

MODUL MATA KULIAH PRAKTIKUM ALGORITMA DAN PEMROGRAMAN

Modul 1 : Review Pemrograman Bahasa C++



Dosen Pengampu:
Herryawan Pujiharsono, S.T., M.Eng.
Slamet Indriyanto, S.T., M.T.
Prasetyo Yuliantoro, S.T., M.T.

**PROGRAM STUDI S1 TEKNIK TELEKOMUNIKASI
FAKULTAS TEKNIK TELEKOMUNIKASI DAN ELEKTRO
INSTITUT TEKNOLOGI TELKOM PURWOKERTO
2019**

I. TUJUAN

1. Memahami lingkungan dasar C++, Codeblocks IDE, serta cara meng-compile dan menjalankan program
2. Membuat program sederhana menggunakan bahasa C++ untuk menampilkan kalimat atau string dan perhitungan aritmatika
3. Memahami dan dapat mengimplementasikan tipe data, value, operator, dan sekuens dalam bahasa pemrograman
4. Memahami dan dapat mengimplementasikan fungsi input dan output dalam C++

II. DASAR TEORI

I. LINGKUNGAN PEMROGRAMAN BAHASA C++

1. Pengantar

Bahasa C++ adalah salah satu bahasa pemrograman populer yang sudah terbukti banyak digunakan oleh para praktisi dan ilmuwan untuk mengembangkan program-program (aplikasi) berskala besar seperti *games* (program permainan di komputer), program untuk penelitian di bidang sains, *embedded system*, dan lain-lain. Bahkan, Bahasa C++ juga dapat digunakan untuk mengembangkan aplikasi berbasis web, yang sering disebut dengan program CGI (Common Gateway Interface). Selama ini, banyak yang mengira bahwa aplikasi web hanya dapat dikembangkan dengan PHP, ASP, JSP maupun Perl. Namun, sebenarnya dengan Bahasa C++ pun kita bisa melakukannya.

Saat ini banyak sekali beredar bahasa pemrograman baru yang saling bersaing guna menjaring banyaknya pengguna dan juga popularitas. Akan tetapi, pada kenyataannya, C++ masih menjadi idola dan merupakan tiga peringkat teratas dan bahasa-bahasa yang paling banyak digunakan oleh para programmer (skala internasional), selain bahasa C dan Java. Selain ketiga bahasa pemrograman tersebut, bahasa lain yang mulai beranjak naik dan mulai banyak penggunanya adalah Python dan Ruby.

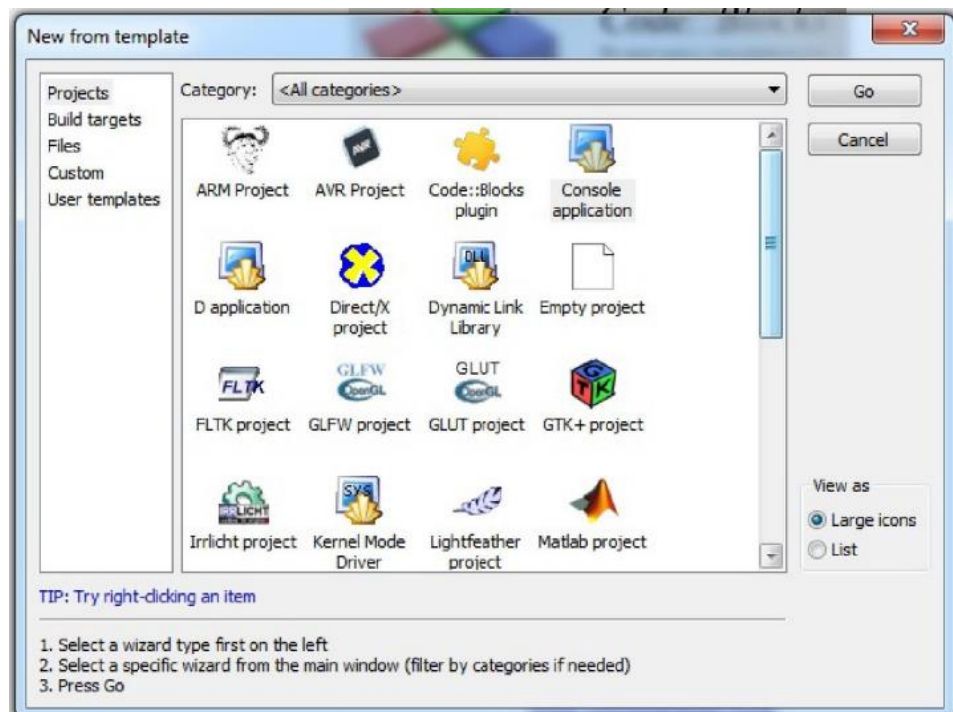
2. Code::Blocks

CodeBlocks merupakan *software* IDE (*Integrated Development Environment*) yang digunakan untuk mengembangkan aplikasi berdasarkan bahasa pemrograman C atau C++. Langkah – langkah membuat program menggunakan Bahasa C++ dengan CodeBlocks adalah sebagai berikut:

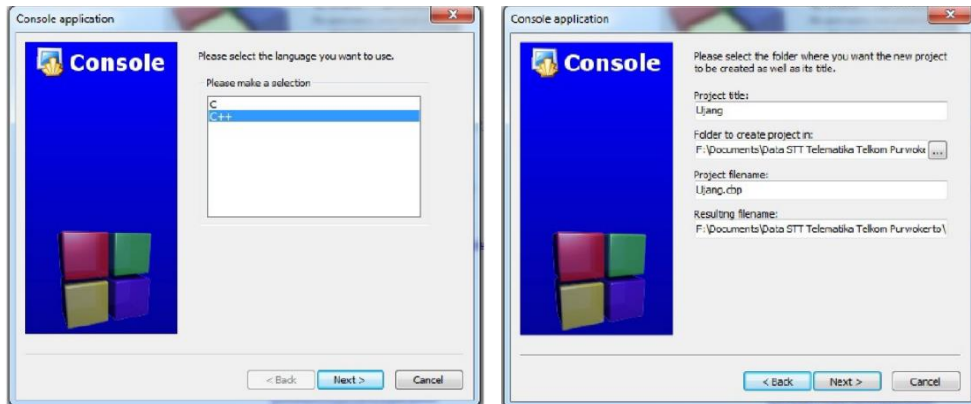
a. Buka program CodeBlocks



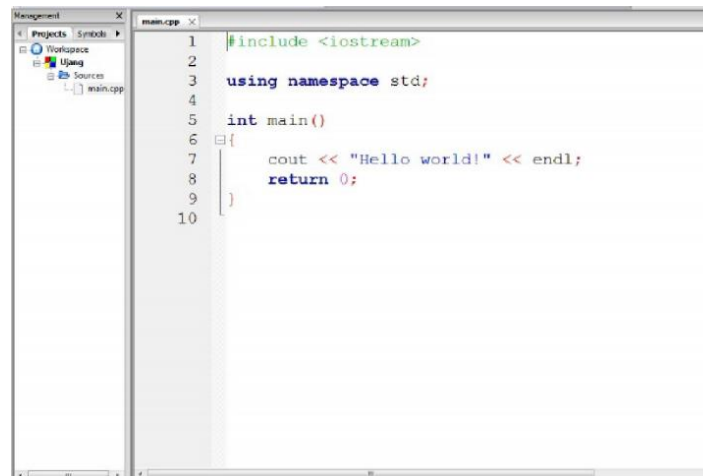
b. Klik Create a new Project => Pilih Console Application



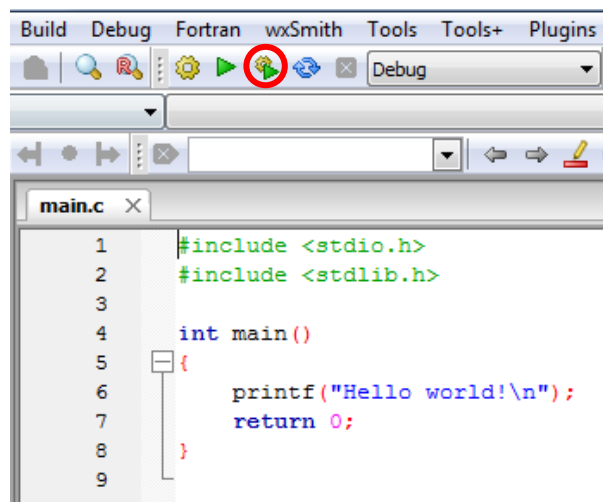
c. Simpan Project dengan nama (misal ujang)



d. Code Blocks secara otomatis akan membuat program sederhana untuk mencetak **“Hello World!”** ke layar.



e. Compile program dengan menekan ikon “Build & Run” atau menekan F9.



3. Dasar-Dasar Pemrograman C++

Suatu Bahasa C++ ditulis dari beberapa elemen, antara lain:

a. Pernyataan

Satu atau beberapa ekspresi (statement dalam satu baris) yang diakhiri tanda semicolon/titik koma(;).

b. Blok pernyataan

Terdiri dari satu atau beberapa pernyataan diawali dengan tanda { dan diakhiri dengan tanda }. Blok pernyataan selalu diawali dengan suatu instruksi.

c. Blok fungsi

Blok fungsi dimulai dengan nama fungsi dan diikuti tanda () dan tanda {...}. Blok tersebut biasanya dirancang untuk beberapa pernyataan dalam melakukan proses tertentu. Contoh:

Kode Program 1.1
<pre>#include <iostream> using namespace std; int main () { cout << "Hello World!"; return 0 ; }</pre>

Keterangan:

#include <iostream>

Perintah yang diawali dengan tanda (#) adalah *preprocessor directive*. Pada statement #include <iostream>, seluruh isi dari file “iostream” akan dimasukkan kedalam kode program. Selanjutnya, bentuk kode yang telah diperluas ini kemudian siap dikirim ke program C++ *Compiler*.

using namespace std;

Statement tersebut menunjukkan bahwa kita menggunakan bahasa C++ standar di dalam kode program yang kita tulis.

int main ()

merupakan deklarasi fungsi main. Fungsi main merupakan sebuah program utama yang berisi perintah program C++ yang akan dieksekusi.

```
cout << "Hello World";
```

Sebuah instruksi program untuk menampilkan tulisan "Hello World". Perintah cout merupakan standard output stream dalam C++, dimana cout dideklarasikan dalam *header file* iostream.h.

Return 0;

Statement diatas berfungsi untuk mengembalikan nilai ke sistem operasi dan menyatakan bahwa program dieksekusi dan dihentikan secara normal. Apabila program mengandung "bug" maupun dihenikan secara paksa (tidak normal), nilai yang akan dikembalikan ke sistem operasi adalah nilai selain 0. Prototype di atas bersifat optional, artinya kita juga diizinkan untuk mendefinisikan fungsi-fungsi sebelum menulis fungsi utama tanpa menuliskan prototype.

II. VARIABLE

1. Tipe Data

a. Tipe Data Dasar

Tipe	Kapasitas Memori	Ukuran (Range)	Keterangan
char	1 byte	-128 s/d 127	Character
int	2 byte	-327 s/d 32768	Integer (bilangan bulat)
Long	4 byte	-21474386 s/d 21474386	Bilangan Bulat
Float	4 byte	3.4E-38 s/d 3.4E+38	Real presisi tunggal
Double	8 byte	1.7E-308 s/d 1.7E+308	Real presisi ganda

b. Tipe Data Majemuk

Tipe data majemuk adalah tipe data yang merangkum berbagai tipe data lain. Tipe data majemuk ini akan menggunakan sintaks **struct**.

c. Tipe Data Bentukan

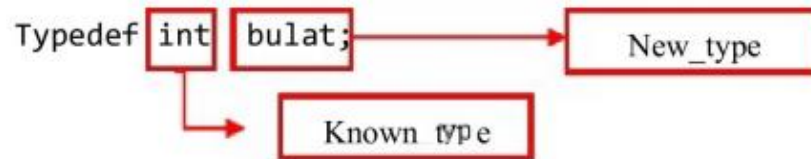
Tipe data bentukan diimplementasikan dengan sintaks **typedef**, yang berarti *Type definition* (atau pendefinisian tipe baru).

Rumus : `typedef [known_type] [new Type];`

Known_type : tipe data dasar

New_type : tipe data baru

Contoh :



2. Variable

Variable adalah representasi tempat penyimpanan dalam memori komputer bagi nilai dari sebuah tipe tertentu. Besar kecilnya variabel akan berpengaruh kinerja program, contoh “a” dengan “A” merupakan berbeda variable meskipun bentuknya sama. Setiap satu variable hanya dapat menyimpan SATU value atau nilai saja. Aturan dalam pendeklarasian variable:

- Variable dapat dideklarasikan dengan menggunakan gabungan huruf dan angka dengan syarat karakter pertama harus huruf.
- Variable yang dideklarasikan tidak boleh menggunakan spasi, apabila ingin menggunakan spasi lebih baik gunakan garis bawah (underscore).
- Variable yang dideklarasikan tidak boleh menggunakan simbol-simbol khusus seperti: \$, &, ?, dll.
- Panjang variable bebas, tetapi yang bisa dikenali hanya 31 karakter pertama.

3. Karakter Escape

Dalam Bahasa C++, juga terdapat karakter khusus yang disebut escape.

Karakter	Keterangan
\0	Karakter NULL (Tulisannya berupa \ dan angka nol)
\a	Karakter bel atau alert
\b	Karakter backspace (mundur satu spasi)
\f	Formfeed (ganti halaman)
\n	Linefeed (disebut juga newline atau pindah baris)

Karakter	Keterangan
\r	Carriage return (menuju kolom pertama, baris yang sama)
\t	Tab horizontal
\v	Tab vertikal
\\	Karakter \
\'	Karakter petik tunggal
\"	Karakter petik ganda
\?	Karakter tanda tanya


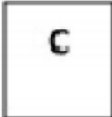
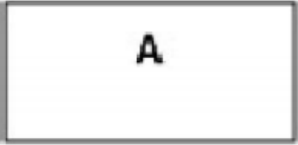
4. Deklarasi

a. Deklarasi Variable

Aturan deklarasi variabel dalam Bahasa C++ adalah sebagai berikut.

[Tipe_data] [nama_variabel];

[Tipe_data] [nama_variabel1], [nama_variabel2];

No	Deklarasi	Arti	Ilustrasi
1	int A;	Memesan tempat dalam memori komputer ruang sebesar integer (4 byte) dan memberi nama tempat penyimpanan itu sebagai A	
2	float C;	Memesan tempat dalam memori komputer ruang sebesar float (4 byte) dan memberi nama tempat penyimpanan itu sebagai C	
3	Double D;	Memesan tempat dalam memori komputer ruang sebesar double (8 byte) dan memberi nama tempat penyimpanan itu sebagai D;	

b. Deklarasi Konstanta

Dalam bahasa C++, konstanta di deklarasikan menggunakan *preprocessor* **#define**. Contoh:

```
#define phi 3.14;  
#define NIM "13101004"  
#define nama "AGUNG WICAKSONO"
```

c. Deklarasi Fungsi dan Prosedur

Berisi subprogram yang nantinya dapat dipanggil oleh program utama. Subprogram tersebut mengurangi penggunaan algoritma yang berulang-ulang yang dapat menyebabkan penulisan kode program menjadi tidak efisien

d. Assignment

Proses pemberian nilai pada sebuah variabel.

Contoh:

```
int a;  
a = 2; // assignment atau int a = 2;
```

5. Operator

Operator merupakan instruksi khusus yang dikenai untuk variabel, ada beberapa jenis operator, yaitu:

a. Operator Penugasan (Assignment Operator)

Operator penugasan dalam bahasa C++ berupa tanda sama dengan atau "=".

Contoh:

```
nilai = 70, A = x * y
```

b. Operator Aritmatika

Operator aritmatika merupakan operator yang digunakan untuk fungsi/operasi matematika, operator aritmatika dasar untuk C++ dan tanpa proses include:

*	: untuk perkalian	+	: untuk penjumlahan
/	: untuk pembagian	-	: untuk pengurangan
%	: untuk sisa hasil bagi (modulus)		

Operator aritmatika yang menggunakan file header **math.h** seperti berikut:

Pow(a,b) : pemangkatan

Min() : mencari nilai minimal

Max() : mencari nilai maximal

Sqrt() : menghitung akar suatu bilangan

c. Operator Perbandingan

Operator perbandingan digunakan untuk membandingkan antara dua buah variabel.

Pembandingan	Keterangan	Contoh
<	Kurang dari	<code>x < y</code> //Apakah x kurang dari y
<=	Kurang dari sama dengan	<code>x <= y</code> //Apakah x kurang dari sama dengan y
>	Lebih dari	<code>x > y</code> //Apakah x lebih dari y
>=	Lebih dari sama dengan	<code>x >= y</code> //Apakah x lebih dari sama dengan y
==	Sama dengan	<code>x == y</code> //Apakah x sama dengan y
!=	Tidak sama dengan	<code>x != y</code> //Apakah x kurang dari y

d. Operator Logika

Operator logika biasanya membandingkan logika hasil dari dua operasi perbandingan dan akan melakukan operasi khusus apabila nilai bernilai benar (true) atau salah (false).

Logika	Keterangan
&&	Logika dan (AND)
	Logika or (ATAU)
!	Logika not (INGKARAN)

Tabel logika AND

P	Q	P&&Q
F	F	F
F	T	F
T	F	F
T	T	T

Tabel logika OR

P	Q	P Q
F	F	F
F	T	T
T	F	T
T	T	T

Tabel logika NOT

P	‘P
F	T
T	F

e. Operator Bitwise

<< : pergeseran bit ke kiri ^ : bitwise XOR (exclusive OR)
 >> : pergeseran bit ke kanan | : bitwise OR
 & : bitwise AND ~ : bitwise NOT

f. Operator Unary

Operator Unary merupakan operator yang hanya membutuhkan satu operand saja. Dalam bahasa C++ terdapat beberapa operator unary, yaitu:

Operator	Arti/Maksud	Letak	Contoh	Equivalen
-	Unary Minus	Sebelum operator	$A + -B * C$	$A + (-B) * C$
++	Peningkatan dengan penambahan nilai 1	Sebelum dan sesudah	A++	$A = A + 1$
--	Penurunan dengan pengurangan nilai 1	Sebelum dan sesudah	A--	$A = A - 1$
sizeof	Ukuran dari operand dalam byte	Sebelum	sizeof(1)	-
!	Unary NOT	Sebelum	!A	-
~	Bitwise NOT	Sebelum	~A	-
&	Menghasilkan alamat memori operand	Sebelum	&A	-
*	menghasilkan nilai dari pointer	Sebelum	*A	-

III.INPUT DAN OUTPUT

1. Fungsi Input dalam `#include <iostream.h>` atau `<iostream>`

a. `cin >> statement`

digunakan untuk menginput data berupa data numerik, karakter dan string.

b. fungsi `getch()` dan `getche()`

fungsi ini sebagai jeda dalam mengeksekusi program sebelum mengakhiri eksekusi. Fungsi ini didukung dengan menggunakan `#include <stdio.h>`

c. `cin.get(var, size_of(var))`

fungsi ini untuk membaca/menginput string lebih dari satu kata, akan tetapi akan mengakibatkan kesalahan jika dalam satu program menggunakan perintah ini lebih dari satu kali.

2. Fungsi Output dalam `#include <iostream.h>` atau `<iostream>`

a. `cout >> statement`

Digunakan untuk menginput data berupa data numerik, karakter dan string.

b. Fungsi Output dalam `#include <iomanip.h>`

Fungsi tersebut adalah fungsi untuk mengatur presisi bilangan pecahan. Dimana n adalah jumlah digit presisi yang ditampilkan.

3. Manipulator dalam C++

Manipulator	Fungsi
<code>endl</code>	End line, identik dengan <code>\n</code> kursor pindah ke posisi awal berikutnya
<code>ends</code>	Menyisipkan karakter Null
<code>flush</code>	Mencetak isi buffer
<code>dec</code>	Mengkonversi kebilangan decimal
<code>hex</code>	Mengkonversi kebilangan hexadecimal
<code>oct</code>	Mengkonversi kebilangan octadecimal
<code>Setbase(n)</code>	Mengkonversi kebilangan berbasis n
<code>Setw(n)</code>	Mengatur lebar cetakan selebar n
<code>Setfill(c)</code>	Mengisi leading field dengan karakter c
<code>Setprecision (n)</code>	Membuat lebar decimal point = n
<code>Setioflags(lf)</code>	Men-set format yang diatur dengan tanda format ios
<code>Resetiosflag (lf)</code>	Men-set format yang diset oleh setioflags

III. PERCOBAAN

1. Menampilkan Welcome to C++

```
#include <iostream>
using namespace std;
int main()
{
    cout << "Welcome to C++!\n";
    return 0;
}
```

2. Operasi Aritmatika

```
#include <iostream>

using namespace std;
int main()
{
    int m, n;
    m = 27;
    n = 5;
    cout << "m + n = " << m + n << endl;
    cout << "m - n = " << m - n << endl;
    cout << "m x n = " << m * n << endl;
    cout << "m / n = " << m / n << endl;
    cout << "m % n = " << m % n << endl;
    cout << "~m = " << ~m << endl;

    return 0;
}
```

3. Operasi Bitwise

```
#include <iostream>

using namespace std;

int main()
{
    int m, n;
    m = 12;
    n = 2;
    cout << "m >> n = " << (m >> n) << endl;
    cout << "m << n = " << (m << n) << endl;
    cout << "m & n = " << (m & n) << endl;
    cout << "m | n = " << (m | n) << endl;
    cout << "m ^ n = " << (m ^ n) << endl;
    cout << "~m = " << (~m) << endl;
    return 0;
}
```

4. Operasi Logika

```
#include <iostream>

using namespace std;

int main()
{
    int m, n;
    cout << "Masukkan nilai m = ";
    cin >> m;
    cout << "Masukkan nilai n = ";
    cin >> n;
    cout << "m = n --> " << (m==n) << endl;
    cout << "m != n --> " << (m!=n) << endl;
    cout << "m > n --> " << (m>n) << endl;
    cout << "m < n --> " << (m<n) << endl;
    cout << "m >= n --> " << (m>=n) << endl;
    cout << "m <= n --> " << (m<=n) << endl;
    return 0;
}
```

5. Menampilkan perhitungan luas persegi panjang

```
#include <iostream>

using namespace std;

int main()
{
    int luas, panjang, lebar;
    cout<<"Masukan panjang :";
    cin>>panjang;
    cout<<"Masukan lebar :";
    cin>>lebar;
    luas = panjang * lebar;
    cout<<"Luas segi empat = "<<luas;
    return 0;
}
```

IV. TUGAS

Buatlah algoritma (*flowchart* atau pseudocode) dan program dalam Bahasa C++ untuk kasus berikut!

1. Menampilkan nama dan NIM sesuai dengan yang diinputkan oleh user
2. Konversi suhu Fahrenheit menjadi Celcius, Reamur, dan Kelvin dimana nilai suhu Fahrenheit ditentukan oleh user dan nilai Celcius hasil konversi ditampilkan di layar monitor.

3. Menghitung volume tabung (v) dimana $v = \phi \times r^2 \times t$ dimana nilai r dan t diinputkan oleh user
4. Menghitung phitagoras $c = \sqrt{a^2 + b^2}$ dimana nilai a dan b diinputkan oleh user