

LAPORAN PRAKTIKUM
Modul 1
Pengenalan CPP bagian 1



Disusun Oleh:
Satria Ariq Adelard Dompas
2211104033

Dosen :
Wahyu Andi Saputra

PROGRAM STUDI S1 SOFTWARE ENGINEERING
FAKULTAS INFORMATIKA
TELKOM UNIVERSITY
PURWOKERTO
2024

1. Tujuan

- Mengenal environment Code Blocks dengan baik.
- Memahami cara menggunakan dan troubleshooting Code Blocks IDE.
- Mengimplementasikan operator-operator dalam program.
- Memahami cara membuat program sederhana dalam bahasa C++.
- Memahami penggunaan tipe data dan variabel dalam bahasa C++.
- Menggunakan operator-operator input/output dengan tepat.
- Memahami dan mengimplementasikan fungsi kondisional dalam program.

2. Landasan Teori

- Bahasa C++

C++ adalah bahasa pemrograman tingkat tinggi yang dikembangkan oleh Bjarne Stroustrup pada tahun 1979 sebagai perpanjangan dari bahasa C. Bahasa tersebut mendukung konsep pemrograman berorientasi objek (OOP) dan pemrograman prosedural. C++ banyak digunakan dalam pengembangan aplikasi yang memerlukan kinerja tinggi seperti perangkat lunak sistem, game, aplikasi perkantoran, dan pengembangan sistem tertanam.

- Code::Blocks

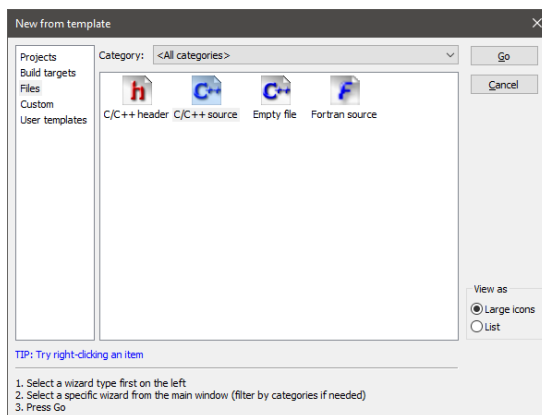
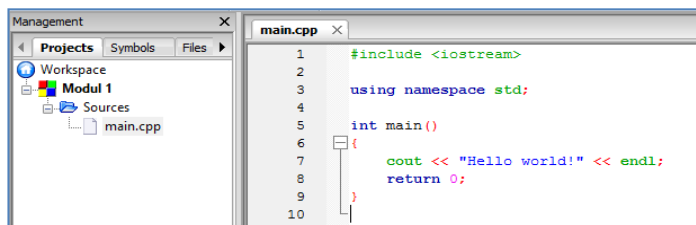
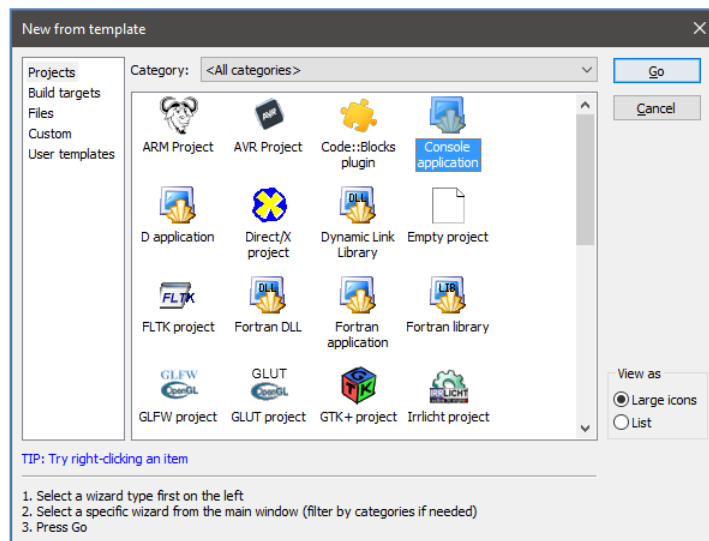
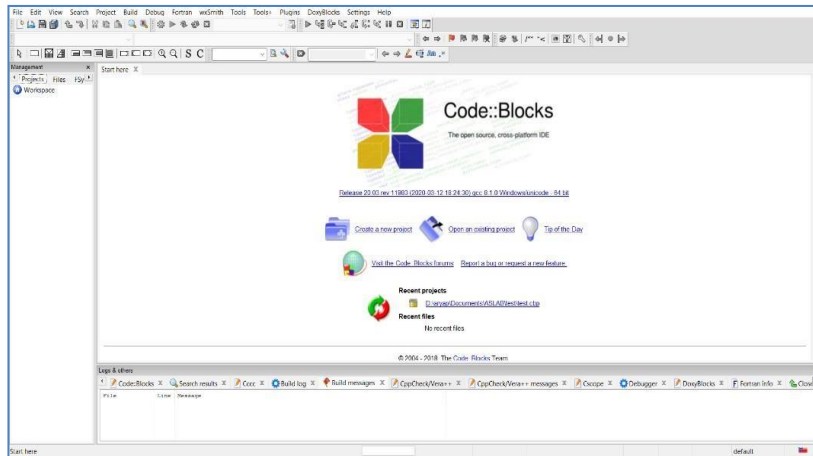
Code: : Blok adalah lingkungan pengembangan terintegrasi (IDE) yang biasa digunakan untuk mengembangkan program C, C++ dan Fortran. Ini adalah perangkat lunak sumber terbuka yang dirancang untuk fleksibilitas dan kemudahan pengembangan program. Code: : Block mendukung berbagai kompiler seperti GCC, MSVC dan Clang, dan memungkinkan debugging dan pembuatan profil langsung di dalamnya.

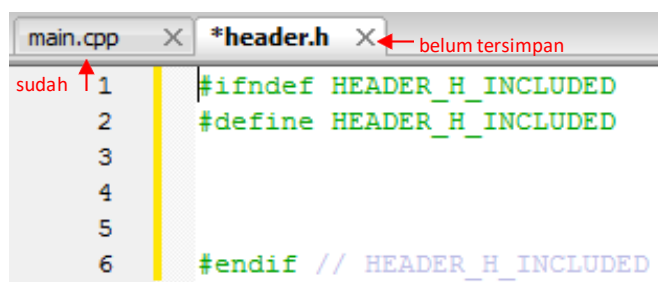
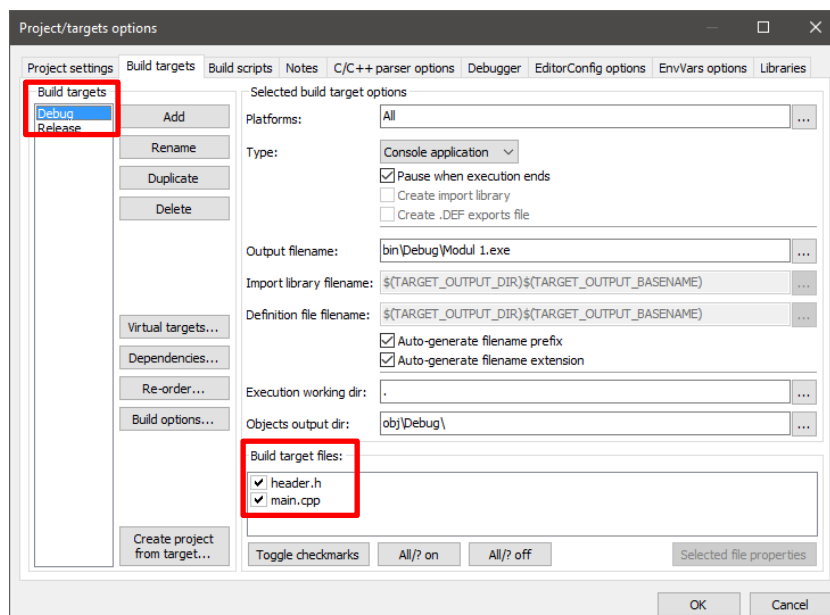
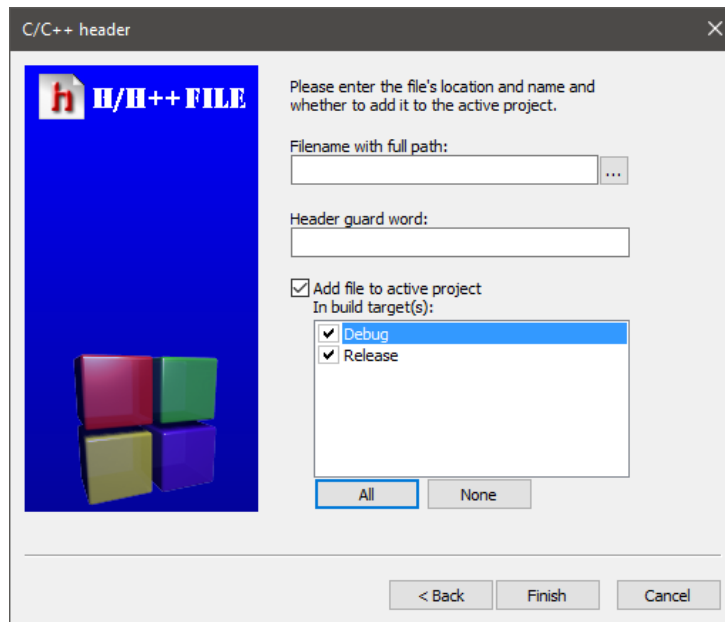
- Tipe data

Tipe data dalam C++ mengacu pada tipe nilai yang dapat ditampung oleh suatu variabel. Tipe data menentukan ukuran memori yang digunakan dan operasi yang dapat dilakukan pada 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) Oprator

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, 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, 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)	<< >>	Operator Pengerjaan Bitwise (<i>assignment</i> AND bitwise, <i>assignment</i> OR bitwise, <i>assignment</i> XOR bitwise, <i>assignment</i> 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.

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

int main()
{
    float a, b;

    cout << "Masukkan bilangan pertama: " << endl;
    cin >> a;

    cout << "Masukkan bilangan kedua: " << endl;
    cin >> b;

    cout << "Hasil penjumlahan: " << a + b << endl;
    cout << "Hasil pengurangan: " << a - b << endl;
    cout << "Hasil perkalian: " << a * b << endl;

    if (b != 0)
        cout << "Hasil pembagian: " << a / b << endl;
    else
        cout << "Pembagian dengan nol tidak diperbolehkan." << endl;

    return 0;
}
```

Masukkan bilangan pertama: 13
Masukkan bilangan kedua: 15
Hasil penjumlahan: 28
Hasil pengurangan: -2
Hasil perkalian: 195
Hasil pembagian: 0.866667
Process returned 0 (0x0) execution time : 3.850 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

```
// 2. Buatlah sebuah program yang menerima masukan angka dan mengeluarkan output nilai angka tersebut dalam bentuk tulisan. Angka yang akan di- input-kan user
#include <iostream>
using namespace std;

string angka_ke_teks(int angka) {
    string satuan[] = {"nol", "satu", "dua", "tiga", "empat", "lima", "enam", "tujuh", "delapan", "sembilan"};
    string belasan[] = {"sepuluh", "sebelas", "dua belas", "tiga belas", "empat belas", "lima belas", "enam belas", "tujuh belas", "delapan belas", "sembilan belas"};
    string puluhan[] = {"", "", "dua puluh", "tiga puluh", "empat puluh", "lima puluh", "enam puluh", "tujuh puluh", "delapan puluh", "sembilan puluh"};

    if (angka < 10)
        return satuan[angka];
    else if (angka < 20)
        return belasan[angka - 10];
    else if (angka < 100)
        return puluhan[angka / 10] + (angka % 10 == 0 ? "" : " " + satuan[angka % 10]);
    else
        return "seratus";
}

int main() {
    int angka;

    cout << "Masukkan angka (0-100): ";
    cin >> angka;

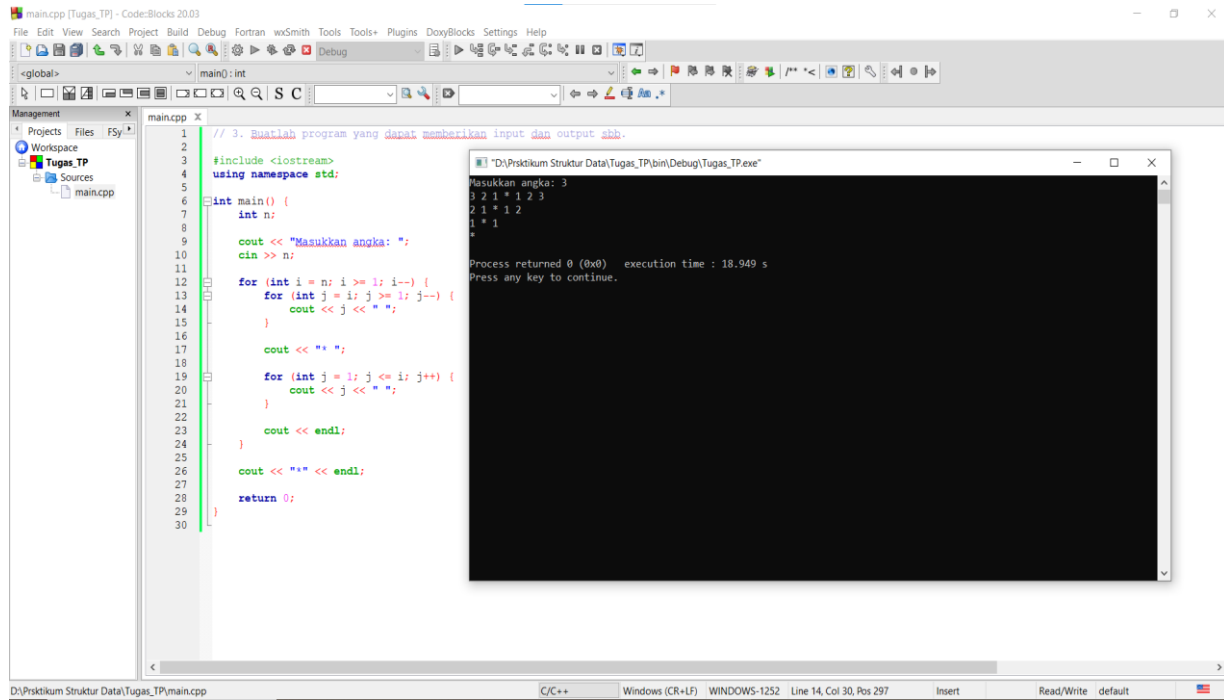
    if (angka < 0 || angka > 100) {
        cout << "Angka di luar jangkauan." << endl;
    } else {
        cout << "Angka " << angka << " : " << angka_ke_teks(angka) << endl;
    }

    return 0;
}
```

Masukkan angka (0-100): 47
Angka 47 : empat puluh tujuh
Process returned 0 (0x0) execution time : 41.527 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
        *
```



The screenshot shows the Code::Blocks IDE with a C++ project named 'Tugas_TP'. The source file 'main.cpp' contains the following code:

```
1 // 3. Buatlah program yang dapat memberikan input dan output sbb.
2
3 #include <iostream>
4 using namespace std;
5
6 int main() {
7     int n;
8     cout << "Masukkan angka: ";
9     cin >> n;
10
11     for (int i = n; i >= 1; i--) {
12         for (int j = i; j >= 1; j--) {
13             cout << j << " ";
14         }
15         cout << " * ";
16
17         for (int j = 1; j <= i; j++) {
18             cout << j << " ";
19         }
20         cout << endl;
21     }
22     cout << "==" << endl;
23     return 0;
24 }
```

The execution window shows the output of the program when the input is 3:

```
"D:\Prskikum Struktur Data\Tugas_TP\bin\Debug\Tugas_TP.exe"
Masukkan angka: 3
3 2 1 * 1 2 3
  2 1 * 1 2
    1 * 1
      *
Process returned 0 (0x0)   execution time : 18.949 s
Press any key to continue.
```

5. Kesimpulan

C++ adalah bahasa pemrograman yang memberikan informasi mendalam tentang struktur pemrograman. Dengan dukungan untuk pemrograman berorientasi objek (OOP), C++ memungkinkan penggunaan kembali kode serta meningkatkan keandalan program. Berbagai fitur seperti fungsi anggota, konstruktor, penentu akses, dan pewarisan memperluas konsep struktur data. Sebagai bahasa tingkat tinggi, C++ membantu pemrogram menyusun algoritma dan instruksi dengan cara yang lebih intuitif bagi manusia. Selain itu, C++ membantu mengembangkan keterampilan manajemen memori serta memecahkan masalah waktu yang kompleks secara efektif.