

LAPORAN PRAKTIKUM
Modul 1
Pengenalan CPP bagian 1



Disusun Oleh:
Ade Fatkhul Anam 2211104051

Dosen :
Wahyu Andi Saputra, S.Pd., M.Eng.

PROGRAM STUDI S1 SOFTWARE ENGINEERING
FAKULTAS INFORMATIKA
TELKOM UNIVERSITY
PURWOKERTO
2024

1. Tujuan

- Menguasai lingkungan kerja Code Blocks dengan baik.
- Memahami cara menggunakan serta melakukan pemecahan masalah pada Code Blocks IDE.
- Mengaplikasikan operator-operator dalam pembuatan program.
- Mengerti cara menyusun program sederhana menggunakan bahasa C++.
- Memahami penggunaan tipe data dan variabel dalam C++.
- Menggunakan operator input/output dengan benar.
- Memahami dan menerapkan fungsi-fungsi kondisional dalam program.

2. Landasan Teori

- **Bahasa C++**

C++ adalah bahasa pemrograman tingkat tinggi yang diciptakan oleh Bjarne Stroustrup pada tahun 1979 sebagai pengembangan dari bahasa C. Bahasa ini mendukung konsep pemrograman berorientasi objek (OOP) serta prosedural. C++ sering digunakan untuk membuat aplikasi yang membutuhkan kinerja tinggi, seperti perangkat lunak sistem, game, aplikasi desktop, dan pengembangan sistem tertanam.

- **Code::Blocks**

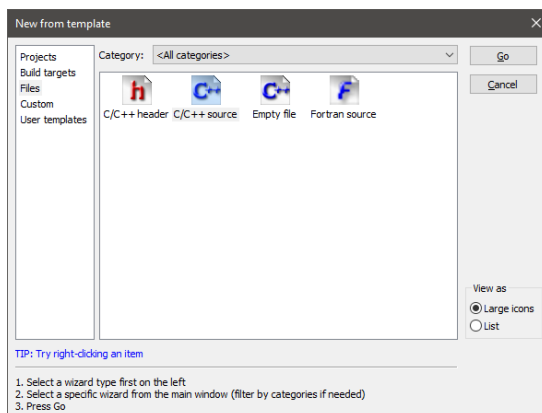
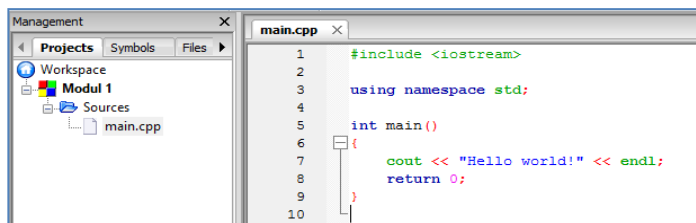
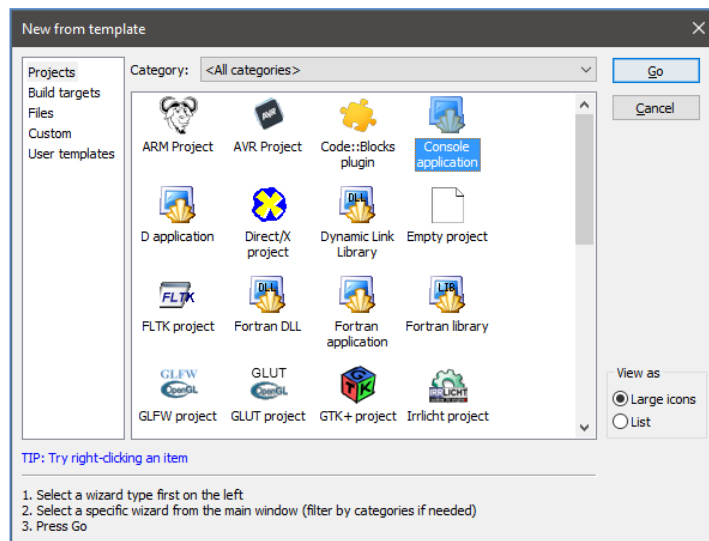
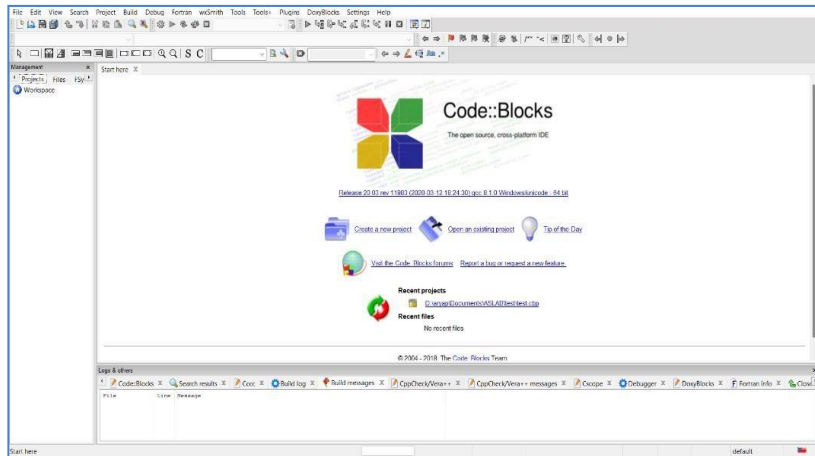
Code::Blocks merupakan Integrated Development Environment (IDE) yang banyak digunakan untuk pengembangan program dalam bahasa C, C++, dan Fortran. Ini adalah perangkat lunak open-source yang dirancang agar fleksibel dan mudah digunakan dalam pengembangan program. Code::Blocks mendukung berbagai compiler seperti GCC, MSVC, dan Clang, serta menyediakan fitur debugging dan profiling secara langsung.

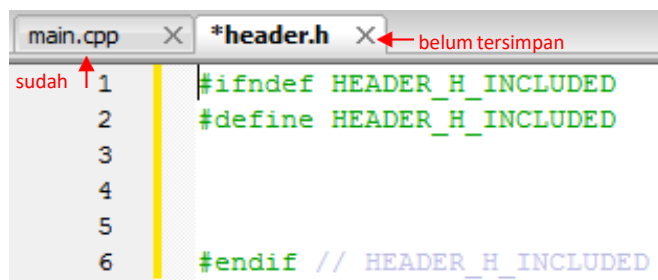
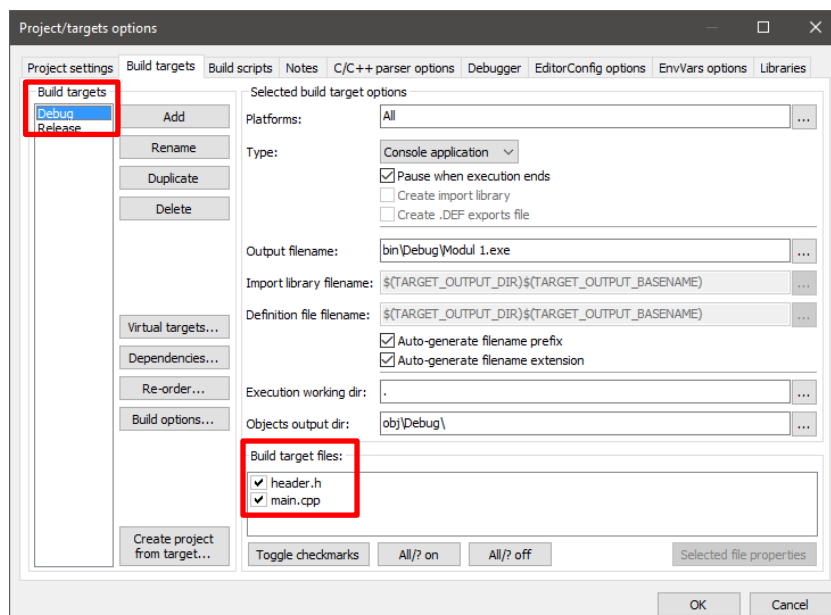
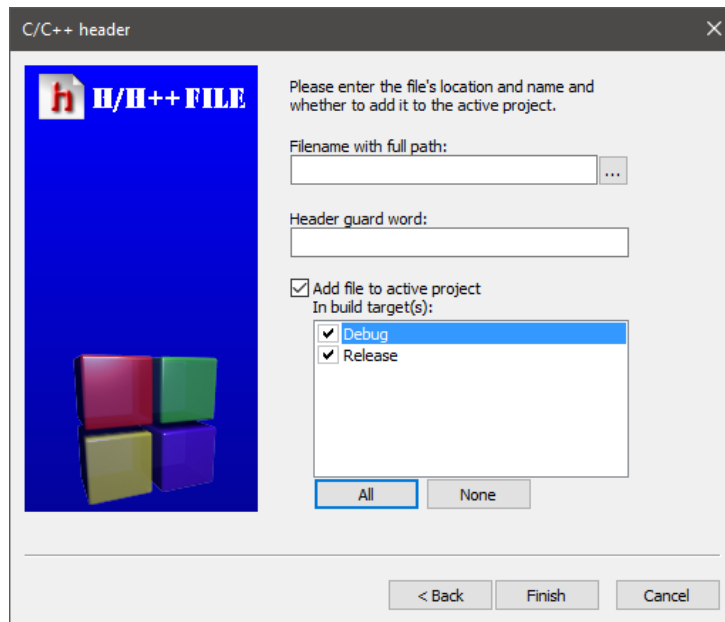
- **Tipe data**

Tipe data dalam C++ mengacu pada jenis nilai yang dapat disimpan oleh sebuah variabel. Tipe data menentukan besaran memori yang dibutuhkan dan operasi yang bisa dilakukan terhadap nilai tersebut.

3. Guided

a) Instalasi Codeblock





b) Tipe data

Type data	Contoh	Ukuran	Jangkauan nilai
Char	char nama[20];	1 Byte	-128 s.d. +127
Int	int nilai; int jumlah = 0;	2 Byte	-32768 s.d +32767
Long	long selisih;	4 Byte	-2.147.438.648 s.d +2.147.438.647
Float	float jumlah;	4 Byte	3.4e-38 s.d 3.4e+38
Double	double hasil;	8 Byte	1.7e-308 s.d 1.7e+308

```
// Tipe data Integer
int angka = 10;

// Tipe data desimal
float desimal = 10.5;

// Tipe data String
string kalimat = "aldi";

// Tipe data double
double tinggi = 10.4;

// Tipe data char
char jenis_kelamin = 'L';

// Tipe data boolean
bool isSunny = true;

cout << "Angka: " << angka << endl;
```

c) Input Output

Fungsi cin() digunakan untuk menerima input dari keyboard yang dimasukkan oleh pengguna. Sedangkan, fungsi cout() berfungsi untuk menampilkan data di layar, baik data numerik maupun teks, termasuk konstanta atau variabel.

```
int angka;
cout << "Masukan angka: ";
cin >> angka;
cout << "Angka: " << angka << endl;
getch();
```

d) Operator

Kategori (Arti)	Operator	Kategori (Arti)	Operator
Panggilan fungsi, subscript <i>array</i> , dan elemen struktur data	() [] ->	Operator Hubungan (sama dengan, tidak sama dengan)	== !=
Operator Unary (NOT, komplemen, negasi, inkremen, dekremen , <i>address</i> , <i>indirection</i>)	!	Operator Bitwise AND	&
	~	Operator Bitwise XOR	^
	-	Operator Bitwise OR	
	++	Operator Logika AND	&&
	--	Operator Logika OR	
	& *	Operator Kondisional	?:
Operator Aritmatika(Perkalian, pembagian, Sisa Pembagian/mod)	* / %	Operator Pengerjaan Aritmatika (<i>assignment</i> , <i>assignment</i> perkalian, <i>assignment</i> pembagian, <i>assignment</i> mod, <i>assignment</i> penjumlahan, <i>assignment</i> pengurangan)	= *= /= %= += -=
Operator Aritmatika (Pertambahan, Pengurangan)	+ -		
Operator Bitwise Pergeseran Bit (shift kiri, shift kanan)	<< >>		&= ^= = <<= >>=
Operasi Hubungan (kurang dari, kurang dari atau sama dengan lebih dari, lebih dari atau sama dengan)	< <= > >=		
		Operator Koma	,

```

int angka = 10;
int angka1 = 15;
int penjumlahan = angka + angka1;
int pengurangan = angka - angka1;
int perkalian = angka * angka1;
int pembagian = angka / angka1;
int modulus = angka % angka1;
int perbandingan = angka == angka1;

cout << "Hasilnya adalah: " << hasil << endl;

```

e) Boolean

```

bool konsisi1 = true;
bool kondisi2 = true;

bool hasil = konsisi1 && kondisi2;
cout << "Hasilnya adalah: " << boolalpha << hasil << endl;

```

f) If Else (Percabangan)

```
string kata;

cout << "Masukan kata Hallo: " << endl;
cin >> kata;

if(kata == "Hallo"){
    cout << "Kata sesuai" << endl;
} else {
    cout << "Kata tidak sesuai" << endl;
}
```

g) Switch case (Percabangan)

```
int tv;

cout << "daftar chanel TV" << endl;
cout << "1. RCTI" << endl;
cout << "2. Indosiar" << endl;
cout << "Masukan chanel pilihan: ";
cin >> tv;

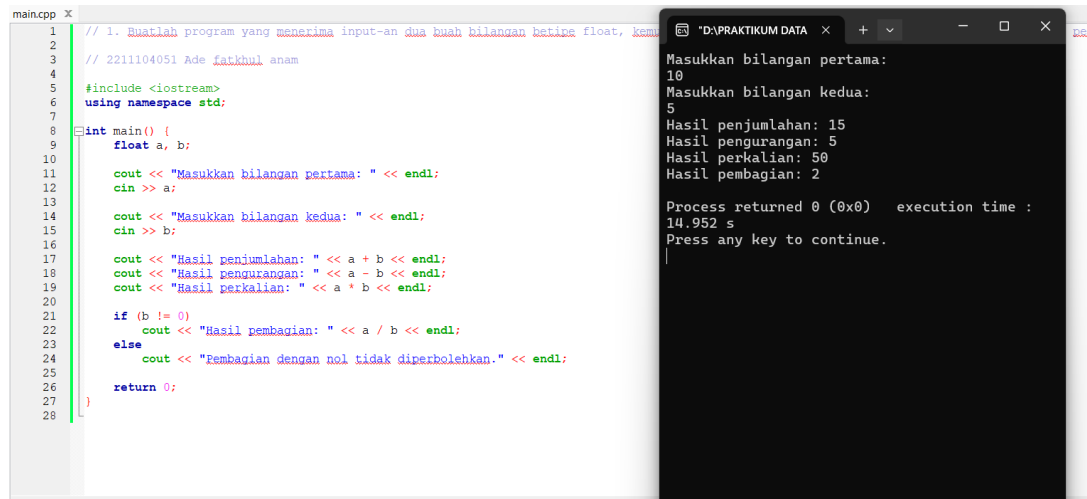
switch(tv) {
case 1 :
    cout << "Chanell yang dipilih rcti" << endl;
    break;
case 2 :
    cout << "Chanell yang dipilih indosiar" << endl;
    break;
default:
    cout << "Chanel tidak tersedia" << endl;
    break;
}
```

h) For Loop (Looping)

```
for(int i = 5; i > 0; i--){
    cout << i << "Hello Word!" << endl;
}
```

4. Unguided

- a) Buatlah program yang menerima *input*-an dua buah bilangan betipe float, kemudian memberikan *output*-an hasil penjumlahan, pengurangan, perkalian, dan pembagian dari dua bilangan tersebut.

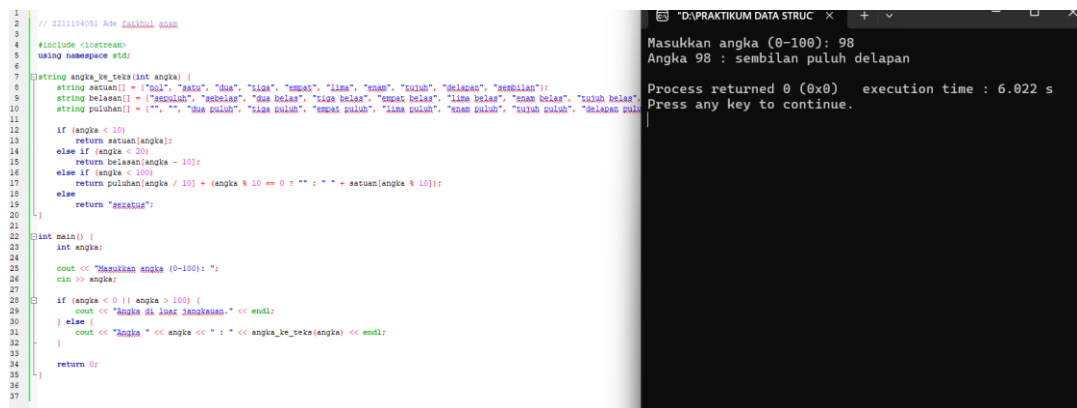


```
1 // 1. Buatlah program yang menerima input-an dua buah bilangan betipe float, kemudian
2 // 2211104051 Ade Fatkhul anam
3
4 #include <iostream>
5 using namespace std;
6
7 int main() {
8     float a, b;
9
10    cout << "Masukkan bilangan pertama: " << endl;
11    cin >> a;
12
13    cout << "Masukkan bilangan kedua: " << endl;
14    cin >> b;
15
16    cout << "Hasil penjumlahan: " << a + b << endl;
17    cout << "Hasil pengurangan: " << a - b << endl;
18    cout << "Hasil perkalian: " << a * b << endl;
19
20    if (b != 0)
21        cout << "Hasil pembagian: " << a / b << endl;
22    else
23        cout << "Pembagian dengan nol tidak diperbolehkan." << endl;
24
25    return 0;
26 }
```

Masukkan bilangan pertama:
10
Masukkan bilangan kedua:
5
Hasil penjumlahan: 15
Hasil pengurangan: 5
Hasil perkalian: 50
Hasil pembagian: 2
Process returned 0 (0x0) execution time : 14.952 s
Press any key to continue.

- b) Buatlah sebuah program yang menerima masukan angka dan mengeluarkan *output* nilai angka tersebut dalam bentuk tulisan. Angka yang akan di- *input*-kan user adalah bilangan bulat positif mulai dari 0 s.d 100. contoh:

79 : tujuh puluh Sembilan



```
1 // 2211104051 Ade Fatkhul anam
2
3 #include <iostream>
4 using namespace std;
5
6 string angka_ke_teks(int angka) {
7     string satuan[] = {"", "satu", "dua", "tiga", "empat", "lima", "enam", "tujuh", "delapan", "sembilan"};
8     string belasan[] = {"", "sepuluh", "sebelas", "dua belas", "tiga belas", "empat belas", "lima belas", "enam belas", "tujuh belas", "delapan belas", "sembilan belas"};
9     string puluhan[] = {"", "puluhan", "dua puluh", "tiga puluh", "empat puluh", "lima puluh", "enam puluh", "tujuh puluh", "delapan puluh", "sembilan puluh"};
10
11     if (angka < 10)
12         return satuan[angka];
13     else if (angka < 20)
14         return belasan[angka - 10];
15     else if (angka < 100)
16         return puluhan[angka / 10] + (angka % 10 == 0 ? "" : " " + satuan[angka % 10]);
17     else
18         return "satusatua";
19 }
20
21 int main() {
22     int angka;
23
24     cout << "Masukkan angka (0-100): ";
25     cin >> angka;
26
27     if (angka < 0 || angka > 100) {
28         cout << "Angka di luar jangkauan." << endl;
29     } else {
30         cout << "Angka " << angka << " : " << angka_ke_teks(angka) << endl;
31     }
32
33     return 0;
34 }
```

Masukkan angka (0-100): 98
Angka 98 : sembilan puluh delapan
Process returned 0 (0x0) execution time : 6.822 s
Press any key to continue.

c) Buatlah program yang dapat memberikan *input* dan *output* sbb.

```
input: 3
output:
  3 2 1 * 1 2 3
    2 1 * 1 2
      1 * 1
        *
```

```
1 // 2211104051 Ade fatkhul anam
2
3
4 #include <iostream>
5 using namespace std;
6
7
8 int main() {
9     int n;
10
11     cout << "Masukkan angka: ";
12     cin >> n;
13
14     for (int i = n; i >= 1; i--) {
15         for (int j = i; j >= 1; j--) {
16             cout << j << " ";
17         }
18         cout << "* ";
19
20         for (int j = 1; j <= i; j++) {
21             cout << j << " ";
22         }
23         cout << endl;
24     }
25     cout << "+" << endl;
26     return 0;
27 }
```

```
"D:\PRAKTIKUM DATA STRUC" x + v
Masukkan angka: 3
3 2 1 * 1 2 3
2 1 * 1 2
1 * 1
*
Process returned 0 (0x0)   execution time : 14.161 s
Press any key to continue.
```

5. Kesimpulan

C++ merupakan bahasa pemrograman yang menawarkan pemahaman mendalam mengenai struktur pemrograman. Mendukung konsep pemrograman berorientasi objek (OOP), C++ memungkinkan penggunaan ulang kode yang meningkatkan keandalan program. Beragam fitur seperti fungsi anggota, konstruktor, kontrol akses, dan pewarisan memperkaya konsep data. Sebagai bahasa tingkat tinggi, C++ memudahkan perancangan algoritma dan instruksi yang lebih manusiawi. Selain itu, C++ juga membantu pengembangan keterampilan dalam manajemen memori dan efisiensi pengelolaan kompleksitas waktu.