

Tugas Pendahuluan Modul 1  
STRUKTUR DATA - Ganjil 2024/2025  
"Pengenalan C++: Subprogram & Array"

Ketentuan Tugas Pendahuluan

1. Tugas Pendahuluan dikerjakan secara **Individu**.
2. TP ini bersifat **WAJIB**, tidak mengerjakan = **PENGURANGAN POIN JURNAL / TES ASESMEN**.
3. Hanya **MENGUMPULKAN** tetapi **TIDAK MENGERJAKAN** = **PENGURANGAN POIN JURNAL / TES ASESMEN**.
4. Deadline pengumpulan TP Modul 2 adalah Senin, 30 September 2024 pukul 07.30 WIB.
5. **TIDAK ADA TOLERANSI KETERLAMBATAN, TERLAMBAT ATAU TIDAK MENGUMPULKAN TP MAKA DIANGGAP TIDAK MENGERJAKAN**.
6. **DILARANG PLAGIAT (PLAGIAT = E)**.
7. Kerjakan TP dengan jelas agar dapat dimengerti.
8. Codingan diupload di Github dan upload Laporan di Lab menggunakan format **PDF** dengan ketentuan: **TP\_MOD\_[XX]\_NIM\_NAMA.pdf**

**CP (WA):**

- Andini (082243700965)
- Imelda (082135374187)

**SELAMAT MENGERJAKAN^^**

# **LAPORAN PRAKTIKUM PERTEMUAN 1 STRUKTUR DATA**



**Nama :**

Zulfa Mustafa Akhyar Iswahyudi (2311104010)

**Dosen :**

Yudha Islami Sulistya

**PROGRAM STUDI S1 REKAYASA PERANGKAT LUNAK FAKULTAS  
INFORMATIKA**

**TELKOM UNIVERSITY PURWOKERTO**

**2024**

A. Tujuan

Untuk melatih kompetensi Mahasiswa untuk memperdalam skill pemrograman C++

B. Tools

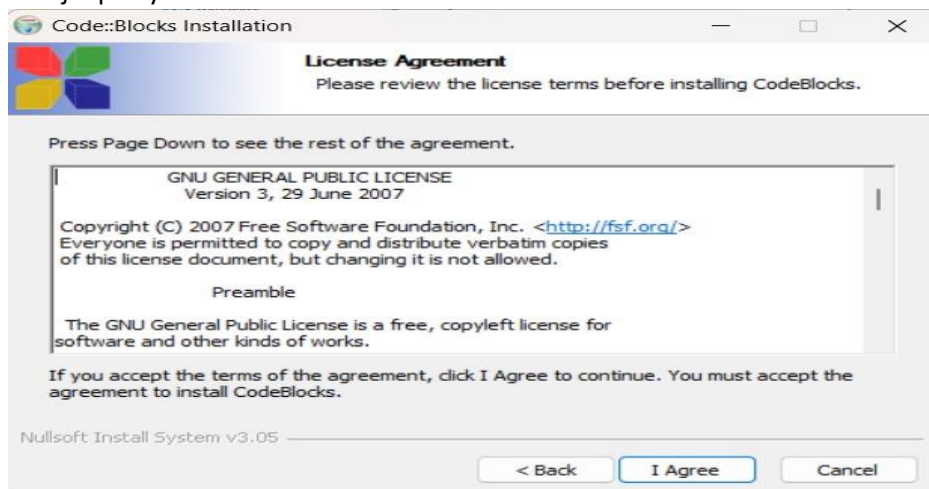
Codeblocks, VSCode, Github

## PROSES INSTALASI CODEBLOCKS/TEXT EDITOR

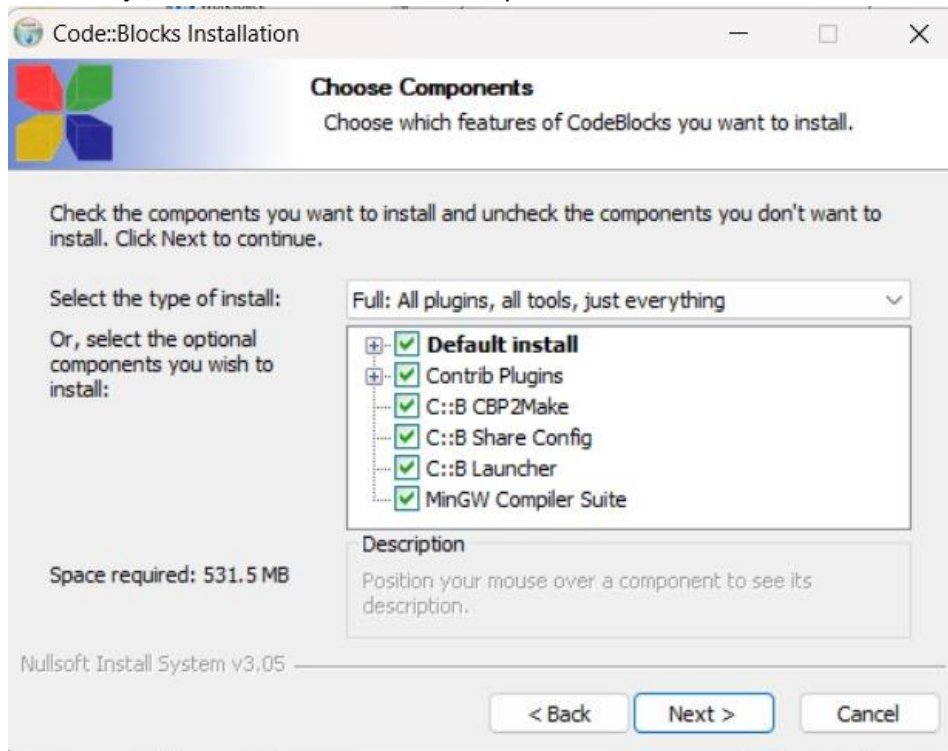
1.) Buka Installer, lalu next ->



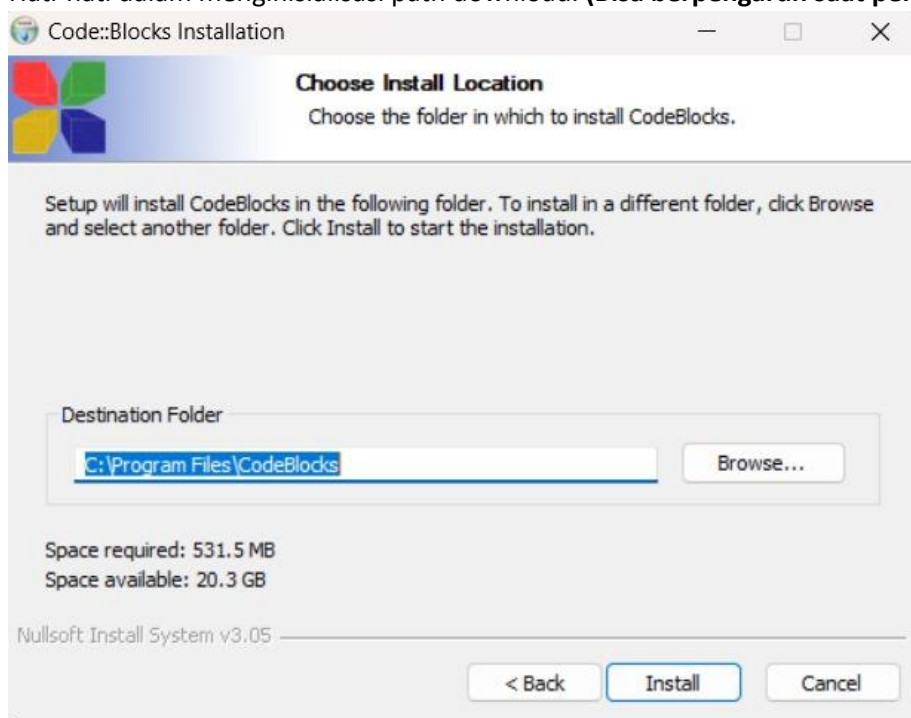
2.) Setujui prasyarat ->



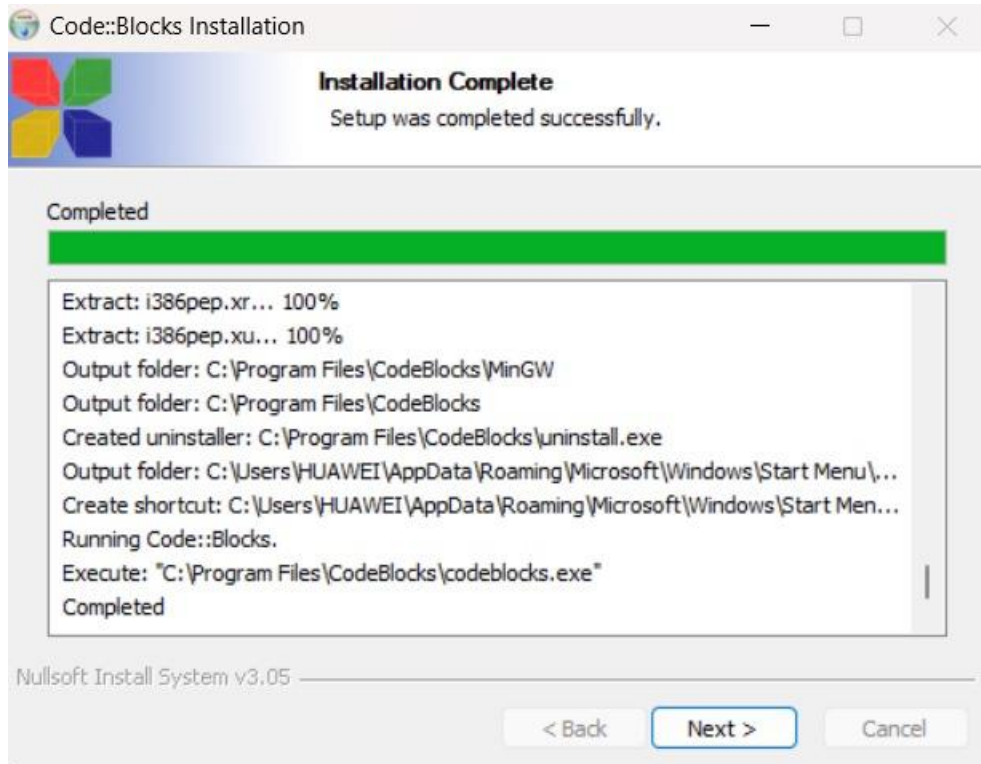
- 3.) Ini menu customization, bisa *check* dan *uncheck* tools/plugin yang diinginkan. Tapi sebaiknya ikuti default saja, checklist semua kebutuhan aplikasi.



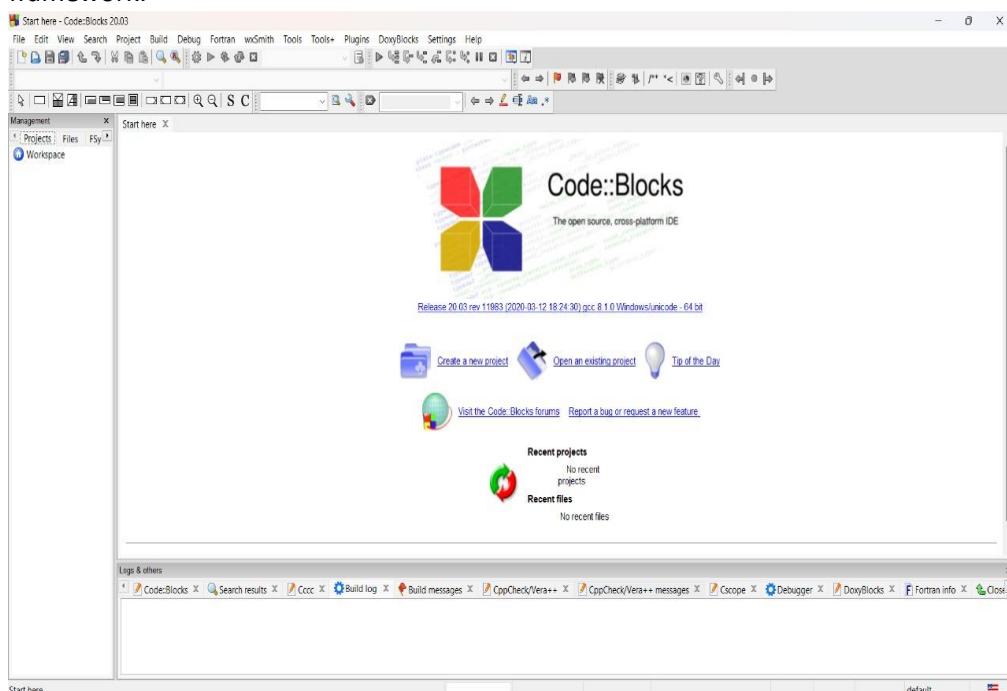
- 4.) Hati-hati dalam menginisialisasi path download. **(Bisa berpengaruh saat pemakaian)**



5.) Tunggu proses instalasi selesai, jangan diinterupsi.



6.) Instalasi selesai. Codeblocks bisa digunakan untuk pembuatan project dengan berbagai basis dan framework.



# TUGAS PENDAHULUAN - GUIDED

## Soal Tugas Pendahuluan

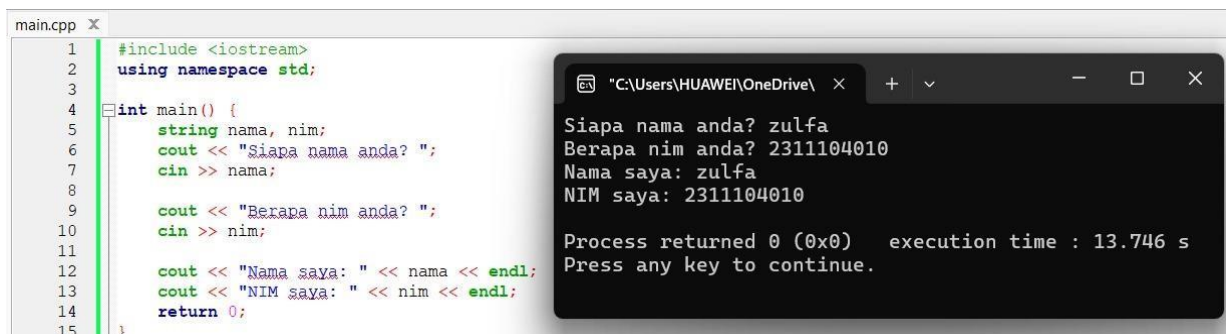
1. (Input/Output) Ini adalah contoh kodingan untuk menginput Nama dan NIM. Setelah terinput, program akan menampilkan nilai dari deklarasi 'nama' dan 'NIM'.

Input :



```
1 #include <iostream>
2
3 using namespace std;
4
5 int main(){
6     string nama, nim;
7     cout << "Siapa nama anda? ";
8     cin >> nama;
9     cout << "Berapa nim anda? ";
10    cin >> nim;
11    cout << "Nama saya:" << nama << endl;
12    cout << "NIM saya:" << nim << endl;
13    return 0;
14 }
```

Output :



```
1 #include <iostream>
2 using namespace std;
3
4 int main() {
5     string nama, nim;
6     cout << "Siapa nama anda? ";
7     cin >> nama;
8
9     cout << "Berapa nim anda? ";
10    cin >> nim;
11
12    cout << "Nama saya: " << nama << endl;
13    cout << "NIM saya: " << nim << endl;
14    return 0;
15 }
```

Process returned 0 (0x0) execution time : 13.746 s  
Press any key to continue.



2. Operasi Aritmatika. Ini adalah kodingan yang mendeklarasikan operasi hitung pada matematika yang diimplementasikan ke program C++. Pada hasil outputnya kita dapatkan enam hasil operasi dari penambahan, pengurangan, perkalian, pembagian, dan modulus.

```
main.cpp x
1  #include <iostream>
2
3  using namespace std;
4
5  int main(){
6      int bil1 = 3, bil2 = 4, hasil1;
7      float bil3 = 3.0, bil4 = 4.0, hasil2;
8      hasil1 = bil1 + bil2;
9      cout << hasil1 << endl;
10     hasil1 = bil1 - bil2;
11     cout << hasil1 << endl;
12     hasil1 = bil1 * bil2;
13     cout << hasil1 << endl;
14     hasil1 = bil1 / bil2; // integer division
15     cout << hasil1 << endl;
16     hasil1 = bil2 / bil1; // integer division
17     cout << hasil1 << endl;
18     hasil1 = bil1 % bil2; // modulo
19     cout << hasil1 << endl;
20     hasil1 = bil2 % bil1; // modulo
21     cout << hasil1 << endl;
22     hasil2 = bil3 / bil4;
23     cout << hasil2 << endl;
24     return 0;
25 }
```

Output :

```
main.cpp x
1  #include <iostream>
2  using namespace std;
3
4  int main() {
5      int bil1 = 3, bil2 = 4, hasil1;
6      float bil3 = 3.0, bil4 = 4.0, hasil2;
7
8      // Operasi penambahan
9      hasil1 = bil1 + bil2;
10     cout << hasil1 << endl;
11
12     // Operasi pengurangan
13     hasil1 = bil1 - bil2;
14     cout << hasil1 << endl;
15
16     // Operasi perkalian
17     hasil1 = bil1 * bil2;
18     cout << hasil1 << endl;
19
20     // Operasi pembagian integer
21     hasil1 = bil1 / bil2;
22     cout << hasil1 << endl; // integer division
23
24     // Operasi modulus
25     hasil1 = bil1 % bil2;
26     cout << hasil1 << endl; // modulo
27
28     // Pembagian floating-point
29     hasil2 = bil3 / bil4; // pembagian bilangan bulat, tetapi disimpan sebagai float
30     cout << hasil2 << endl;
31
32     // Pembagian floating-point asli
33     hasil2 = bil3 / bil4;
34     cout << hasil2 << endl;
35
36     return 0;
37 }
```

```
"C:\Users\HUAWEI\OneDrive\ x + - _ □ ×
-1
12
0
3
0
0.75

Process returned 0 (0x0)   execution time : 0.068 s
Press any key to continue.
```

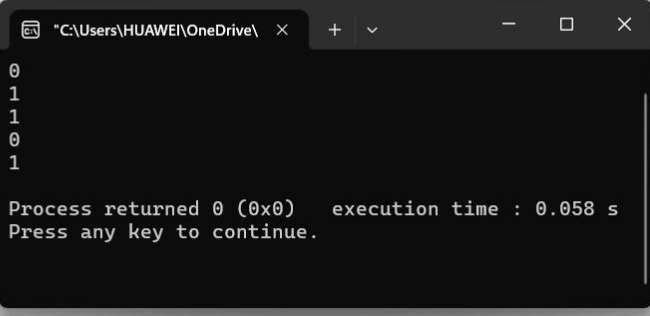


3. Operasi Perbandingan. Ini adalah kodingan yang membandingkan dua variabel dari bil1 dan bil2. Hasilnya bisa kita lihat pada gambar sebagai berikut.

```
main.cpp x
1  #include <iostream>
2
3  using namespace std;
4
5  int main(){
6      int bil1 = 2, bil2 = 3, hasil;
7      hasil = bil1 > bil2;
8      cout << hasil << endl;
9      hasil = bil1 >= bil2;
10     cout << hasil << endl;
11     hasil = bil1 < bil2;
12     cout << hasil << endl;
13     hasil = bil1 <= bil2;
14     cout << hasil << endl;
15     hasil = bil1 == bil2;
16     cout << hasil << endl;
17     hasil = bil1 != bil2;
18     cout << hasil << endl;
19     return 0;
20 }
```

Output :

```
main.cpp x
1  #include <iostream>
2  using namespace std;
3
4  int main(){
5      int bil1 = 2, bil2 = 3, hasil;
6      hasil = bil1 > bil2;
7      cout << hasil << endl;
8      hasil = bil1 >= bil2;
9      cout << hasil << endl;
10     hasil = bil1 < bil2;
11     cout << hasil << endl;
12     hasil = bil1 <= bil2;
13     cout << hasil << endl;
14     hasil = bil1 == bil2;
15     cout << hasil << endl;
16     hasil = bil1 != bil2;
17     cout << hasil << endl;
18     return 0;
19 }
```



```
0
1
1
0
1
Process returned 0 (0x0)   execution time : 0.058 s
Press any key to continue.
```

4. Berikut adalah kodingan yang mengoperasikan logika.

```
main.cpp x
1  #include <iostream>
2
3  using namespace std;
4
5  int main(){
6      int bil1 = 2, bil2 = 3, hasil;
7      hasil = bil1 <= bil2 and bil1 < bil2;
8      cout << hasil << endl;
9      hasil = bil1 >= bil2 or bil1 < bil2;
10     cout << hasil << endl;
11     hasil = not(bil1 >= bil2) or bil1 < bil2;
12     cout << hasil << endl;
13     return 0;
14 }
```

Output :

```
main.cpp x
1  #include <iostream>
2
3  using namespace std;
4
5  int main() {
6      int bil1 = 2, bil2 = 3, hasil;
7      hasil = bil1 <= bil2 and bil1 < bil2;
8      cout << hasil << endl;
9      hasil = bil1 >= bil2 or bil1 < bil2;
10     cout << hasil << endl;
11     hasil = not(bil1 >= bil2) or bil1 < bil2;
12     cout << hasil << endl;
13     return 0;
14 }
```

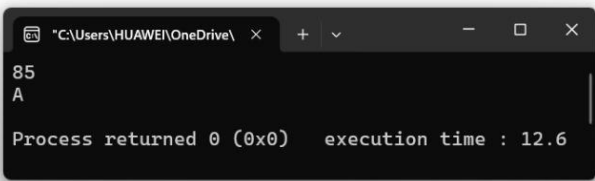


5. Ini adalah kodingan berbentuk pengkondisian. Kita atur inputan nilai berupa angka, jika nilainya lebih besar dari 80, maka indeks nilainya adalah A.

```
main.cpp x
1  #include <iostream>
2
3  using namespace std;
4
5  int main(){
6      int nilai;
7      cin >> nilai;
8      if (nilai > 80) {
9          cout << "A" << endl;
10     } else {
11         cout << "Bukan A" << endl;
12     }
13     return 0;
14 }
15
```

Output :

```
main.cpp x
1  #include <iostream>
2
3  using namespace std;
4
5  int main(){
6      int nilai;
7      cin >> nilai;
8      if (nilai > 80) {
9          cout << "A" << endl;
10     } else {
11         cout << "Bukan A" << endl;
12     }
13     return 0;
14 }
```

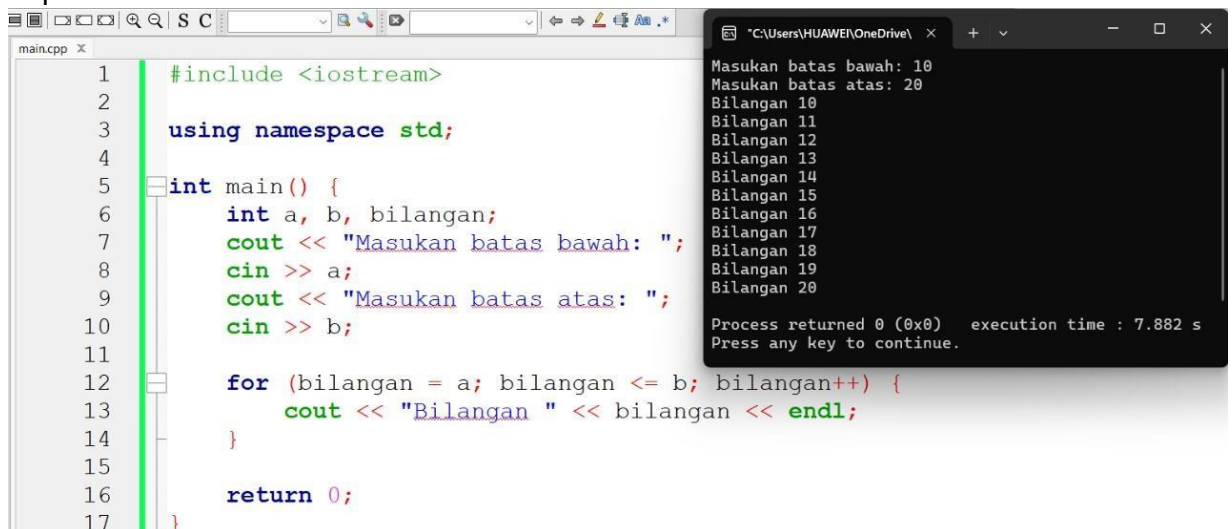


```
C:\Users\HUAWEI\OneDrive\ x + - □ ×
85
A
Process returned 0 (0x0)   execution time : 12.6
```

6. Ini disebut perulangan for-to-do. Program ini akan menginisialisasikan perulangan berdasarkan angka batas atas dan batas bawah. Buat inputan untuk batas-batas tersebut contoh 10(bawah) dan 20(atas). Maka perulangan yang terbentuk adalah angka-angka diantara 10 dan 20.

```
main.cpp x
1  #include <iostream>
2
3  using namespace std;
4
5  int main(){
6      int a, b, bilangan;
7      cout << "Masukan batas bawah: ";
8      cin >> a;
9      cout << "Masukan batas atas: ";
10     cin >> b;
11     for (bilangan = a; bilangan <= b; bilangan++) {
12         cout << "Bilangan " << bilangan << endl;
13     }
14     return 0;
15 }
```

Output :



The screenshot shows a code editor with the same C++ code as above. To the right, a terminal window displays the program's output. The user has entered 10 for the lower limit and 20 for the upper limit. The program then prints numbers from 10 to 20, each on a new line. At the bottom of the terminal, it shows 'Process returned 0 (0x0) execution time : 7.882 s' and 'Press any key to continue.'

```
main.cpp x
1  #include <iostream>
2
3  using namespace std;
4
5  int main() {
6      int a, b, bilangan;
7      cout << "Masukan batas bawah: ";
8      cin >> a;
9      cout << "Masukan batas atas: ";
10     cin >> b;
11
12     for (bilangan = a; bilangan <= b; bilangan++) {
13         cout << "Bilangan " << bilangan << endl;
14     }
15
16     return 0;
17 }
```

```
"C:\Users\HUAWEI\OneDrive\ x
Masukan batas bawah: 10
Masukan batas atas: 20
Bilangan 10
Bilangan 11
Bilangan 12
Bilangan 13
Bilangan 14
Bilangan 15
Bilangan 16
Bilangan 17
Bilangan 18
Bilangan 19
Bilangan 20

Process returned 0 (0x0)   execution time : 7.882 s
Press any key to continue.
```

7. Ini adalah perulangan while-do. Artinya ini adalah program yang melakukan perulangan yang dimana akan terus mengeksekusi kondisi selama kondisinya benar.

```
main.cpp X
1  #include <iostream>
2
3  using namespace std;
4
5  int main(){
6      int bilangan, asli, jumlah;
7
8      cout << "Masukkan bilangan asli: ";
9      cin >> asli;
10
11     bilangan = 1;
12     jumlah = 0;
13     while (bilangan <= asli) {
14         if (bilangan % 2 == 0) {
15             jumlah += bilangan;
16         }
17         bilangan++;
18     }
19     cout << "Jumlah bilangan genap: " << jumlah << endl;
20     return 0;
21 }
```

Output :

```
main.cpp X
1  #include <iostream>
2
3  using namespace std;
4
5  int main() {
6      int bilangan, asli, jumlah;
7
8      cout << "Masukkan bilangan asli: ";
9      cin >> asli;
10
11     bilangan = 1;
12     jumlah = 0;
13
14     while (bilangan <= asli) {
15         if (bilangan % 2 == 0) {
16             jumlah += bilangan;
17         }
18         bilangan++;
19     }
20
21     cout << "Jumlah bilangan genap: " << jumlah << endl;
22     return 0;
23 }
```

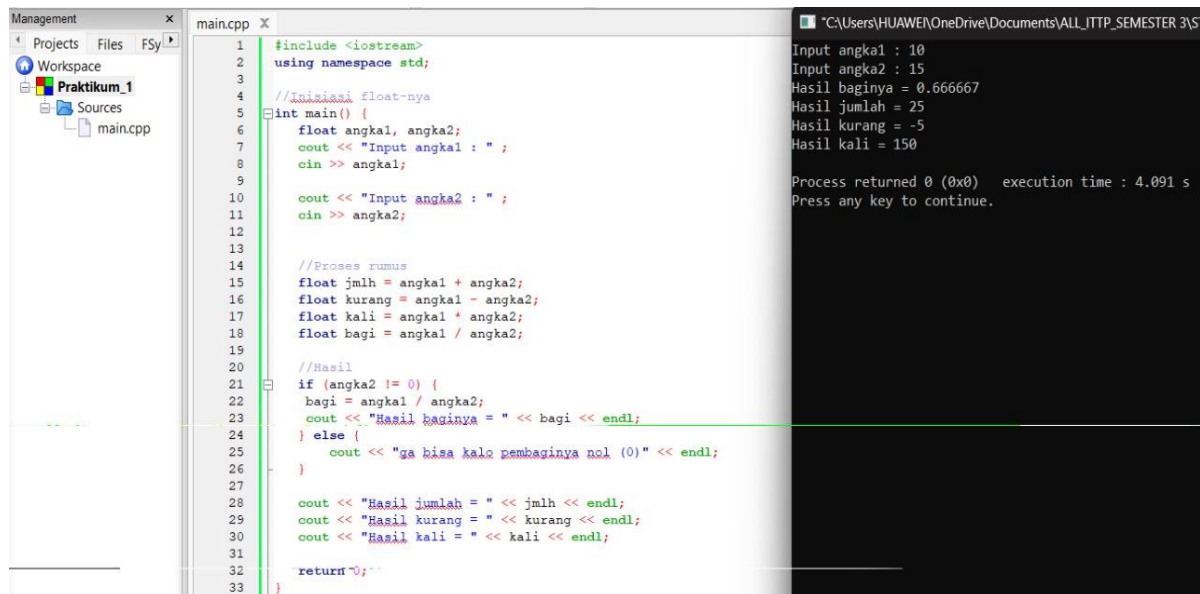


```
"C:\Users\HUAWEI\OneDrive\ X + - □ X
Masukkan bilangan asli: 10
Jumlah bilangan genap: 30
Process returned 0 (0x0) execution time : 1.549 s
Press any key to continue.
```

# TUGAS PENDAHULUAN - UNGUIDED

## A.) UNGUIDED

1.) Operasi Float Aritmatika. Kodingan ini menjalankan proses inputan angka bertipe data Float. Disini kita deklarasikan operasi aritmatika seperti penambahan, pengurangan, perkalian, dan pembagian. Kita buat rumusan operasi tersebut didalamnya dan buat pengkondisian tertera pada gambar jika inputannya adalah 0 (nol)



The screenshot shows a C++ IDE with a project named 'Praktikum\_1'. The source file 'main.cpp' contains the following code:

```
1 #include <iostream>
2 using namespace std;
3
4 //Inisiasi float-nya
5 int main() {
6     float angka1, angka2;
7     cout << "Input angka1 : " ;
8     cin >> angka1;
9
10    cout << "Input angka2 : " ;
11    cin >> angka2;
12
13
14    //Proses rumus
15    float jmlh = angka1 + angka2;
16    float kurang = angka1 - angka2;
17    float kali = angka1 * angka2;
18    float bagi = angka1 / angka2;
19
20    //Hasil
21    if (angka2 != 0) {
22        bagi = angka1 / angka2;
23        cout << "Hasil baginya = " << bagi << endl;
24    } else {
25        cout << "ga bisa kalo pembaginya nol (0)" << endl;
26    }
27
28    cout << "Hasil jumlah = " << jmlh << endl;
29    cout << "Hasil kurang = " << kurang << endl;
30    cout << "Hasil kali = " << kali << endl;
31
32    return 0;
33 }
```

The output window shows the execution results:

```
Input angka1 : 10
Input angka2 : 15
Hasil baginya = 0.666667
Hasil jumlah = 25
Hasil kurang = -5
Hasil kali = 150

Process returned 0 (0x0)   execution time : 4.091 s
Press any key to continue.
```

SC :

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

```
//Inisiasi float-nya int main() {
float angka1, angka2;    cout <<
"Input angka1 : " ;    cin >>
angka1;
```

```
    cout << "Input angka2 : " ;    cin
>> angka2;
```

```
    //Proses rumus    float jmlh =
angka1 + angka2;    float kurang =
angka1 - angka2;    float kali = angka1
* angka2;    float bagi = angka1 /
angka2;
```

```
    //Hasil    if (angka2 != 0) {
bagi = angka1 / angka2;
```

```

    cout << "Hasil baginya = " << bagi << endl;
} else {
    cout << "ga bisa kalo pembagiya nol (0)" << endl;
}

cout << "Hasil jumlah = " << jmlh << endl; cout
<< "Hasil kurang = " << kurang << endl; cout <<
"Hasil kali = " << kali << endl;

return 0;
}

```

2.) Konversi Angka ke Huruf. Kodingan ini menjalankan proses inputan sebuah bilangan apapun yang diinputkan, maka akan menjadi huruf/kata. Contoh digambar ini angka 20 menjadi dua puluh.

The screenshot shows a C++ IDE with a file named 'main.cpp'. The code implements a function 'konversi' that takes an integer 'nomor' and returns a string representing its value in Indonesian words. It uses arrays for units ('Satu' to 'Sembilan'), tens ('Sepuluh' to 'Sembilan Belas'), and hundreds ('Dua Puluh' to 'Sembilan Puluh'). The 'main' function prompts the user to input a number between 0 and 100, calls the 'konversi' function, and prints the result. The terminal window shows the program running successfully, with the input '20' resulting in the output 'Dua Puluh'.

```

1 #include <iostream>
2 #include <string>
3 using namespace std;
4
5 //Buat dulu library-nya
6 string satuan[] = {"", "Satu", "Dua", "Tiga", "Empat", "Lima", "Enam", "Tujuh", "Delapan", "Sembilan"};
7 string belasan[] = {"Sepuluh", "Sebelas", "Dua Belas", "Tiga Belas", "Empat Belas", "Lima Belas", "Enam Belas", "Tujuh Belas", "Delapan Belas", "Sembilan Belas"};
8 string puluhan[] = {"", "", "Dua Puluh", "Tiga Puluh", "Empat Puluh", "Lima Puluh", "Enam Puluh", "Tujuh Puluh", "Delapan Puluh", "Sembilan Puluh"};
9
10 //Proses Rumus-nya
11 string konversi(int nomor) {
12     if (nomor == 0)
13         return "Nol";
14     else if (nomor < 10)
15         return satuan[nomor];
16     else if (nomor < 20)
17         return belasan[nomor - 10];
18     else if (nomor < 100) {
19         int puluh = nomor / 10;
20         int sisa = nomor % 10;
21         if (sisa == 0)
22             return puluhan[puluh];
23         else
24             return puluhan[puluh] + " " + satuan[sisa];
25     } else if (nomor == 100) {
26         return "Seratus";
27     }
28     return "";
29 }
30
31 //Mau operasi-nya
32 int main() {
33     int nomor;
34     cout << "Input nomor antara 0-100: ";
35     cin >> nomor;
36
37     if (nomor < 0 || nomor > 100) {
38         cout << "Error! Tidak bisa. Bukan hasil inputan angkanya." << endl;
39     } else {
40         cout << nomor << " : " << konversi(nomor) << endl;
41     }
42
43     return 0;
44 }

```

```

C:\Users\HUAWEI\OneDrive\ x + v - □ x
Input nomor antara 0-100: 20
20 : Dua Puluh

Process returned 0 (0x0)   execution time : 2.532 s
Press any key to continue.

```

SC :

```

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

```

```

//Buat dulu library-nya
string satuan[] = {"", "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"};

```

```

//Proses Rumus-nya string
konversi(int nomor) { if (nomor

```

```

== 0)    return "Nol";  else if
(nomor < 10)    return
satuan[nomor];  else if (nomor <
20)    return belasan[nomor - 10];
else if (nomor < 100) {    int puluh
= nomor / 10;
    int sisa = nomor % 10;
    if (sisa == 0)    return
puluhan[puluh];    else
    return puluhan[puluh] + " " + satuan[sisa];
} else if (nomor == 100) {
    return "Seratus";
}    return "";
}

```

```

//Mulai Operasinya int

```

```

main() {    int nomor;
    cout << "Input nomor antara 0-100: ";    cin
>> nomor;

```

```

    if (nomor < 0 || nomor > 100) {    cout << "Eiyo, tidak bisa. Bukan
begitu inputan angkanya." << endl;
    } else {
        cout << nomor << " : " << konversi(nomor) << endl;
    }

```

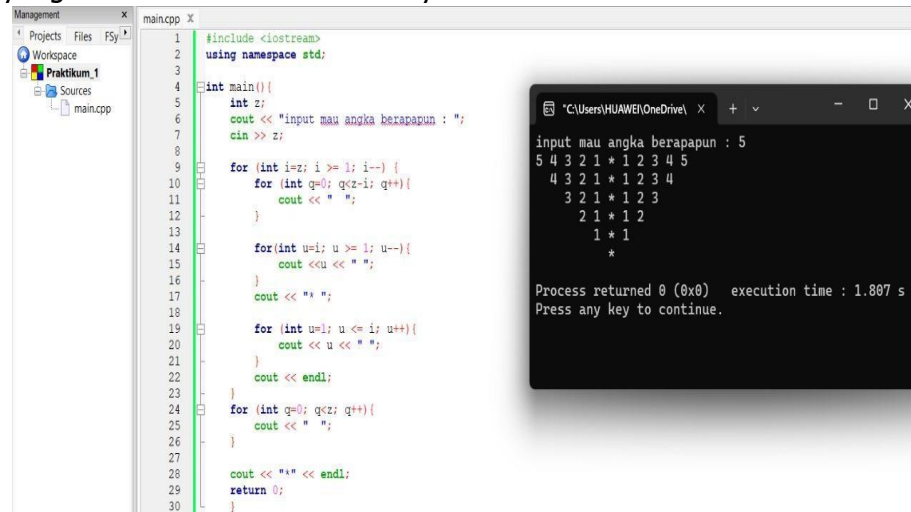
```

    return 0;
}

```



3.) Operasi Piramida. Ini adalah kodingan yang akan membentuk suatu pola menurun seperti piramida terbalik. Panjang dan banyaknya pola tergantung inputan angka yang dimasukkan. Berikut hasilnya



The screenshot shows a C++ IDE with a file named 'main.cpp'. The code defines a function 'main()' that takes an integer 'z' as input. It uses nested loops to print a reversed pyramid pattern of asterisks. The first loop prints spaces to align the pattern to the right. The second loop prints the asterisks for each row. The output in the terminal window shows the pattern for 'z = 5'.

```

1 #include <iostream>
2 using namespace std;
3
4 int main(){
5     int z;
6     cout << "input mau angka berapapun : ";
7     cin >> z;
8
9     for (int i=z; i >= 1; i--) {
10         for (int q=0; q<z-i; q++){
11             cout << " ";
12         }
13
14         for(int u=i; u >= 1; u--){
15             cout <<u << " ";
16         }
17         cout << "* ";
18
19         for (int u=1; u <= i; u++){
20             cout << u << " ";
21         }
22         cout << endl;
23     }
24     for (int q=0; q<z; q++){
25         cout << " ";
26     }
27
28     cout << "*" << endl;
29     return 0;
30 }

```

```

input mau angka berapapun : 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
*

Process returned 0 (0x0)   execution time : 1.807 s
Press any key to continue.

```

SC :

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

```
int main(){
```

```
    int z;
```

```
    cout << "input mau angka berapapun : ";
```

```
    cin >> z;
```

```
    for (int i=z; i >= 1; i--) {
```

```
        for (int q=0; q<z-i; q++){
```

```
            cout << " ";
```

```
        }
```

```
        for(int u=i; u >= 1; u--){
```

```
            cout <<u << " ";
```

```
        }
```

```
        cout << "* ";
```

```
        for (int u=1; u <= i; u++){
```

```
            cout << u << " ";
```

```
        }
```

```
        cout << endl;
```

```
    }
```

```
    for (int q=0; q<z; q++){
```

```
        cout << " ";
```

```
    }
```

```
    cout << "*" << endl;
```

```
    return 0;
```

```
}
```

# SOURCE CODE WAKTU DI KELAS – KODINGAN ASPRAK

Berikut adalah evidence dari kelas bersama Asprak :

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

```
int main() {
    cout << "OI YAROU" << endl;
    return 0;
}*/
```

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

```
int main() {
    int inp;    cin
    >> inp;
    cout << "nilai = " << inp << endl;
    return 0;
}*/
```

```
=====
```

```
/*#include <iostream>
#include <stdio.h>
using namespace std;
```

```
int main() {    float W, X, Y; float Z;
X = 7; Y =3; W=1;    Z = (X+Y) /
(Y+W);    cout << "nilai Z = " << Z
<< endl;
    return 0;
}*/
```

```
=====
```

```
/*#include <iostream>
using namespace std; int
main(){    double tot_beli,
diskon;    cout << "total
beli = Rp";    cin >>
tot_beli;    diskon = 0;
```

```
    if(tot_beli >= 100000)
diskon =0.75*tot_beli;
else    diskon = 0;
    cout << "Besar diskon = Rp" << diskon;
}*/
```

```

=====
/*#include <stdio.h>
int main(){  int
kode_hari;
puts("Tentuin hari\n");
puts("")
}*/
=====
/*#include <iostream>
#include <conio.h>

using namespace std;
int main(){
int i=1;
    int jum;
cin>>jum;

    do{
        cout << "Baris ke-" << i+1 << endl;
i++;
    } while(i<jum);
getch();
    return 0;
}*/

```

Semoga Selalu diberi kemudahan^^