

I. COVER

**LAPORAN PRAKTIKUM
Modul 1
Pengenal Code Blocks dan C++**



**Disusun Oleh:
Haza Zaidan Zidna Fann
(2311104056)**

**Kelas:
SISE 07 02**

**Dosen :
Wahyu Andi Saputra
PROGRAM STUDI S1 SOFTWARE ENGINEERING
FAKULTAS INFORMATIKA
TELKOM UNIVERSITY
PURWOKERTO
2024**

II. TUJUAN

Code::Blocks adalah sebuah Integrated Development Environment (IDE) yang sangat populer digunakan oleh para programmer, terutama untuk bahasa pemrograman C dan C++

III. Landasan Teori

C++ adalah bahasa pemrograman tingkat menengah yang mendukung pemrograman berorientasi objek (OOP), serta prosedural dan generik. Dikembangkan oleh Bjarne Stroustrup pada 1980-an, C++ digunakan untuk aplikasi yang memerlukan efisiensi tinggi seperti game dan sistem operasi.

Code::Blocks adalah sebuah lingkungan pengembangan terintegrasi (IDE) yang dirancang khusus untuk membantu para programmer dalam menulis, mengkompilasi, menjalankan, dan men-debug kode program, terutama dalam bahasa pemrograman C dan C++.

Code::Blocks menyediakan berbagai fitur untuk memudahkan pengembangan program, seperti:

Editor kode: Tempat menulis kode dengan fitur penyorotan sintaks, auto-completion, dan indentasi otomatis untuk mempermudah pembacaan.

Compiler: Menerjemahkan kode menjadi bahasa mesin.

Debugger: Membantu menemukan dan memperbaiki kesalahan dalam kode.

Project manager: Mengelola proyek pemrograman dengan efisien.

IV. GUIDED :

Berikut cara install code block dan cara menggunakan :

1. Download code block di tautan ini:
<https://www.codeblocks.org/downloads/>
2. Pilih code block sesuai perangkat

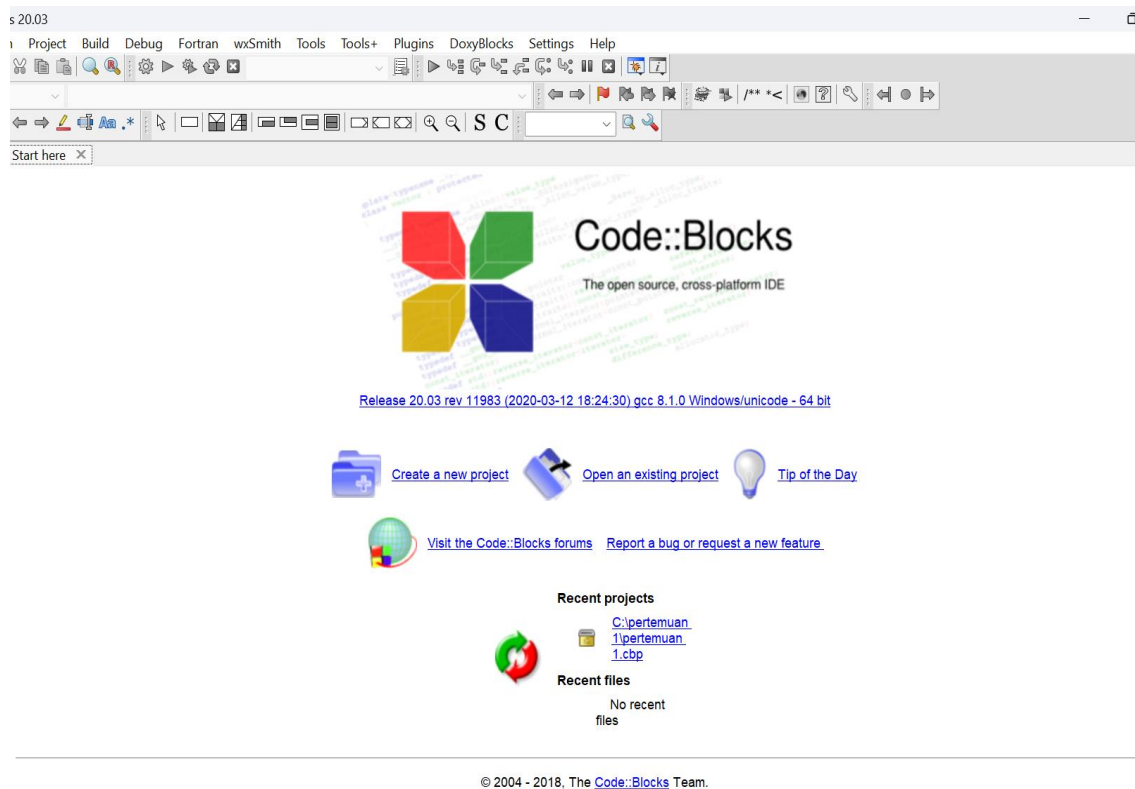


| File | Download from |
|---|--|
| codeblocks-20.03-setup.exe | FossHUB or Sourceforge.net |
| codeblocks-20.03-setup-nonadmin.exe | FossHUB or Sourceforge.net |
| codeblocks-20.03-nosetup.zip | FossHUB or Sourceforge.net |
| codeblocks-20.03mingw-setup.exe | FossHUB or Sourceforge.net |
| codeblocks-20.03mingw-nosetup.zip | FossHUB or Sourceforge.net |
| codeblocks-20.03-32bit-setup.exe | FossHUB or Sourceforge.net |
| codeblocks-20.03-32bit-setup-nonadmin.exe | FossHUB or Sourceforge.net |
| codeblocks-20.03-32bit-nosetup.zip | FossHUB or Sourceforge.net |
| codeblocks-20.03mingw-32bit-setup.exe | FossHUB or Sourceforge.net |
| codeblocks-20.03mingw-32bit-nosetup.zip | FossHUB or Sourceforge.net |

NOTE: The codeblocks-20.03-setup.exe file includes Code::Blocks with all plugins. The codeblocks-20.03-setup-nonadmin.exe file is provided for convenience to users that do not have administrator rights on their machine(s).

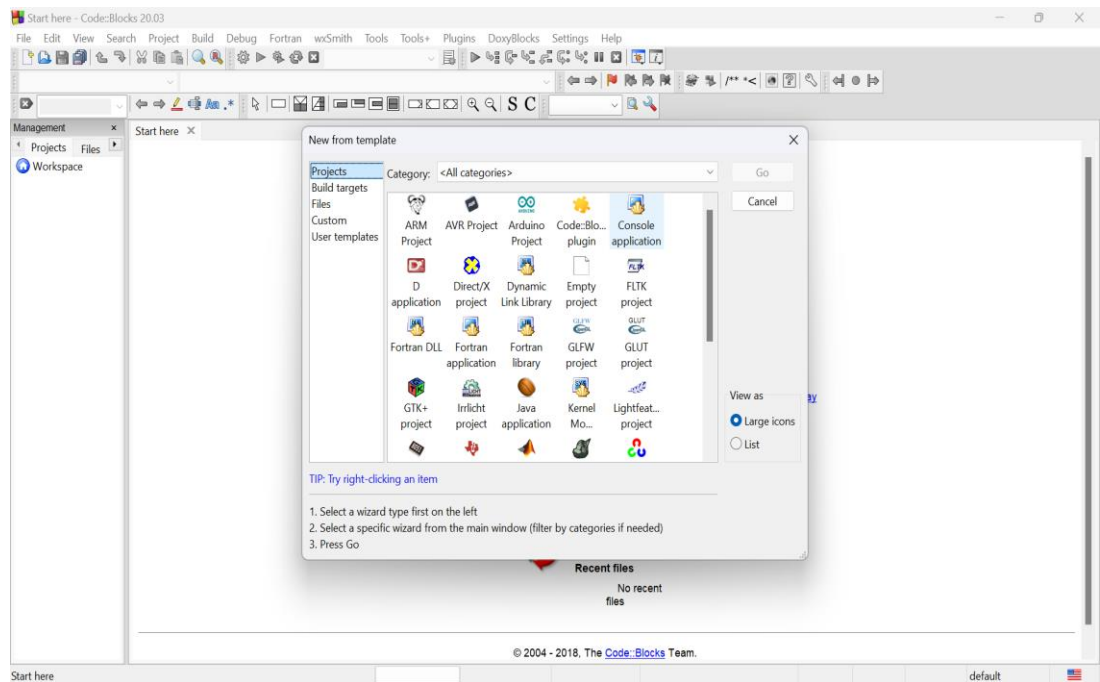
NOTE: The codeblocks-20.03mingw-setup.exe file includes additionally the GCC/G++/GFortran compiler and GDB debugger from [MinGW-W64 project](#) (version 8.1.0, 32/64 bit, SEH).

3. Setelah selesai maka code block bisa digunakan

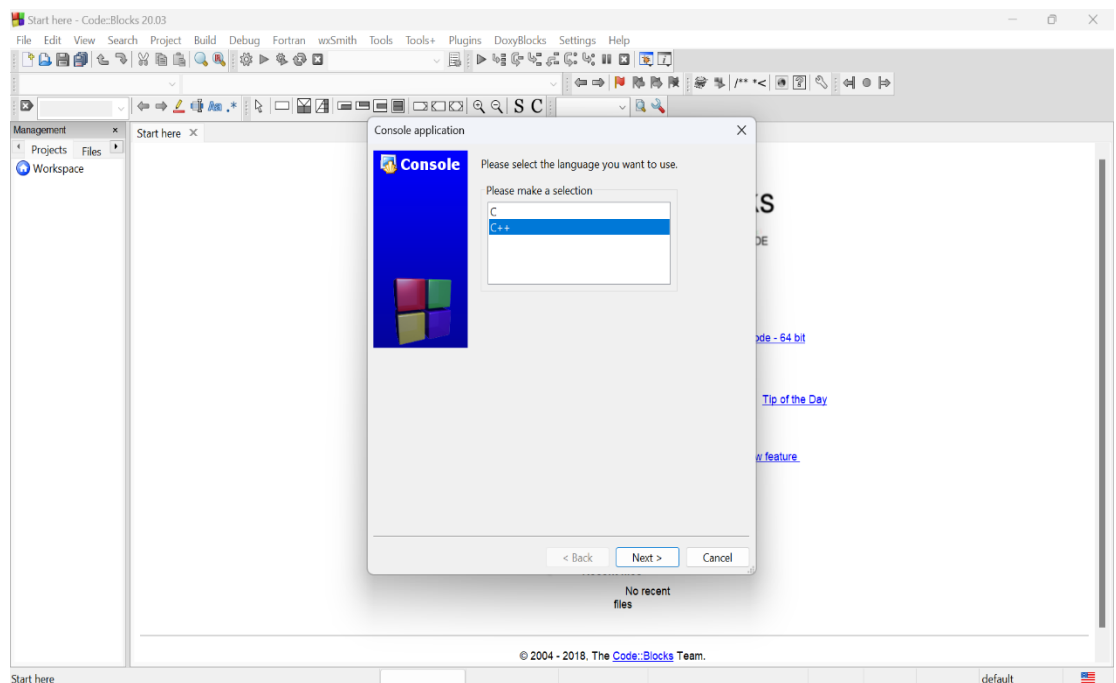


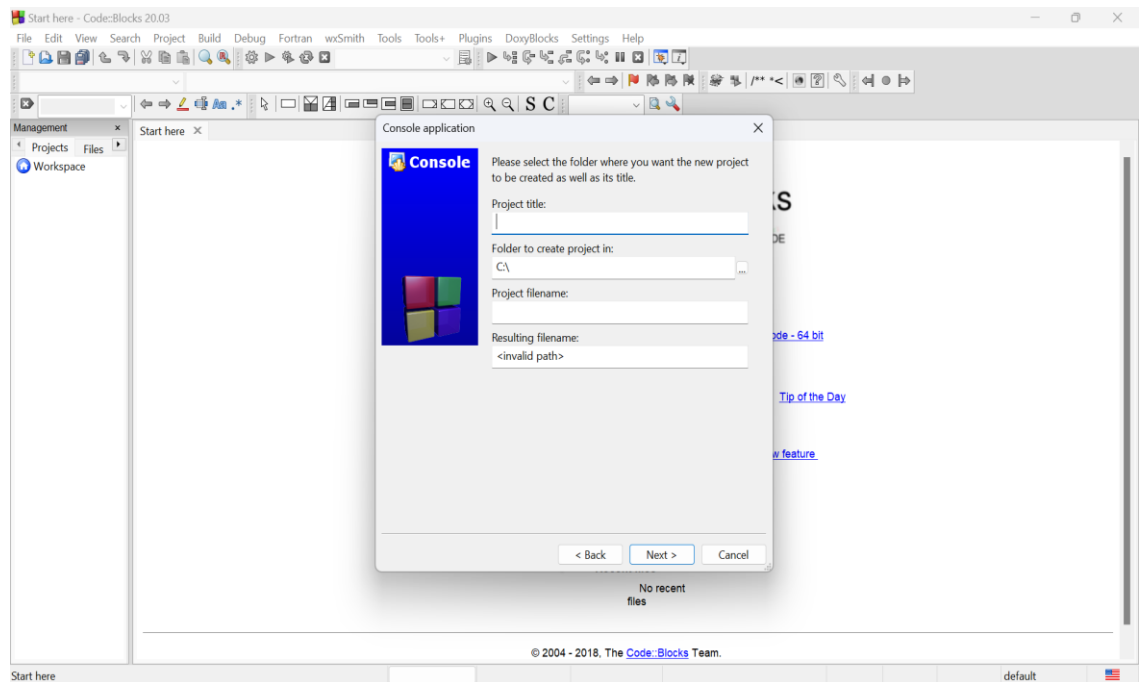
Cara menggunakan code block :

1. Pencet create project dan pilih console application

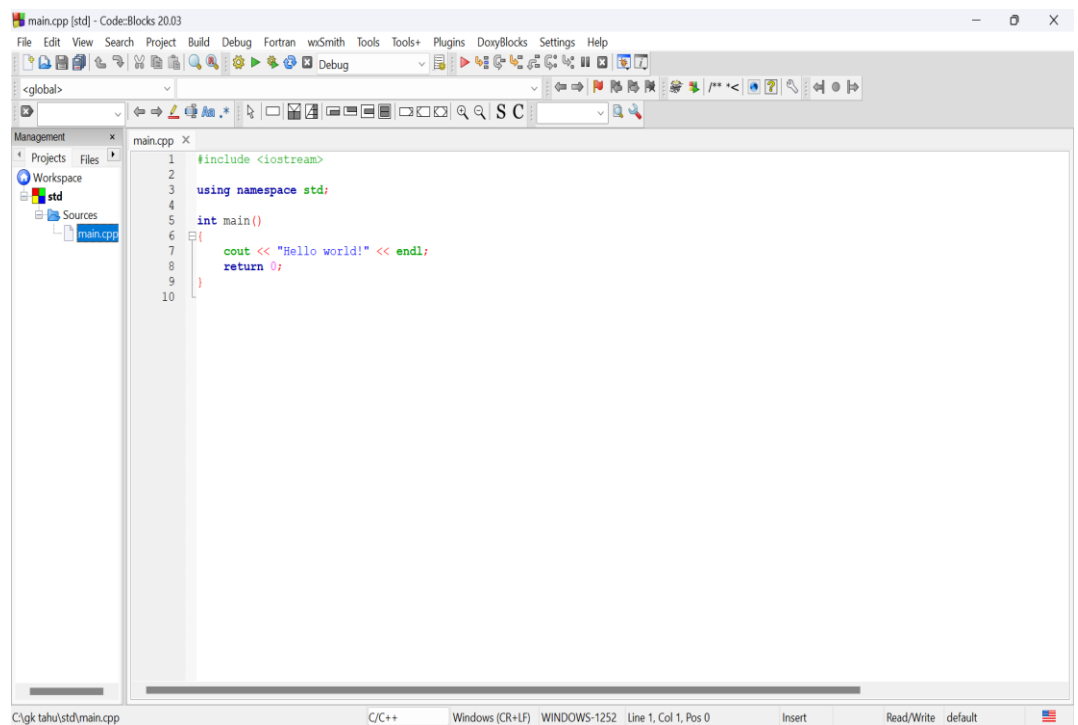


2. Karna kita belajar bahasa C++ maka pilih bahasa C++ lalu pencet next setelah itu kita pilih file mau disimpan dimana





3. Setelah pilih file untuk disimpan maka bisa langsung digunakan



1.Deklarasi Variabel



```
1  int angka = 10;  
2      float desimal = 10.5;  
3      string nama = "haza";  
4      double tinggi = 10.4;  
5      char jenis_kelamin = 'L';  
6      bool isSunny = true;
```

int angka: Mendeklarasikan variabel angka bertipe integer dengan nilai 10.

float desimal: Mendeklarasikan variabel desimal bertipe float dengan nilai 10.5.

string nama: Mendeklarasikan variabel nama bertipe string dengan nilai "haza".

double tinggi: Mendeklarasikan variabel tinggi bertipe double dengan nilai 10.4.

char jenis_kelamin: Mendeklarasikan variabel jenis_kelamin bertipe karakter dengan nilai 'L'.

bool isSunny: Mendeklarasikan variabel boolean isSunny dengan nilai true.

2. Menginput dan Mencetak Nilai dari Pengguna



```
1  int angka;  
2  
3      cout << "Masukkan angka: ";  
4      cin >> angka;  
5  
6      cout << "Angka: " << angka << endl;
```

cin >> angka: Mengambil input dari pengguna dan menyimpannya dalam variabel angka.

cout << "Angka: " << angka: Menampilkan angka yang diinput oleh pengguna ke layar

3. Operator Aritmatika



```
1  int angka1 = 10;  
2      int angka2 = 2;  
3  
4      int hasil = angka1 % angka2;  
5  
6      cout << "=" << hasil << endl;
```

% adalah operator modulus, yang menghasilkan sisa bagi. Di sini, $10 \% 2$ menghasilkan 0 karena 10 habis dibagi 2.

4. operator perbandingan



```
1  int angka1 = 5;
2      int angka2 = 5;
3
4      bool hasil = (angka1 == angka2);
5
6      cout << "= " << hasil << endl;
```

5. Operator logika

Logika or:



```
1  bool kondisi1 = false;
2      bool kondisi2 = false;
3
4      bool hasil = (kondisi1 || kondisi2);
5      cout << "= " << boolalpha << hasil << endl;
```

|| (OR) menghasilkan true jika salah satu dari kedua kondisi adalah true. Di sini, kedua kondisi false, jadi hasilnya adalah false.

boolalpha memastikan boolean dicetak sebagai true atau false, bukan 1 atau 0

logika not :



```
1 bool kondisi1 = false;
2     bool hasil = !kondisi1;
3     cout << "= " << boolalpha << hasil << endl;
```

(NOT) membalikkan nilai boolean. Di sini, !false menjadi true

6. Percabangan (If-Else)

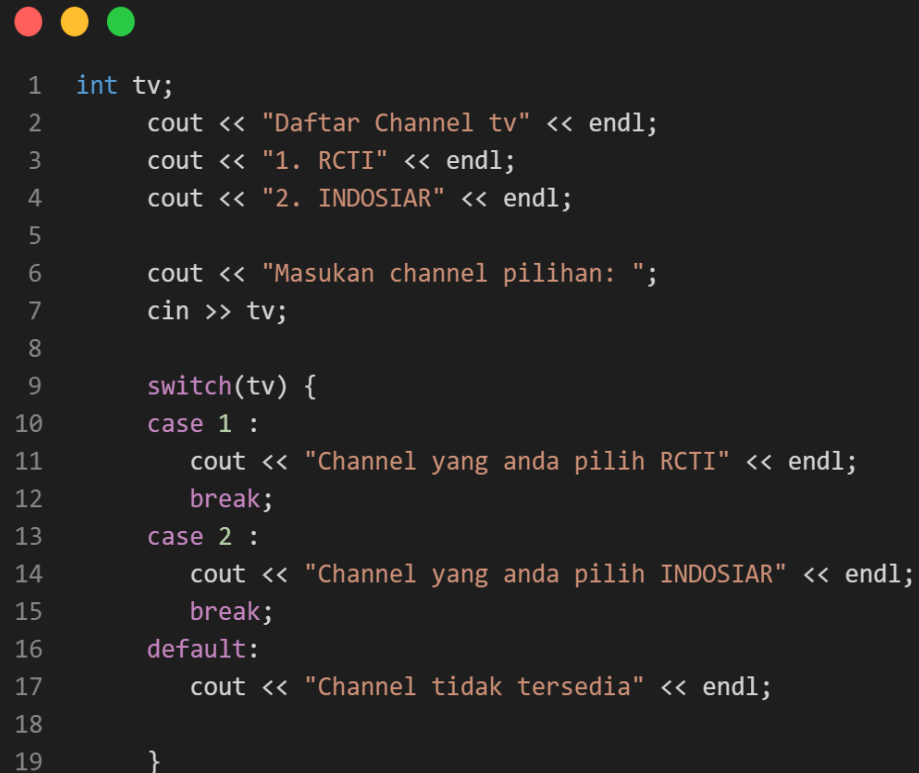


```
1 string kata;
2     cout << "Masukkan kata = HALO" << endl;
3     cin >> kata;
4
5     if(kata == "HALO"){
6         cout << "Kata sesuai" << endl;
7     }
8
9     else {
10         cout << "Kata tidak sesuai" << endl;
11     }
```

if(kata == "HALO"): Memeriksa apakah input pengguna sama dengan "HALO".

Jika kondisi benar, mencetak "Kata sesuai", jika salah mencetak "Kata tidak sesuai".

7. Switch Case



```
1  int tv;
2      cout << "Daftar Channel tv" << endl;
3      cout << "1. RCTI" << endl;
4      cout << "2. INDOSIAR" << endl;
5
6      cout << "Masukan channel pilihan: ";
7      cin >> tv;
8
9      switch(tv) {
10     case 1 :
11         cout << "Channel yang anda pilih RCTI" << endl;
12         break;
13     case 2 :
14         cout << "Channel yang anda pilih INDOSIAR" << endl;
15         break;
16     default:
17         cout << "Channel tidak tersedia" << endl;
18
19 }
```

switch(tv): Menggunakan nilai input tv untuk memilih salah satu dari beberapa pilihan (RCTI atau INDOSIAR).

Jika tv adalah 1, menampilkan "Channel yang anda pilih RCTI". Jika tv adalah 2, menampilkan "Channel yang anda pilih INDOSIAR". Jika nilai lain, mencetak "Channel tidak tersedia".

8. Perulangan For



```
1  int i;  
2      for (i=0; i<5; i++){  
3          cout << "Hello world" << endl;  
4      }  
5  
6  
7      for(int i = 0; i<5; i++){  
8          cout << i+1 << " Hello world" << endl;  
9      }
```

for (i=0; i<5; i++): Loop akan dijalankan 5 kali (dari i=0 sampai i=4).

Pada loop pertama, hanya mencetak "Hello world".

Pada loop kedua, mencetak "Hello world" dengan nomor urut, mulai dari 1 hingga 5.

V. UNGUIDED :



```
1  #include <iostream>
2  using namespace std;
3
4  int main() {
5      float num1, num2;
6
7      // Input dua bilangan float
8      cout << "Masukkan dua bilangan float: ";
9      cin >> num1 >> num2;
10
11     // Operasi aritmatika
12     cout << "Penjumlahan: " << num1 + num2 << endl;
13     cout << "Pengurangan: " << num1 - num2 << endl;
14     cout << "Perkalian: " << num1 * num2 << endl;
15     cout << "Pembagian: " << num1 / num2 << endl;
16
17     return 0;
18 }
19
```

```
"C:\gk tahu\std\bin\Debug\st" X + -
Masukkan dua bilangan float: 5.5 2.2
Penjumlahan: 7.7
Pengurangan: 3.3
Perkalian: 12.1
Pembagian: 2.5

Process returned 0 (0x0)   execution time : 15.746 s
Press any key to continue.
```

1. `cin >> num1 >> num2;` meminta dua input dari pengguna untuk bilangan float.
2. Program kemudian menampilkan hasil operasi aritmatika untuk bilangan float tersebut:

Penjumlahan: Hasil dari $\text{num1} + \text{num2}$.

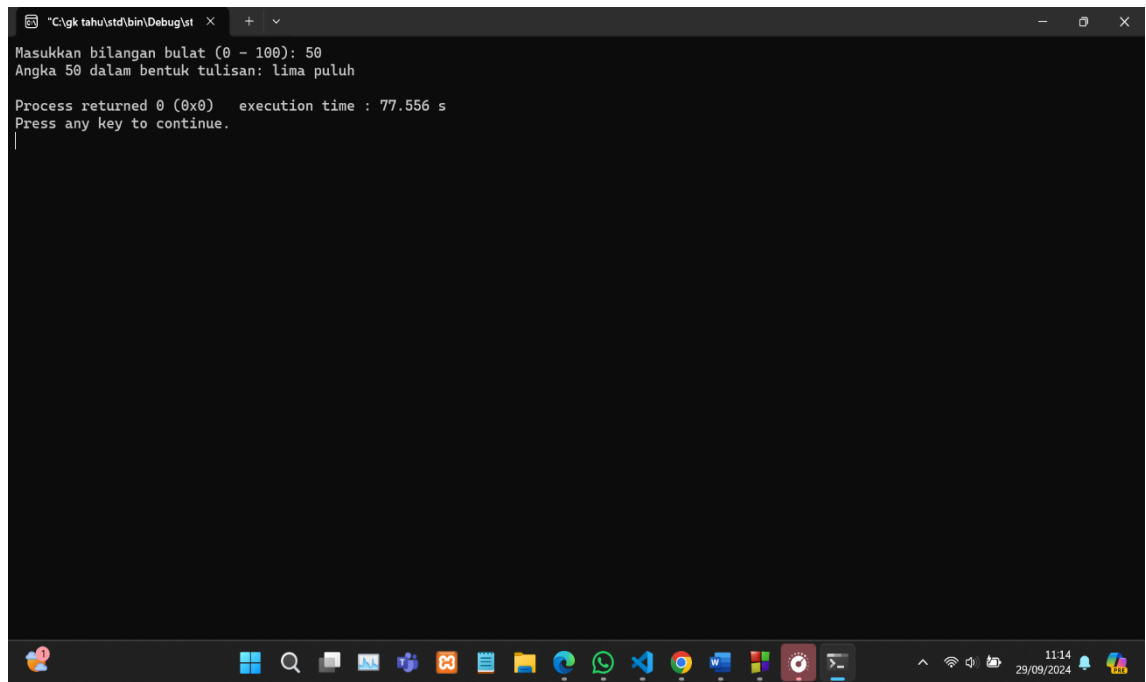
Pengurangan: Hasil dari $\text{num1} - \text{num2}$.

Perkalian: Hasil dari $\text{num1} * \text{num2}$.

Pembagian: Hasil dari $\text{num1} / \text{num2}$ (dengan asumsi bahwa num2 tidak nol, karena pembagian dengan nol tidak terdefinisi).

Program ini akan terus berjalan dengan baik selama num2 tidak bernilai nol. Jika num2 adalah nol, hasil pembagian akan menyebabkan masalah.

```
1  #include <iostream>
2  using namespace std;
3
4  string convertToWords(int num) {
5      string angka[] = { "nol", "satu", "dua", "tiga", "empat", "lima", "enam", "tujuh", "delapan", "sembilan",
6                          "sepuluh", "sebelas", "dua belas", "tiga belas", "empat belas", "lima belas", "enam belas",
7                          "tujuh belas", "delapan belas", "sembilan belas" };
8      string puluhan[] = { "", "", "dua puluh", "tiga puluh", "empat puluh", "lima puluh",
9                            "enam puluh", "tujuh puluh", "delapan puluh", "sembilan puluh" };
10
11     if (num < 20)
12         return angka[num];
13     else if (num <= 99) {
14         int tens = num / 10;
15         int ones = num % 10;
16         return puluhan[tens] + (ones != 0 ? " " + angka[ones] : "");
17     } else if (num == 100)
18         return "seratus";
19
20     return "";
21 }
22
23 int main() {
24     int num;
25
26     // Input bilangan bulat dari 0 hingga 100
27     cout << "Masukkan bilangan bulat (0 - 100): ";
28     cin >> num;
29
30     if (num < 0 || num > 100) {
31         cout << "Masukkan angka dalam rentang 0 s.d. 100!" << endl;
32     } else {
33         // Konversi angka ke teks
34         cout << "Angka " << num << " dalam bentuk tulisan: " << convertToWords(num) << endl;
35     }
36
37     return 0;
38 }
39
```



```
"C:\gk tahu\std\bin\Debug\st" X + v
Masukkan bilangan bulat (0 - 100): 50
Angka 50 dalam bentuk tulisan: lima puluh
Process returned 0 (0x0)   execution time : 77.556 s
Press any key to continue.
```

1. Angka menyimpan representasi kata-kata untuk bilangan dari 0 hingga 19.
2. Puluhan menyimpan kata-kata untuk puluhan (20, 30, 40, dll.).

Jika bilangan kurang dari 20, ia akan langsung mengakses array angka[].

Jika bilangan lebih dari atau sama dengan 20 tetapi kurang dari 100, ia akan mengambil kata dari array puluhan[] dan menambahkan angka satuan jika diperlukan.

Bilangan 100 dikonversi secara langsung menjadi "seratus".



```
1  #include <iostream>
2  using namespace std;
3
4  int main() {
5      int n;
6
7      // Input
8      cout << "Masukkan angka: ";
9      cin >> n;
10
11     // Loop untuk mencetak pola
12     for (int i = n; i >= 1; i--) {
13         // Bagian angka menurun
14         for (int j = i; j >= 1; j--) {
15             cout << j << " ";
16         }
17
18         // Cetak tanda bintang
19         cout << "* ";
20
21         // Bagian angka menaik
22         for (int j = 1; j <= i; j++) {
23             cout << j << " ";
24         }
25
26         cout << endl; // Ganti baris setelah setiap iterasi
27     }
28
29     // Cetak tanda 'x' di akhir
30     cout << "x" << endl;
31
32     return 0;
33 }
34
```

```
"C:\gk tahu\std\bin\Debug\st" X + v
Masukkan angka: 5
5 4 3 2 1 * 1 2 3 4 5
4 3 2 1 * 1 2 3 4
3 2 1 * 1 2 3
2 1 * 1 2
1 * 1
x
Process returned 0 (0x0)   execution time : 2.292 s
Press any key to continue.
```

1. Pada iterasi pertama ($i = 5$), angka menurun dicetak dari 5 hingga 1, diikuti oleh tanda bintang *, kemudian angka menaik dicetak dari 1 hingga 5.
2. Pada iterasi kedua ($i = 4$), angka menurun dari 4 hingga 1, diikuti bintang *, lalu angka menaik dari 1 hingga 4.
3. Pola ini terus berulang hingga iterasi terakhir ketika $i = 1$.
4. Setelah seluruh pola selesai, huruf "x" dicetak pada baris terakhir sebagai tanda akhir program

VI. KESIMPULAN

C++ adalah bahasa pemrograman yang kuat dan fleksibel, cocok untuk berbagai jenis proyek. **Code::Blocks** adalah IDE yang sangat baik untuk mengembangkan aplikasi C++. Kombinasi keduanya memungkinkan programmer untuk menciptakan aplikasi yang performanya tinggi dan handal.