

**TUGAS PENDAHULUAN MODUL 1**  
**STRUKTUR DATA**



Disusun Oleh :

Nurul Maulina Nainggolan

21104053/SE07-01

**PROGRAM STUDI REKAYASA PERANGKAT LUNAK**

**FAKULTAS INFORMATIKA**

**TELKOM UNIVERAITY PURWOKERTO**

**2024**

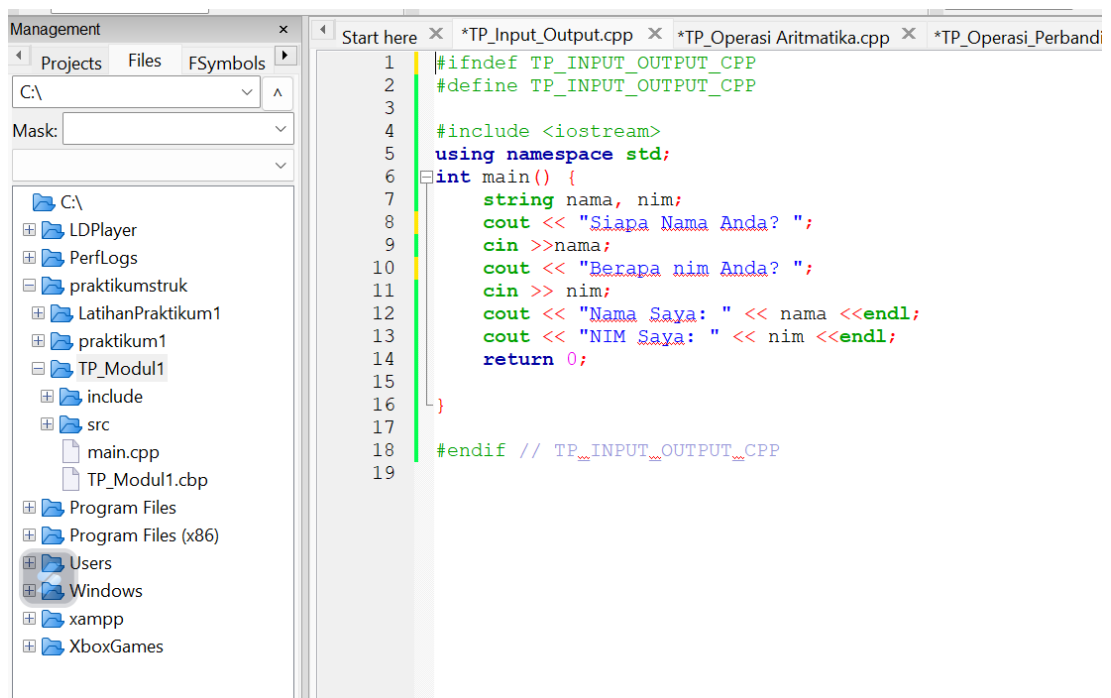
**MODUL 1**

**TUGAS PENDAHULUAN MODUL 1**  
**STRUKTUR DATA- GANJIL 2024/2025**  
**“Pengenalan C++ : Subprogram & Array”**

**1. (Input/Output)**

- a) Masukkan nama lengkap Anda dan nim Anda. Screenshot kode dan hasilnya, lalu tempelkan pada jawaban.
- b) Masukkan nama pertama Anda dan nim Anda. Screenshot kode dan hasilnya, lalu tempelkan pada jawaban.

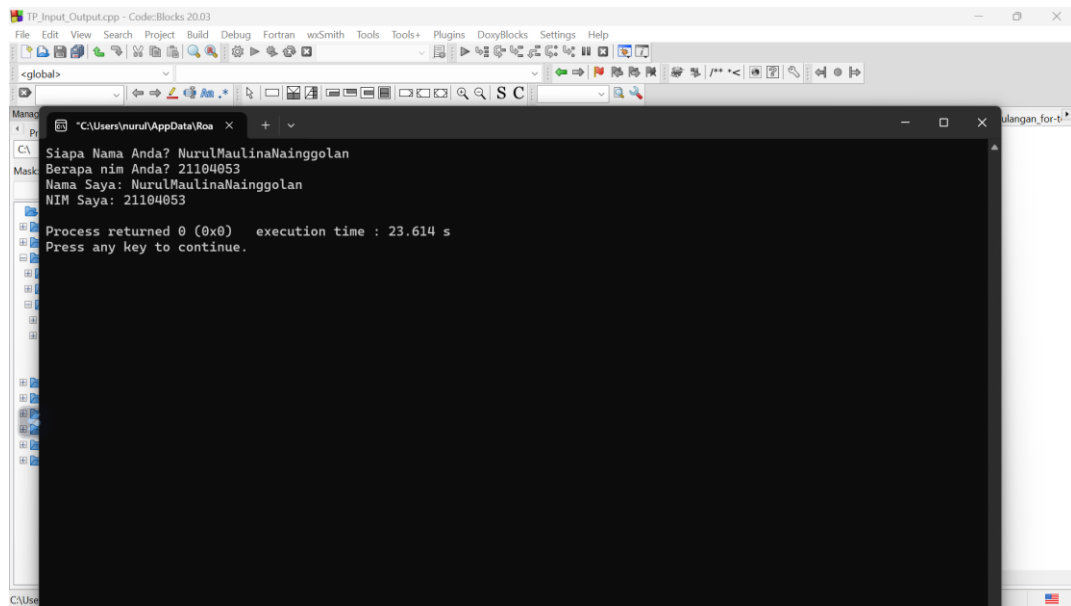
**Pendeklarasian :**



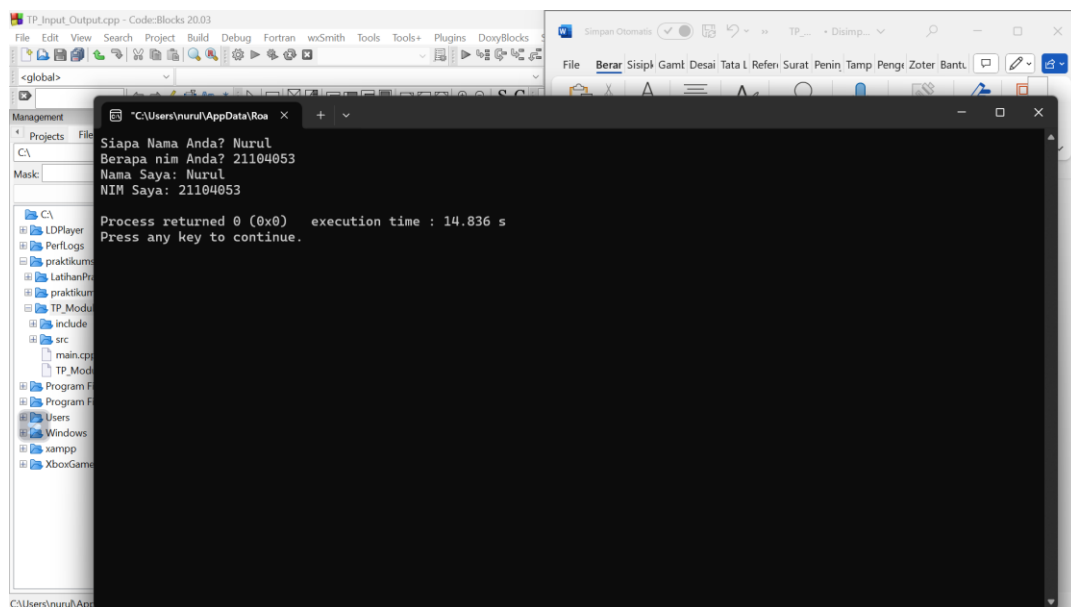
```
1 |#ifndef TP_INPUT_OUTPUT_CPP
2 |#define TP_INPUT_OUTPUT_CPP
3 |
4 |#include <iostream>
5 |using namespace std;
6 |int main() {
7 |    string nama, nim;
8 |    cout << "Siapa Nama Anda? ";
9 |    cin >> nama;
10 |    cout << "Berapa nim Anda? ";
11 |    cin >> nim;
12 |    cout << "Nama Saya: " << nama << endl;
13 |    cout << "NIM Saya: " << nim << endl;
14 |    return 0;
15 |}
16 |
17 |
18 |#endif // TP_INPUT_OUTPUT_CPP
19 |
```

Gambar 1. Output Nama Depan

**Hasil Run :**



Gambar 2. Output Nama Lengkap



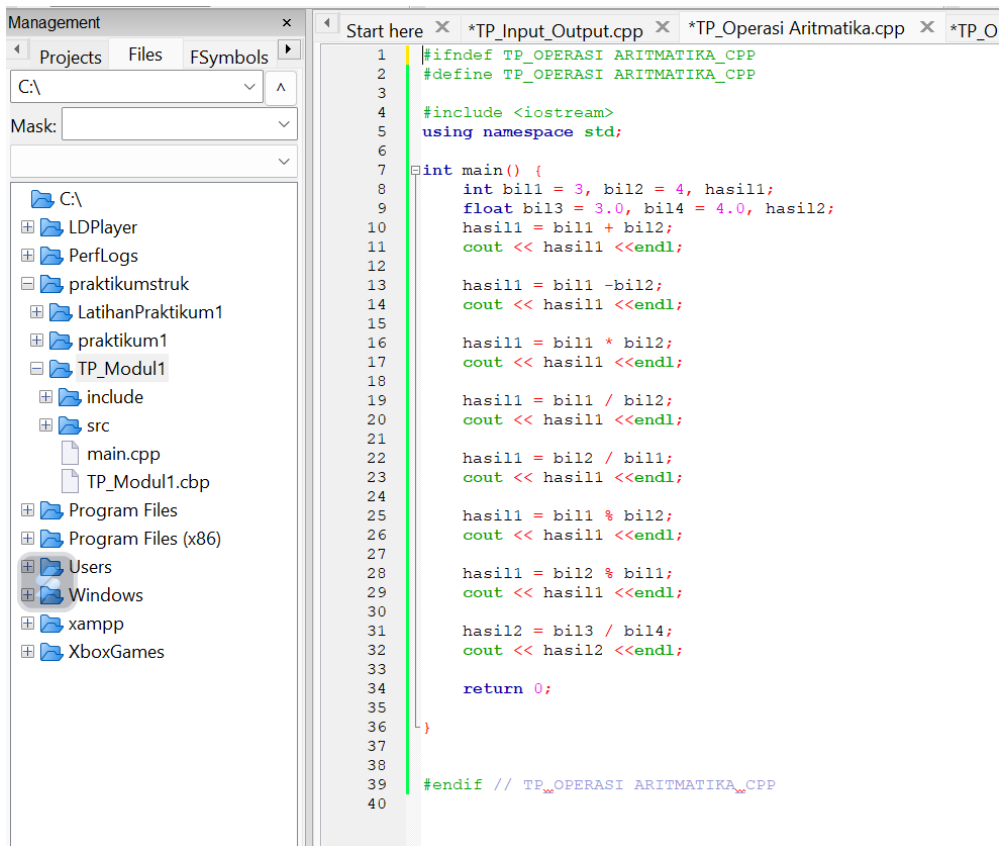
Gambar 3. Output Nama Depan

Program ini menunjukkan cara melakukan input dan output dalam c++. Mahasiswa diminta untuk memasukkan nama dan nim yang ditampilkan kembali. Pada program ini menggunakan **cout** untuk menampilkan pertanyaan kepada pengguna dan **cin** untuk mengambil input.

## 2. (Operasi Aritmatika)

Tuliskan kode berikut dan jalankan. Screenshoot kode dan hasilnya, lalu tempelkan pada jawaban.

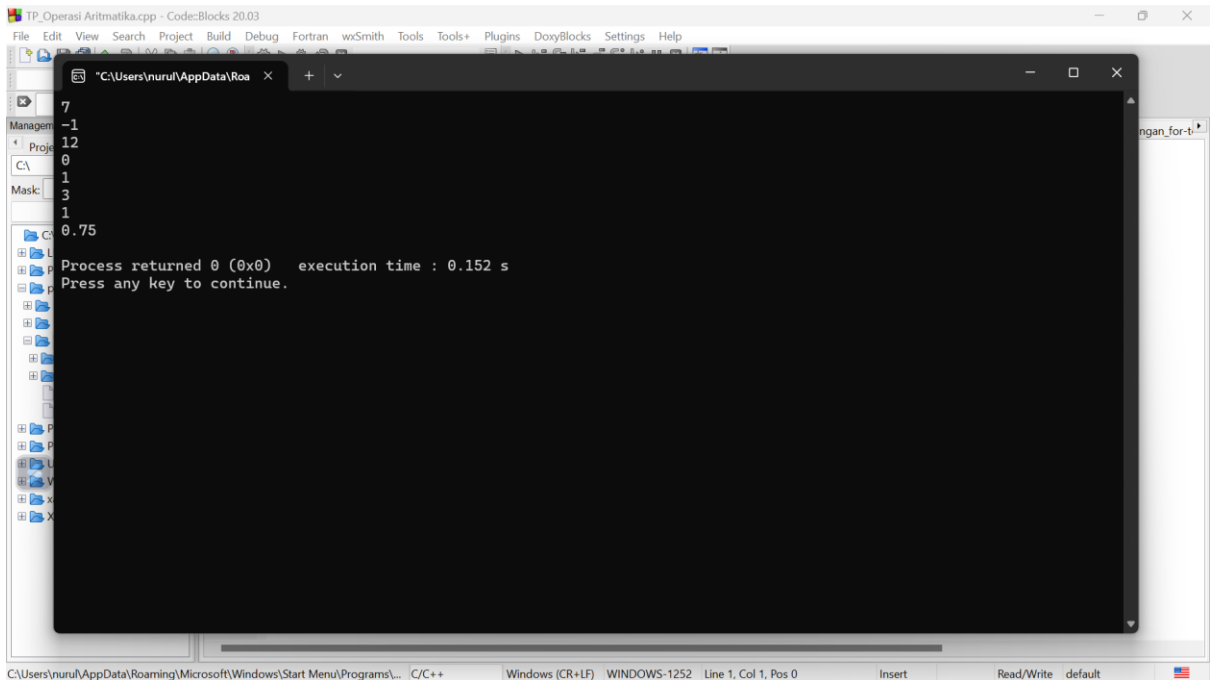
## Pendklarasian :



```
1 #ifndef TP_OPERASI_ARITMATIKA_CPP
2 #define TP_OPERASI_ARITMATIKA_CPP
3
4 #include <iostream>
5 using namespace std;
6
7 int main() {
8     int bil1 = 3, bil2 = 4, hasil1;
9     float bil3 = 3.0, bil4 = 4.0, hasil2;
10    hasil1 = bil1 + bil2;
11    cout << hasil1 << endl;
12
13    hasil1 = bil1 - bil2;
14    cout << hasil1 << endl;
15
16    hasil1 = bil1 * bil2;
17    cout << hasil1 << endl;
18
19    hasil1 = bil1 / bil2;
20    cout << hasil1 << endl;
21
22    hasil1 = bil2 / bil1;
23    cout << hasil1 << endl;
24
25    hasil1 = bil1 % bil2;
26    cout << hasil1 << endl;
27
28    hasil1 = bil2 % bil1;
29    cout << hasil1 << endl;
30
31    hasil2 = bil3 / bil4;
32    cout << hasil2 << endl;
33
34    return 0;
35 }
36
37
38
39 #endif // TP_OPERASI_ARITMATIKA_CPP
40
```

Gambar 4. Kode Operasi Aritmatika

## Hasil Run :



```
TP_Operasi Aritmatika.cpp - Code::Blocks 20.03
File Edit View Search Project Build Debug Fortran wxSmith Tools Tools+ Plugins DoxyBlocks Settings Help
"C:\Users\nurul\AppData\Roaming\Code::Blocks\20.03\bin\codeblocks.exe"
7
-1
12
0
1
3
1
0.75
Process returned 0 (0x0)   execution time : 0.152 s
Press any key to continue.
```

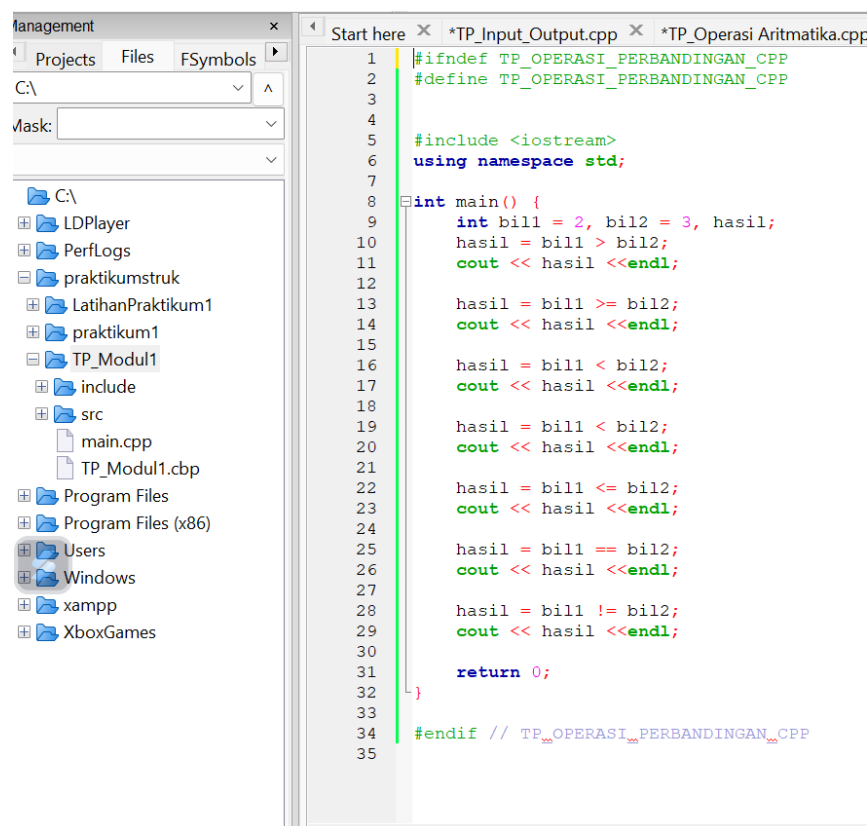
Gambar 5. Output Operasi Aritmatika

Pada program ini menggunakan variabel bertipe **float** dan **integer**. Float digunakan untuk operasi aritmatik untuk bilangan desimal sedangkan integer untuk operasi aritmatika bilangan bulat. Cout untuk menampilkan hasil perhitungan kepada pengguna. Ini adalah cara standar di C++ untuk mencetak output

### 3. (Operasi Perbandingan)

Tuliskan kode berikut dan jalankan. Screenshot kode dan hasilnya, lalu tempelkan pada jawaban.

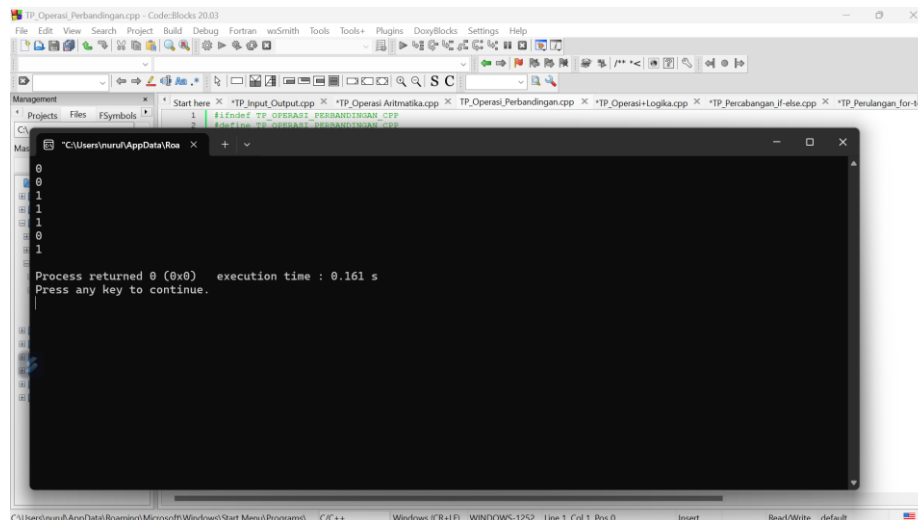
**Pendeklarasian :**



```
1  #ifndef TP_OPERASI_PERBANDINGAN_CPP
2
3  #define TP_OPERASI_PERBANDINGAN_CPP
4
5  #include <iostream>
6  using namespace std;
7
8  int main() {
9      int bil1 = 2, bil2 = 3, hasil;
10     hasil = bil1 > bil2;
11     cout << hasil << endl;
12
13     hasil = bil1 >= bil2;
14     cout << hasil << endl;
15
16     hasil = bil1 < bil2;
17     cout << hasil << endl;
18
19     hasil = bil1 <= bil2;
20     cout << hasil << endl;
21
22     hasil = bil1 <= bil2;
23     cout << hasil << endl;
24
25     hasil = bil1 == bil2;
26     cout << hasil << endl;
27
28     hasil = bil1 != bil2;
29     cout << hasil << endl;
30
31     return 0;
32 }
33
34 #endif // TP_OPERASI_PERBANDINGAN_CPP
35
```

Gambar 6. Kode Operasi Perbandingan

**Hasil Run :**



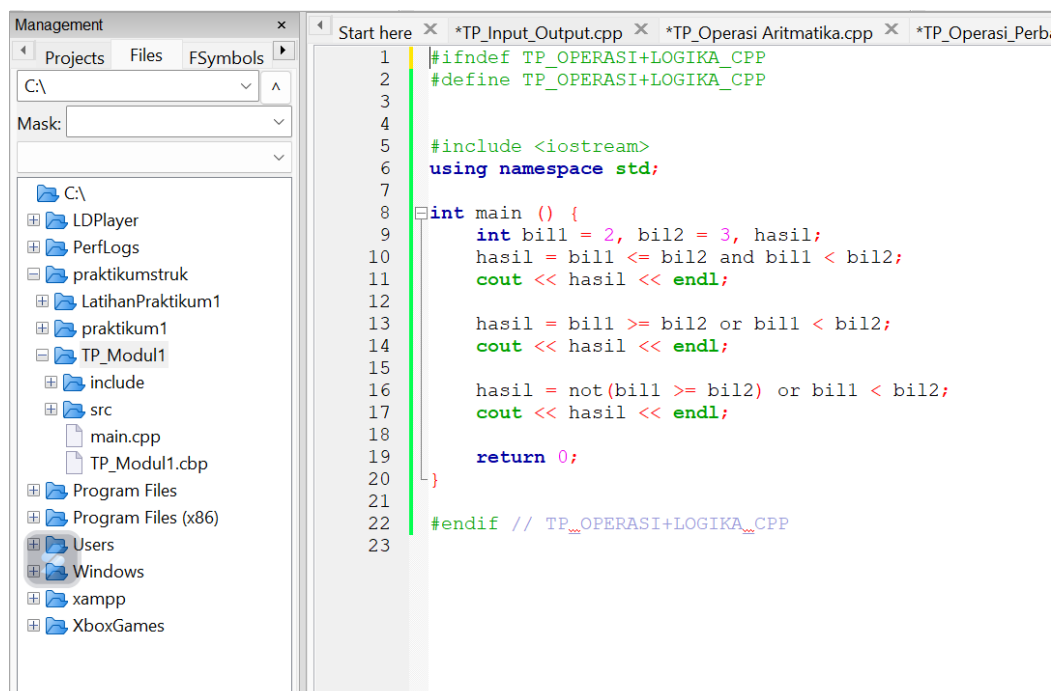
Gambar 7. Output Operasi Perbandingan

Pada program ini menggunakan dua variabel yang dideklarasikan dengan nilai masing-masing. Program ini menjelaskan operasi perbandingan dengan menggunakan operator `>`, `<`, `==`, `<=`, `>=`, `!=` untuk membandingkan dua bilangan. Fungsi `cout` digunakan untuk menampilkan hasil perbandingan ke layar. Misalnya, jika `bill1` lebih besar dari `bill2`, maka akan mencetak 0 (false) dan jika tidak, akan mencetak 1 (true)

#### 4. (Operasi Logika)

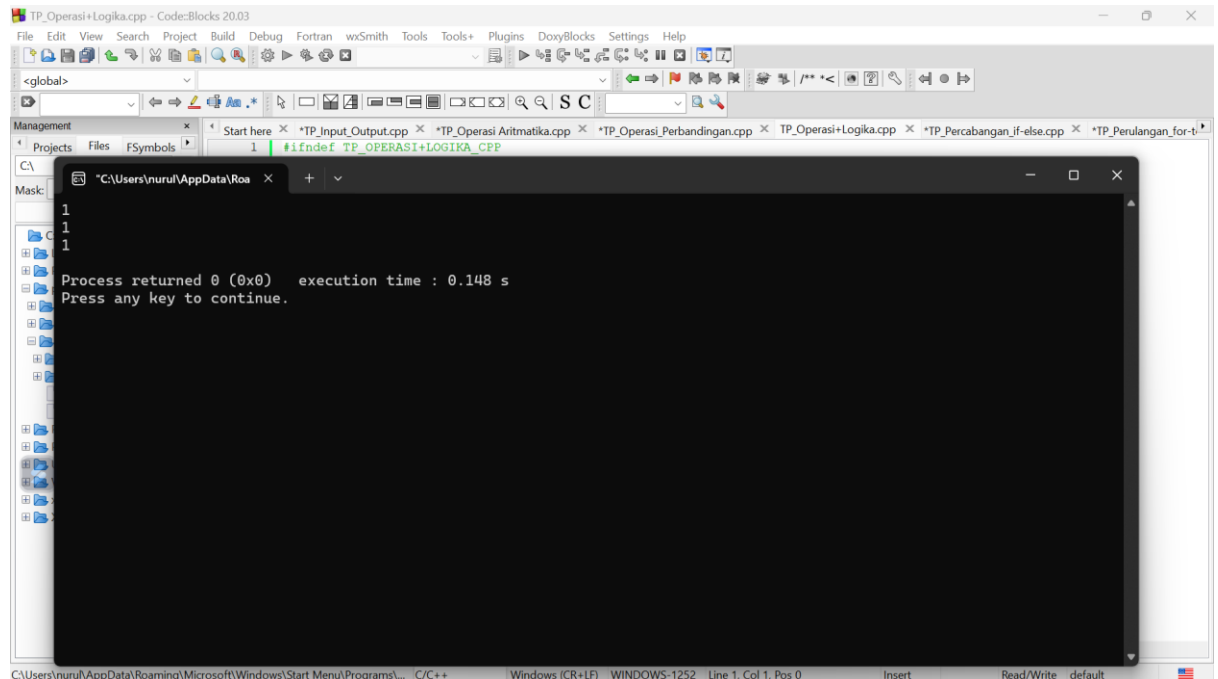
Tuliskan kode berikut dan jalankan. Screenshot kode dan hasilnya, lalu tempelkan pada jawaban.

**Pendeklarasian :**



Gambar 8. Kode Operasi Logika

### Hasil Run :



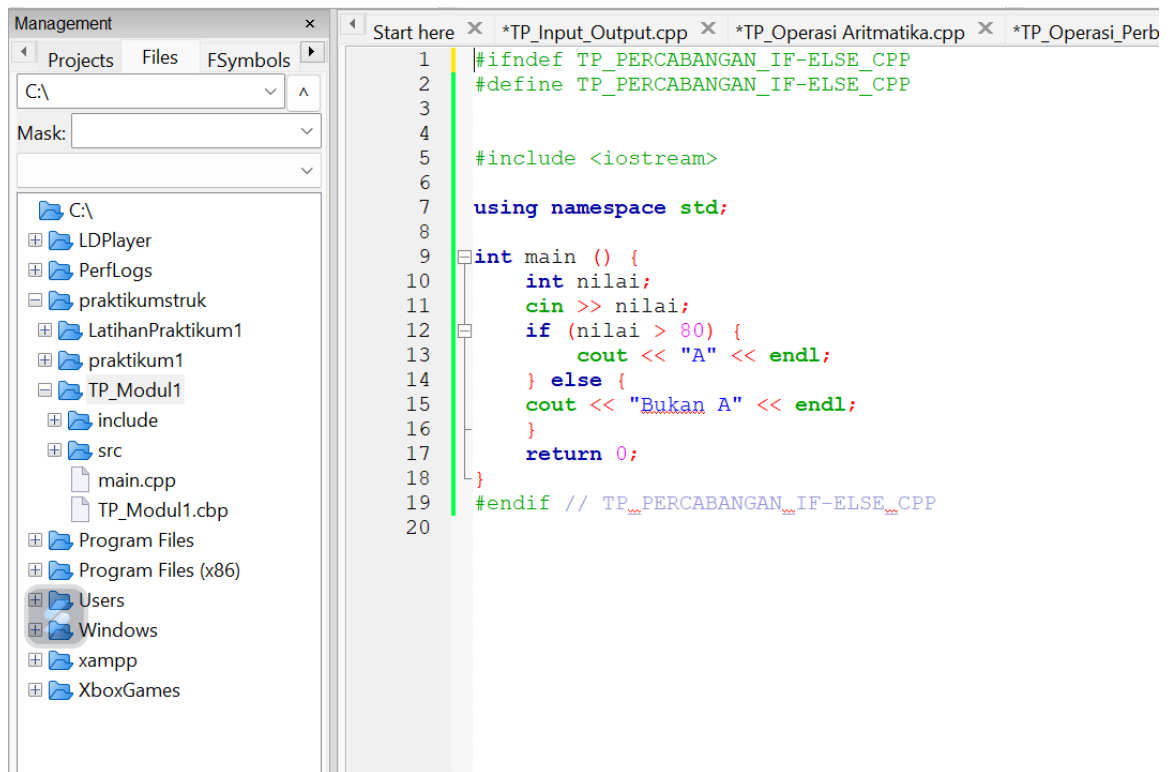
Gambar 9. Output Operasi Logika

Pada program ini menggunakan operasi logika yang merupakan operasi penggabungan nilai boolean (true or False). Disini ditampilkan jika hasil 1 maka true dan 0 maka false. Operator logika yang digunakan adalah and ( menghasilkan true jika kedua nilai bernilai true dan sebaliknya). Or (Menghasilkan true jika salah satu operand bernilai true. Hasil hanya false jika kedua operand bernilai false), dan not (Membalik nilai boolean. Jika operand bernilai true, hasilnya adalah false, dan sebaliknya). Pada program menghasilkan 1, maka dikatakan semua true.

### 5. (Percabangan if-else)

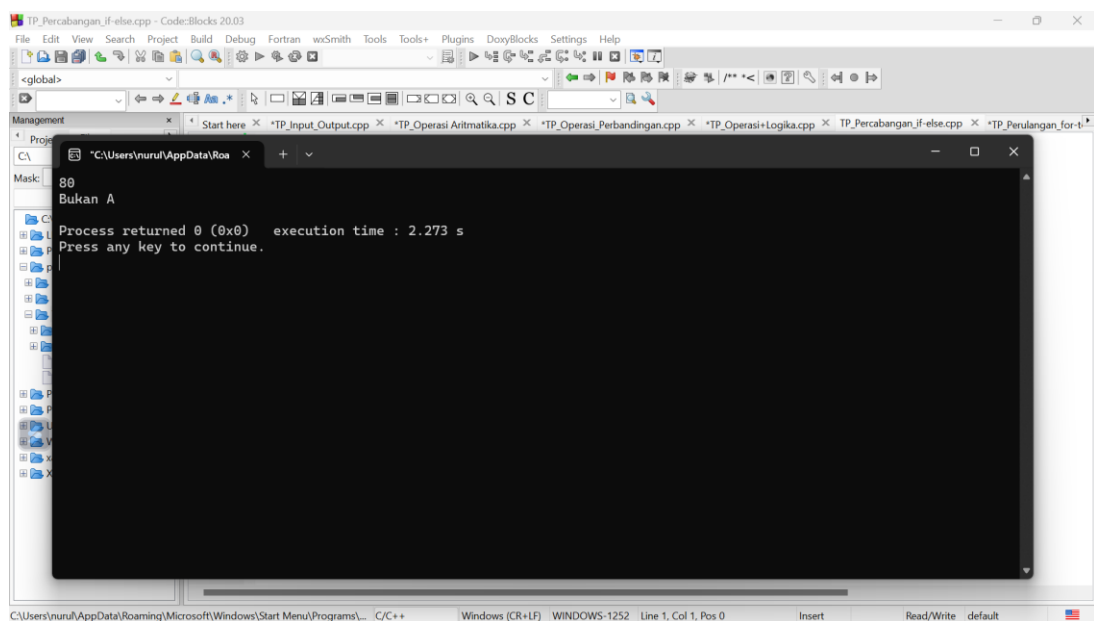
Tuliskan kode berikut dan jalankan. Masukkan input 80, 81 , dan 79. Screenshot kode dan hasilnya, lalu tempelkan pada jawaban.

### Pendeklarasian :



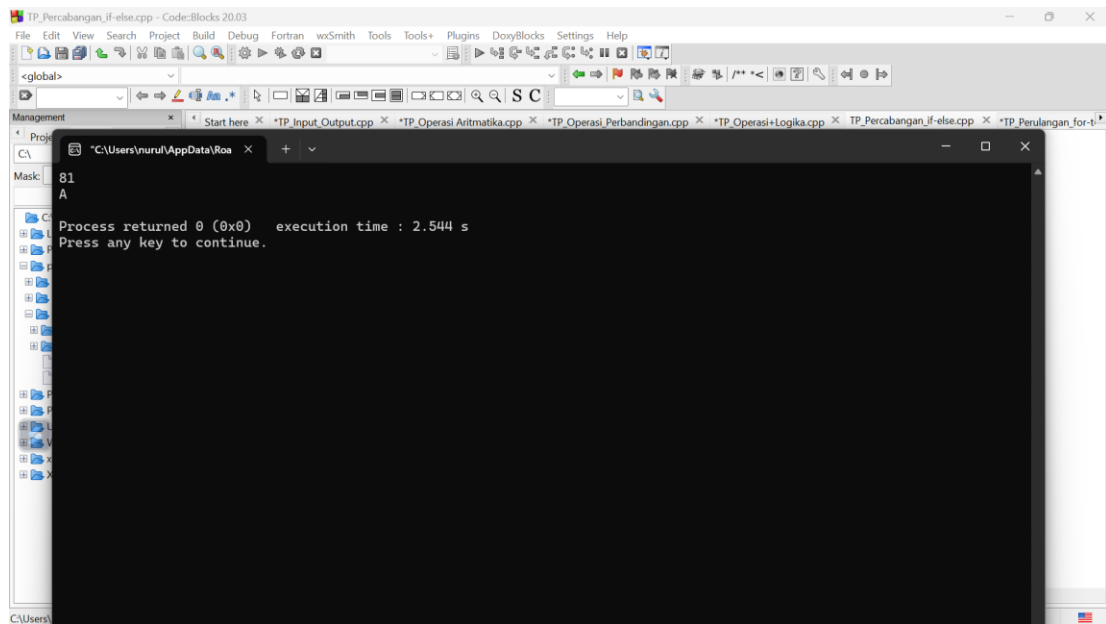
Gambar 10. Kode Percabangan if-else

### Hasil Run :

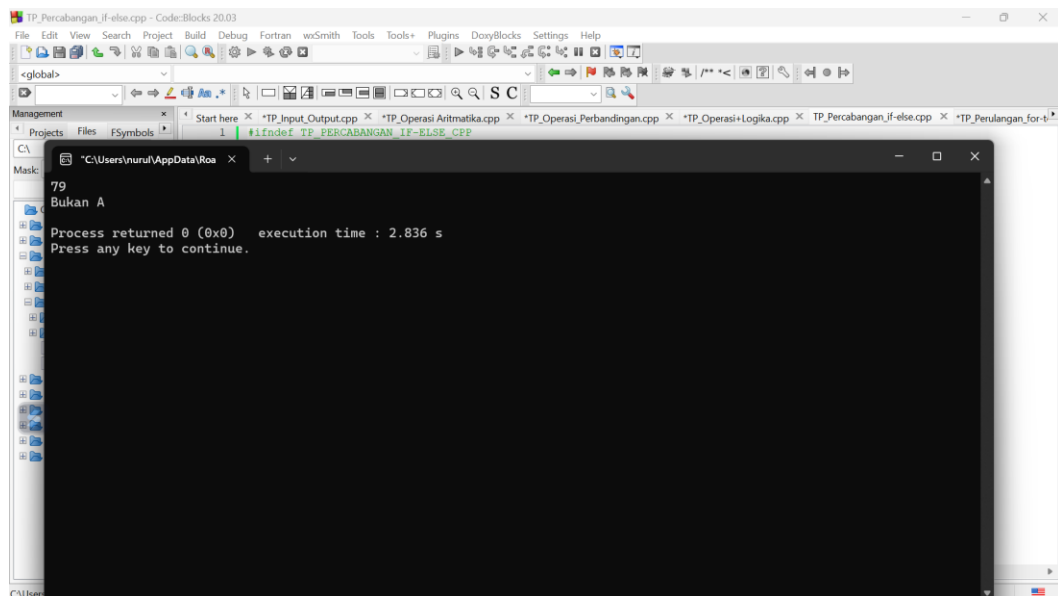


Gambar 11. Output 80 Percabangan if-else





Gambar 12. Output 81 Percabangan if-else

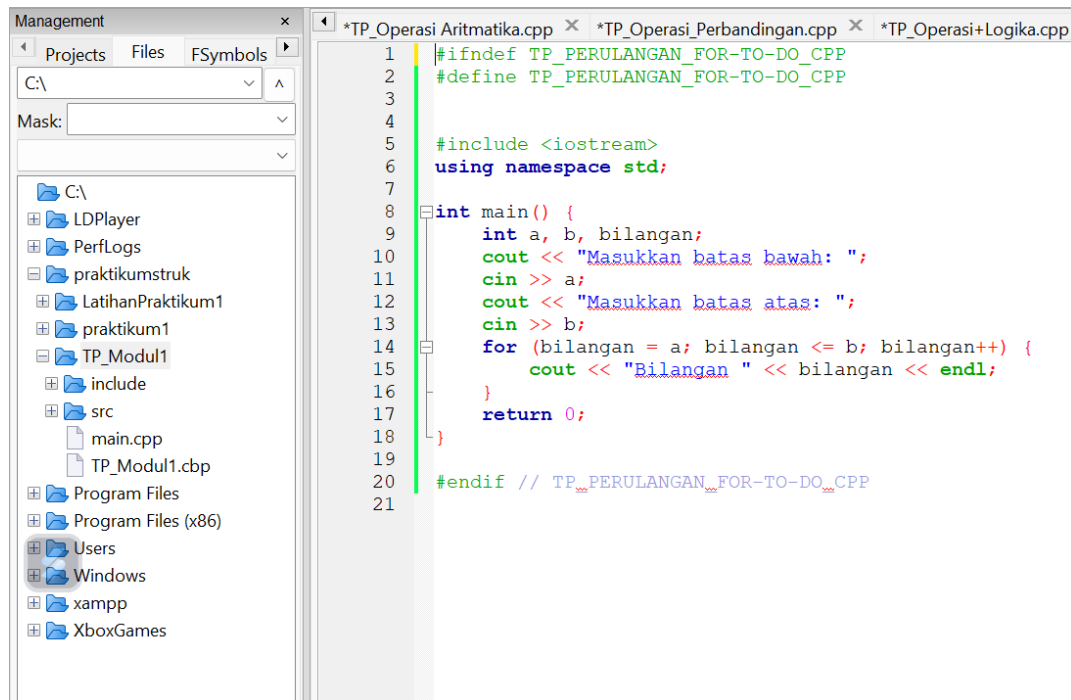


Gambar 13. Output 79 Percabangan if-else

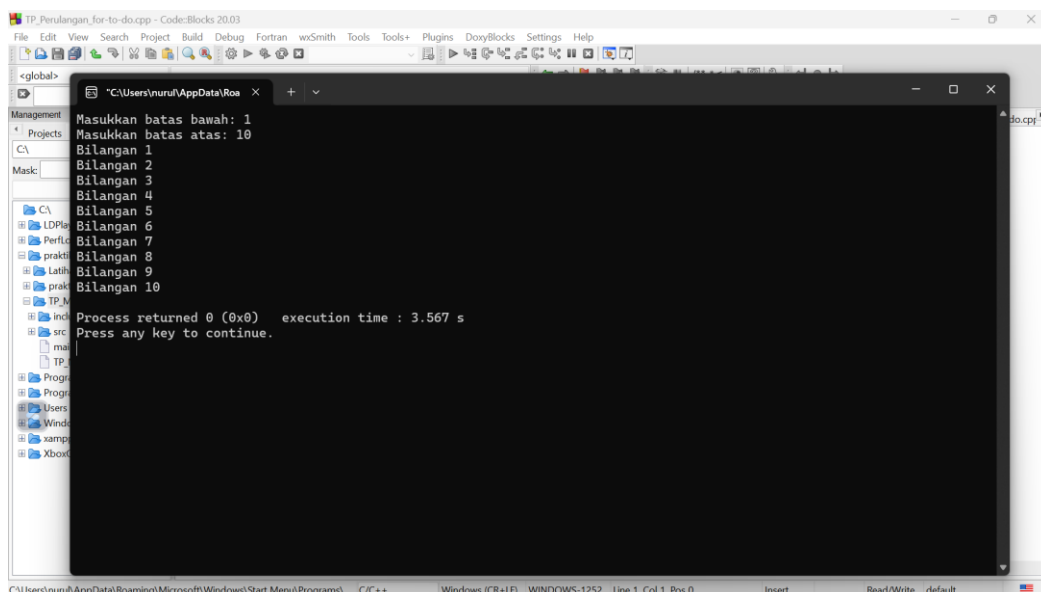
Pada program ini menjelaskan tentang if-else yang digunakan untuk memeriksa apakah nilai yang dimasukkan oleh pengguna lebih besar dari 80. If-else digunakan pada kondisi benar atau salah, dimana jika kondisi benar itu if dan salah itu else.

## 6. (Perulangan for-to-do)

Tuliskan kode berikut dan jalankan. Masukkan 1 dan 10. Screenshoot kode hasilnya, lalu tempelkan pada jawaban.



Gambar 14. Kode Perulangan For-to-do



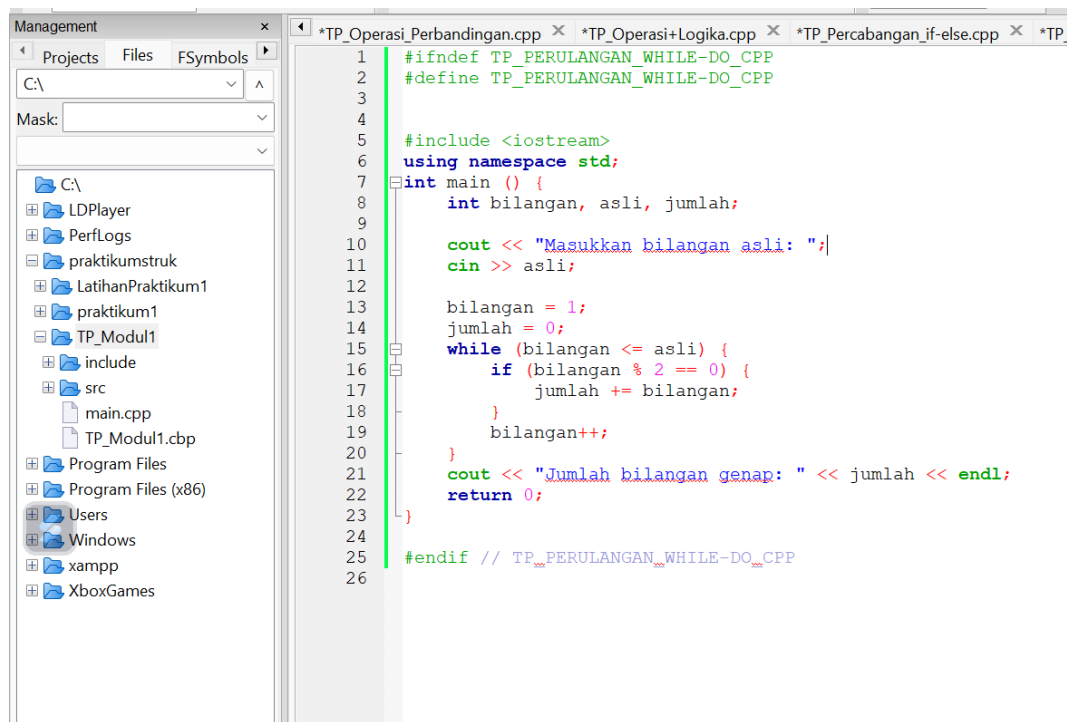
Gambar 15. Output Perulangan For-to-do

Perulangan for merupakan salah satu perulangan yang digunakan untuk melakukan tindakan berulang kali dengan sedikit variasi. Program ini mencetak semua bilangan bulat dari batas bawah hingga batas atas, termasuk kedua batas tersebut. Nilai batas menentukan bilangan pertama yang akan dicetak dan batas atas menentukan bilangan terakhir yang akan dicetak.

## 7. (Perulangan While-do)

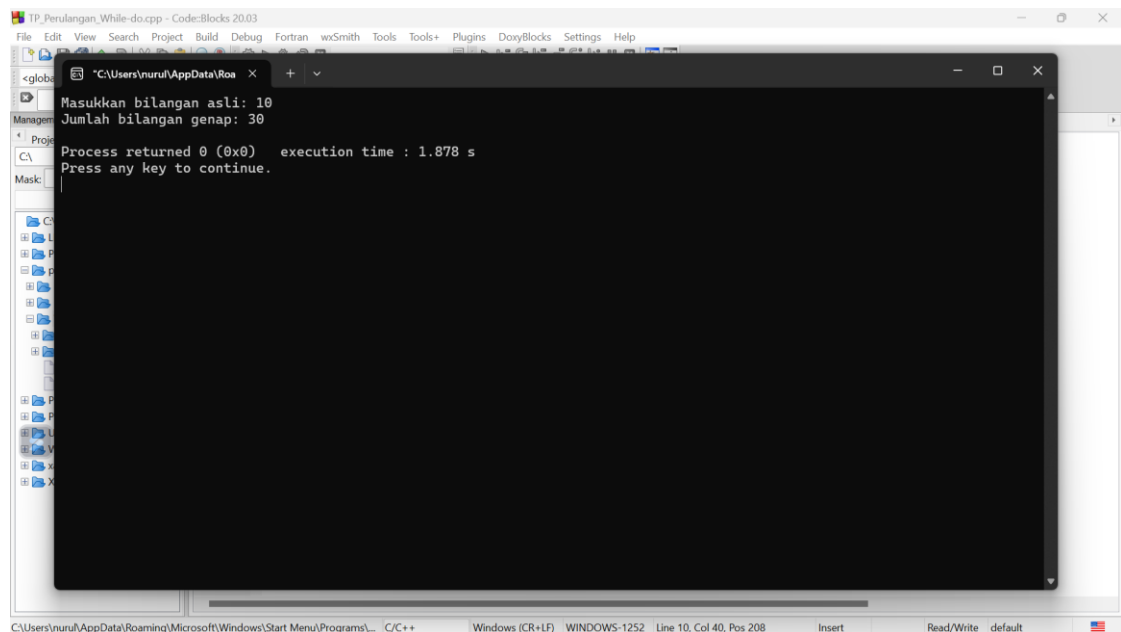
Tuliskan kode berikut dan jalankan. Masukkan pada input bilangan 10. Screenshoot kode dan hasilnya, lalu tempelkan pada jawaban.

### Pendeklarasian :



Gambar 16. Kode Perulangan While-do

**Hasil Run :**



Gambar 17. output Perulangan While-do

Kode ini menunjukkan penggunaan perulangan while-do dan pernyataan if untuk menghitung jumlah bilangan genap dalam suatu rentang. Perulangan while-do sangat berguna ketika kondisi perulangan tidak diketahui sebelumnya, dan perulangan akan terus berjalan hingga kondisi tertentu terpenuhi.