {
6      float CLO1, CLO2, CLO3, CLO4, nilaiAkhir;
7      string indeksNilai;
8  };
9
10 int main() {
11     nilaiSTD mahasiswa1;
12     nilaiSTD mahasiswa2;
13
14     cout << "Dua variabel mahasiswa1 dan mahasiswa2 berhasil dideklarasikan." << endl;
15     return 0;
16
17 }
```

Output:

```
PROBLEMS  OUTPUT  TERMINAL  DEBUG CONSOLE  PORTS  SPELL CHECKER 23 JavaSE-21 L
C:\Users\ACER\Documents\Tugas_ADT>soal2
Dua variabel mahasiswa1 dan mahasiswa2 berhasil dideklarasikan.

C:\Users\ACER\Documents\Tugas_ADT>
```

Deskripsi:

Pada soal kedua, dilakukan deklarasi dua variabel dengan tipe nilaiSTD. Variabel ini nantinya dapat digunakan untuk menyimpan data dua mahasiswa berbeda. Deklarasi ini menunjukkan bagaimana tipe bentukan yang sudah dibuat bisa dipakai untuk membangun objek nyata.

3) source code:

```
soal3.cpp > main()
1  #include <iostream>
2  using namespace std;
3
4  int main() {
5      float CLO1, CLO2, CLO3, CLO4;
6
7      cout << "Masukkan nilai CLO-1 : ";
8      cin >> CLO1;
9      cout << "Masukkan nilai CLO-2 : ";
10     cin >> CLO2;
11     cout << "Masukkan nilai CLO-3 : ";
12     cin >> CLO3;
13     cout << "Masukkan nilai CLO-4 : ";
14     cin >> CLO4;
15
16     cout << "\nData yang dimasukkan:" << endl;
17     cout << "CLO-1: " << CLO1 << endl;
18     cout << "CLO-2: " << CLO2 << endl;
19     cout << "CLO-3: " << CLO3 << endl;
20     cout << "CLO-4: " << CLO4 << endl;
21
22     return 0;
23 }
24
```

Output :

```
PROBLEMS  OUTPUT  TERMINAL  DEBUG CONSOLE  PORTS  SPELL CHECKER 32

C:\Users\ACER\Documents\Tugas_ADT>soal2
Dua variabel mahasiswa1 dan mahasiswa2 berhasil dideklarasikan.

C:\Users\ACER\Documents\Tugas_ADT>soal3
Masukkan nilai CLO-1 : 80
Masukkan nilai CLO-2 : 90
Masukkan nilai CLO-3 : 80
Masukkan nilai CLO-4 : 70

Data yang dimasukkan:
CLO-1: 80
CLO-2: 90
CLO-3: 80
CLO-4: 70

C:\Users\ACER\Documents\Tugas_ADT>
```

Deskripsi :

Program ini meminta pengguna memasukkan nilai CLO1 sampai CLO4 melalui input keyboard. Data yang dimasukkan kemudian ditampilkan kembali ke layar untuk memastikan nilai tersimpan dengan benar. Hal ini melatih penggunaan *cin* dan *cout* dalam menerima serta menampilkan data.

4) source code :

```
soal4.cpp > ...
1  #include <iostream>
2  #include <string>
3  using namespace std;
4
5  struct nilaiSTD {
6      float CLO1, CLO2, CLO3, CLO4, nilaiAkhir;
7      string indeksNilai;
8  };
9
10 nilaiSTD simpanNilai(float c1, float c2, float c3, float c4);
11
12 int main() {
13     cout << "Judul function simpanNilai telah dibuat." << endl;
14     return 0;
15 }
16
```

Output:

```
PROBLEMS  OUTPUT  TERMINAL  DEBUG CONSOLE  PORTS  SPELL CHECKER 45

C:\Users\ACER\Documents\Tugas_ADT>soal4
Judul function simpanNilai telah dibuat.

C:\Users\ACER\Documents\Tugas_ADT>
```

Deskripsi :

Pada soal keempat, dibuat sebuah judul fungsi simpanNilai yang menerima input nilai CLO1 hingga CLO4. Fungsi ini didefinisikan akan menghasilkan sebuah objek nilaiSTD. Tahap ini berfungsi sebagai rancangan awal sebelum mengisi logika fungsi secara lengkap.

5)source code:

```
G+ soal5.cpp > ...
1  #include <iostream>
2  #include <string>
3  using namespace std;
4
5  struct nilaiSTD {
6      float CLO1, CLO2, CLO3, CLO4, nilaiAakhir;
7      string indeksNilai;
8  };
9
10
11  nilaiSTD simpanNilai(float c1, float c2, float c3, float c4) {
12      nilaiSTD mhs;
13      mhs.CLO1 = c1;
14      mhs.CLO2 = c2;
15      mhs.CLO3 = c3;
16      mhs.CLO4 = c4;
17      mhs.nilaiAakhir = 0;
18      mhs.indeksNilai = "";
19      return mhs;
20  }
21
22  int main() {
23      nilaiSTD mahasiswa = simpanNilai(80, 90, 70, 60);
24
25      cout << "Data tersimpan:" << endl;
26      cout << "CLO-1: " << mahasiswa.CLO1 << endl;
27      cout << "CLO-2: " << mahasiswa.CLO2 << endl;
28      cout << "CLO-3: " << mahasiswa.CLO3 << endl;
29      cout << "CLO-4: " << mahasiswa.CLO4 << endl;
30
31      return 0;
32  }
```

Source code :

```
PROBLEMS  OUTPUT  TERMINAL  DEBUG CONSOLE  PORTS  SPELL CH
C:\Users\ACER\Documents\Tugas_ADT>soal5
Data tersimpan:
CLO-1: 80
CLO-2: 90
CLO-3: 70
CLO-4: 60

C:\Users\ACER\Documents\Tugas_ADT>
```

Deskripsi :

Fungsi simpanNilai kemudian diimplementasikan dengan menyimpan nilai CLO1 sampai CLO4 ke dalam struct nilaiSTD. Nilai akhir dan indeks masih diisi dengan nilai default karena belum dihitung. Hasil dari fungsi adalah sebuah objek nilaiSTD yang sudah memiliki nilai CLO terisi.

6) source code :

```
soal6.cpp > ...
1  #include <iostream>
2  using namespace std;
3
4  float hitungNilaiAkhir(float c1, float c2, float c3, float c4) {
5      return (0.3 * c1) + (0.3 * c2) + (0.2 * c3) + (0.2 * c4);
6  }
7
8  int main() {
9      float c1=80, c2=90, c3=70, c4=60;
10     float nilaiAkhir = hitungNilaiAkhir(c1, c2, c3, c4);
11
12     cout << "Nilai akhir: " << nilaiAkhir << endl;
13     return 0;
14 }
15
```

Output :

```
PROBLEMS  OUTPUT  TERMINAL  DEBUG CONSOLE  PORTS

C:\Users\ACER\Documents\Tugas_ADT>soal6
Nilai akhir: 77

C:\Users\ACER\Documents\Tugas_ADT>
```

Deskripsi :

Pada soal keenam, dibuat sebuah fungsi untuk menghitung nilai akhir berdasarkan rumus: 30% CLO1, 30% CLO2, 20% CLO3, dan 20% CLO4. Fungsi ini mengembalikan nilai akhir bertipe float. Program ini mencontohkan bagaimana perhitungan sederhana dapat diimplementasikan dalam fungsi terpisah.

7) source code:

```
soal7.cpp > ...
1  #include <iostream>
2  #include <string>
3  using namespace std;
4
5  string tentukanIndeks(float nilaiAkhir) {
6      if (nilaiAkhir >= 85) return "A";
7      else if (nilaiAkhir >= 80) return "AB";
8      else if (nilaiAkhir >= 70) return "B";
9      else if (nilaiAkhir >= 65) return "BC";
10     else if (nilaiAkhir >= 55) return "C";
11     else if (nilaiAkhir >= 40) return "D";
12     else return "E";
13 }
14
15 int main() {
16     float nilaiAkhir = 78;
17     string indeks = tentukanIndeks(nilaiAkhir);
18
19     cout << "Nilai akhir: " << nilaiAkhir << endl;
20     cout << "Indeks nilai: " << indeks << endl;
21     return 0;
22 }
23
```

Output :

```
C:\Users\ACER\Documents\Tugas_ADT>soal7
Nilai akhir: 78
Indeks nilai: B

C:\Users\ACER\Documents\Tugas_ADT>|
```

Deskripsi :

Program ini membuat fungsi tentukanIndeks untuk menentukan indeks huruf berdasarkan nilai akhir. Fungsi menggunakan percabangan if-else untuk mengelompokkan nilai ke dalam kategori A, AB, B, BC, C, D, atau E. Dengan demikian, setiap nilai akhir mahasiswa dapat diubah ke bentuk indeks mutu sesuai aturan.

C. Kesimpulan

Secara keseluruhan, rangkaian program ini menunjukkan penerapan dasar-dasar pemrograman terstruktur dengan bahasa C++. Dimulai dari pembuatan *struct* untuk mengelola data mahasiswa, deklarasi variabel bertipe bentukan, hingga pembuatan fungsi untuk input, perhitungan, dan penentuan indeks nilai.

Melalui soal-soal ini, konsep penting seperti tipe bentukan, fungsi, input-output, dan percabangan dapat dipahami secara praktis. Setiap soal saling melengkapi, sehingga terbentuk satu alur program yang utuh: dari pengambilan data, pengolahan, hingga interpretasi hasil dalam bentuk indeks.

Dengan implementasi ini, mahasiswa dilatih untuk memecah masalah menjadi bagian kecil (modularisasi), menulis kode yang terstruktur, dan menghasilkan output yang sesuai kebutuhan. Hal ini menjadi dasar penting untuk mempelajari struktur data dan algoritma yang lebih kompleks di tahap berikutnya.

D. Referensi

References

- Cyganek , B. (2020). *Introduction to Programming with C++ for Engineers*. Malaysia: Wiley-IEEE Press.
- Aung, S. L., Funabiki , N., Dim, N. K., Aye , S. M., & Kyaw, H. H. (2022). *Investigation of Value Trace Problem for C++ Programming Self-study of Novice Students*. Myanmar: International Journal of Information and Education Technology.
- Guan, W. (2023). *Research on how to optimize data structures with C++language*. USA: Francis Academic Press.
- Nugroho, A. Y., & Sutanto, H. N. (2024). *Exploring the Code Foundation: A Literature Review of Data Structures in C++*. Semarang: International Journal of Mechanical, Industrial and Control Systems Engineering.
- Shafinah, K., Punzalan, C., Derahman, M. N., Mohd, S. M., Jan, N. M., & Wahab, A. N. (2022). *The Students' Self Perception on Learning C++ Programming via*. Malaysia: Human Resource Management Academic Research Society (.

