

## 1. Menghitung Luas Lingkaran

### ) PSEUDOCODE

Hitung\_Luas\_Lingkaran(input, output)  
{algoritma menghitung luas lingkaran  
berdasarkan nilai jari-jari yang di  
inputkan}

Deklarasi :

```
var jari-jari : REAL (double) ;  
var luas : REAL (double);  
const pi = 3.14 : REAL (double);
```

Deskripsi :

Start

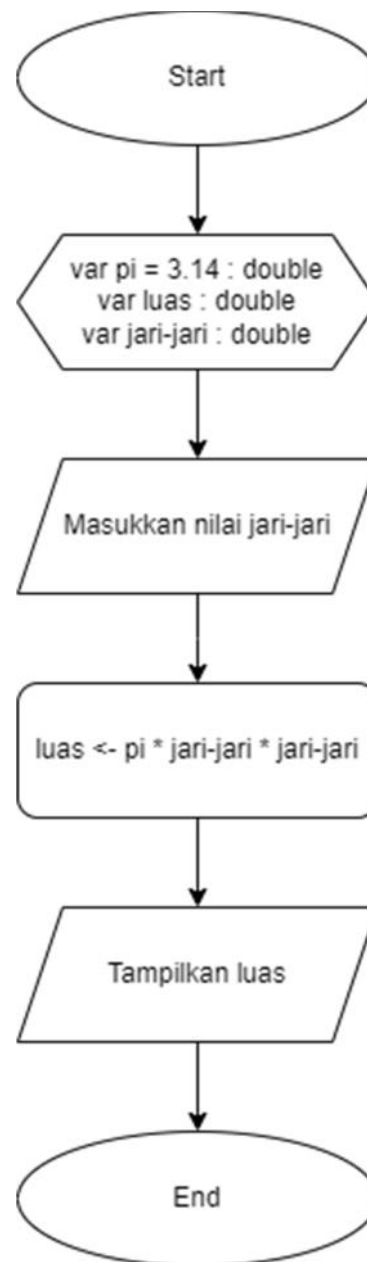
```
write("Ketikkan nilai jari-jari  
lingkaran dalam cm : ");
```

```
read(radius);
```

```
luas <- pi * jari-jari * jari-jari;  
write("Nilai Luas Lingkaran adalah :"  
+ luas + "cm persegi");
```

Stop

### ) FLOWCHART



## ) Implementasi Pada Bahasa C

/\*  
Program untuk menghitung luas lingkaran berdasarkan nilai jari-jari yang di inputkan.

Nama file : luasLingkaran.c

Programmer : M.ilham

\*/

```
#include <stdio.h>
```

```
int main(){
```

```
    const double pi = 3.14;
```

```
    double luas;
```

```
    double jariJari;
```

```
    printf("Masukkan panjang jari-jari : ");
```

```
    scanf("%lf", &jariJari);
```

```
    luas = pi * jariJari * jariJari;
```

```
    printf("Luas lingkaran : %lf\n", luas);
```

```
    getch();
```

```
    return 0;
```

```
}
```

## 2. Menghitung Volume Tabung

### ) PSEUDOCODE

Hitung\_Volume\_Tabung(input,  
output)  
{algoritma menghitung volume  
tabung berdasarkan nilai jari-jari, dan  
tinggi yang di inputkan}

Deklarasi :

```
var jari-jari, tinggi , volume : REAL  
(double);  
const pi = 3.14 : REAL (double);
```

Deskripsi :

Start

```
write("Masukkan nilai jari-jari alas  
tabung dalam cm : ");
```

```
read(jari-jari);
```

```
write("Masukkan nilai tinggi tabung  
dalam cm : ");
```

```
read(tinggi);
```

```
volume <- pi * jari-jari * jari-jari *  
tinggi;  
write("Volume tabung adalah : " +  
volume + "cm kubik");
```

Stop

### ) FLOWCHART



## ) Implementasi Pada Bahasa C

/\*  
program menghitung volume tabung berdasarkan nilai jari-jari, dan tinggi yang di inputkan.

Nama file : volumeTabung.c

Programmer : M.ilham

\*/

#include <stdio.h>

int main(){

    const double pi = 3.14;  
    double jariJari, tinggi, volume;

    printf("Masukkan panjang jari-jari : ");  
    scanf("%lf", &jariJari);

    printf("Masukkan tinggi tabung : ");  
    scanf("%lf", &tinggi);

    volume = pi \* jariJari \* jariJari \* tinggi;

    printf("%lf\n", jariJari);  
    printf("%lf\n", tinggi);

    printf("Volume tabung adalah : %lf\n", volume);  
    getch();  
    return 0;

}

### 3. Menghitung Tinggi Pohon Berdasarkan Sudut Elevasi

#### ) PSEUDOCODE

Hitung\_Tinggi\_Pohon(input, output)  
{algoritma untuk menghitung tinggi pohon berdasarkan nilai jarak, tinggi pengamat, dan sudut elevasi}

Deklarasi :

```
var jarak : REAL(double);  
var tinggi-pengamat : REAL(double);  
var sudut-elevasi : integer  
var tinggi-pohon : REAL(double)
```

Deskripsi :

Start

```
write("Masukkan jarak dalam meter  
: ");
```

```
read(jarak);
```

```
write("Masukkan tinggi pengamat  
dalam meter : ");
```

```
read(tinggi-pengamat);
```

```
write("Masukkan sudut elevasi  
dalam meter : ");
```

```
read(sudut-elevasi);
```

```
tinggi-pohon <- tan(sudut-elevasi) *  
jarak + tinggi-pengamat;
```

```
write("Tinggi pohon adalah : " +  
tinggi-pohon + meter);
```

Stop

#### ) FLOWCHART



## ) Implementasi Pada Bahasa C

```
/*  
program untuk menghitung tinggi pohon berdasarkan nilai jarak, tinggi pengamat, dan sudut  
elevasi.
```

Nama file : tinggiPohon.c

Programmer : M.ilham

```
*/
```

```
#include <stdio.h>
```

```
#include <math.h>
```

```
int main(){
```

```
    int sudutElevasi;
```

```
    double tinggiPengamat, jarak, tinggiPohon;
```

```
    printf("Masukkan jarak dalam meter : ");
```

```
    scanf("%lf", &jarak);
```

```
    printf("Masukkan tinggi pengamat dalam meter: ");
```

```
    scanf("%lf", &tinggiPengamat);
```

```
    printf("Masukan sudut elevasi dalam derajat : ");
```

```
    scanf("%d", &sudutElevasi);
```

```
    tinggiPohon = tanh(sudutElevasi) * jarak + tinggiPengamat; // err
```

```
    printf("Tinggi pohon adalah : %lf meter", tinggiPohon);
```

```
    getch();
```

```
    return 0;
```

```
}
```

#### 4. Menghitung Kebutuhan Kalori Manusia

(Laki-laki dan Perempuan) untuk seluruh (min 3) kategori aktivitas fisik

##### ) PSEUDOCODE

Hitung\_Kebutuhan\_Kalori(input,  
output)  
{algoritma untuk menghitung  
kebutuhan kalori manusia}

Deklarasi :

var kebutuhan-kalori, tinggi-badan,  
berat-badan : REAL(double);

var usia, aktivitas fisik , jenis-  
kelamin: integer;

Deskripsi :

Start

write("Masukkan tinggi-badan : ");

read(tinggi-badan);

write("Masukkan berat-badan : ");

read(berat-badan);

write("Masukkan jenis kelamin (1.  
Laki-laki , 2. perempuan) : ");

read(jenis-kelamin);

if(jenis-kelamin==1) then:

kebutuhan-kalori <- 66 +  
(13.7 \* berat-badan) + (5 \*  
tinggi-badan) – (6.8 \* usia);

else

kebutuhan-kalori <- 655 +  
(9.6 \* berat-badan) + (1.8 \*  
tinggi-badan) – (4.7 \* usia);

endif

write("Masukkan tingkat aktivitas  
fisik (1. ringan, 2.sedang, 3. berat)"  
);

read(aktivitas-fisik);

if(aktivitas-fisik==1) then

kebutuhan kalori \*= 1.2;

else if(aktivitas-fisik==2) then

kebutuhan-kalori \*=1.55;

else

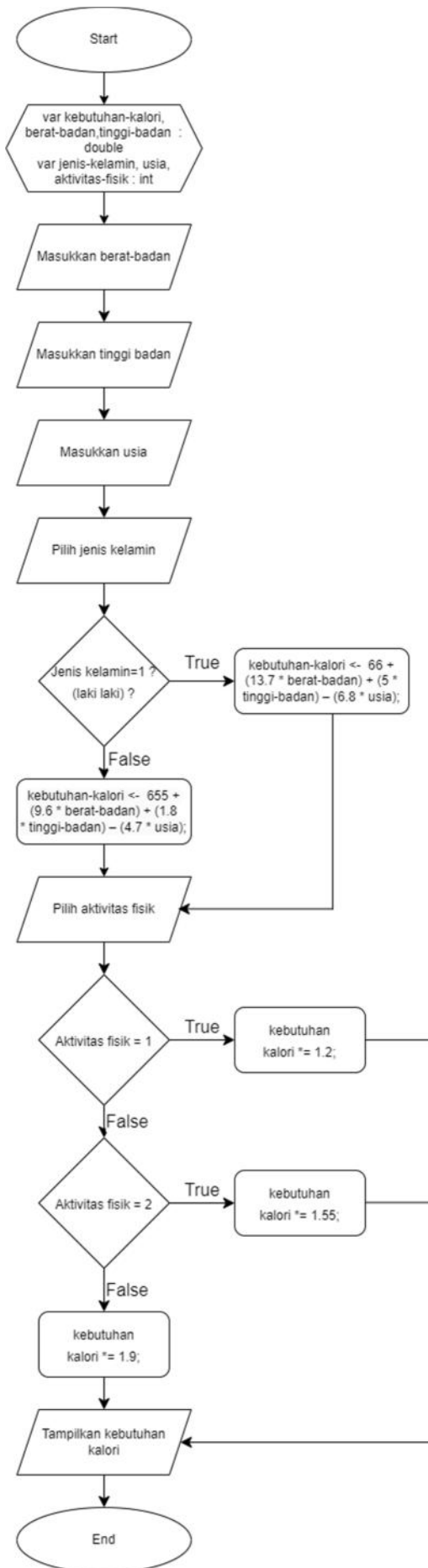
kebutuhan-kalori \*= 1.9;

endif

write("Kebutuhan kalori adalah " +  
kebutuhan-kalori);

Stop

## FLOWCHART





## ) Implementasi pada Bahasa C

```
/*
    algoritma untuk menghitung kebutuhan kalori manusia.
    Nama file : hitungKalori.c
    Programmer : M.ilham
*/

#include <stdio.h>

int main(){

    double kebutuhanKalori, tinggiBadan, beratBadan;
    int usia, jenisKelamin, aktivitasFisik;

    printf("Masukkan tinggi badan (dalam meter) : ");
    scanf("%lf", &tinggiBadan);
    printf("Masukkan berat badan (dalam kilogram) : ");
    scanf("%lf", &beratBadan);

    printf("Masukkan umur : ");
    scanf("%d", &usia);
    printf("Masukkan jenis kelamin (1. laki-laki, 2. perempuan) : ");
    fflush(stdin);
    scanf("%d", &jenisKelamin);

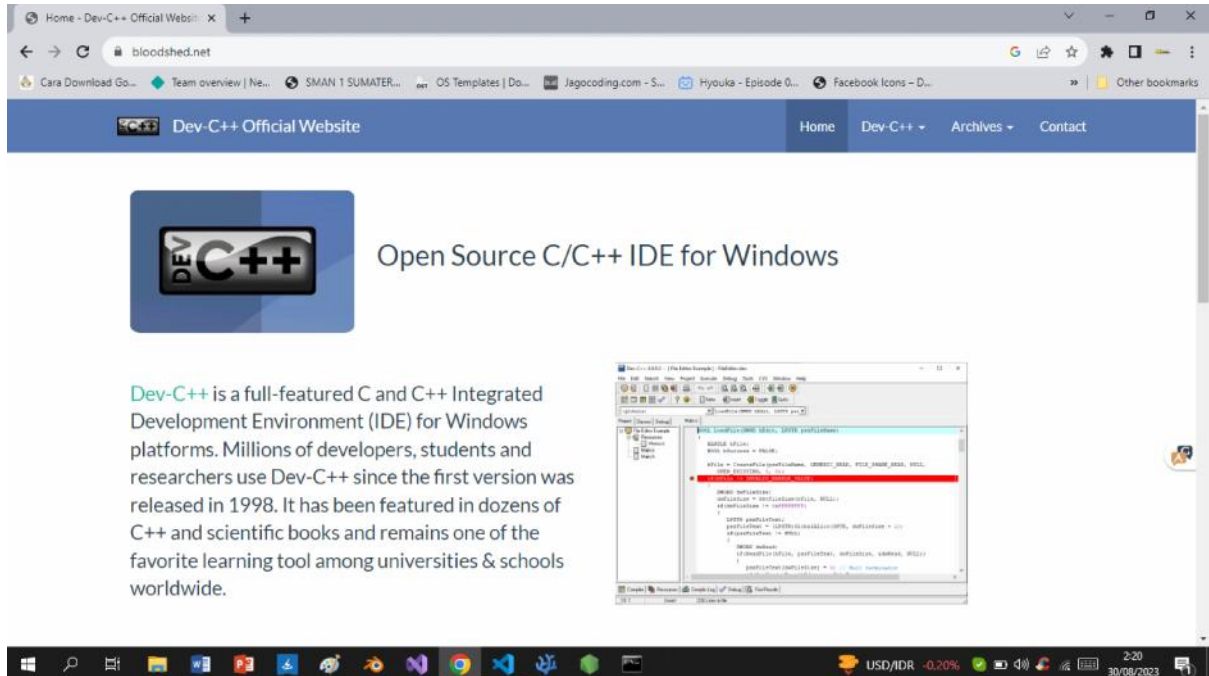
    if(jenisKelamin==1){
        kebutuhanKalori = 66 + (13.7 * beratBadan) + (5 * tinggiBadan) - (6.8 * usia);
    }
    else {
        kebutuhanKalori = 655 + (9.6 * beratBadan) + (1.8 * tinggiBadan) - (4.7 * usia);
    }
    printf("Masukkan tingkat aktivitas fisik (1. ringan, 2. sedang, 3. berat) : ");
    fflush(stdin);
    scanf("%d", &aktivitasFisik);

    if(aktivitasFisik==1){
        kebutuhanKalori *= 1.2;
    }
    else if(aktivitasFisik==2){
        kebutuhanKalori *= 1.55;
    }
    else {
        kebutuhanKalori *= 1.9;
    }
    printf("Kebutuhan Kalori : %lf", kebutuhanKalori);
    return 0;
}
```

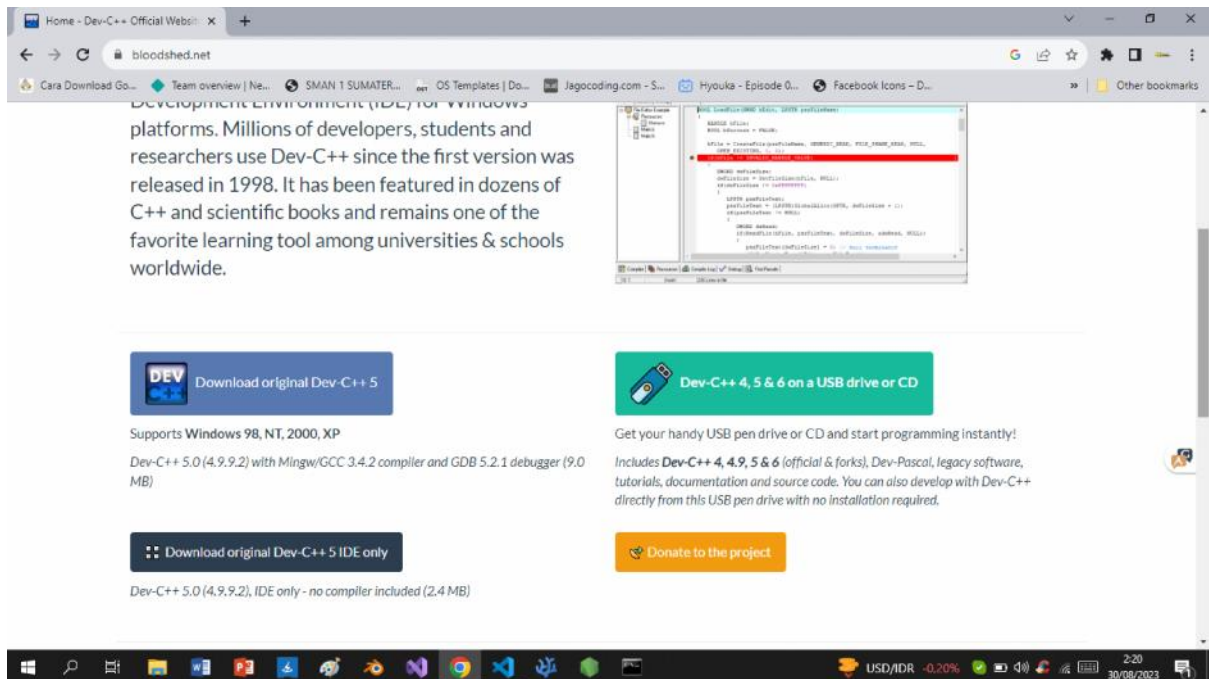
## 5. Cara mendownload, menginstall, dan contoh penggunaan Dev C++

### 1. Mendownload

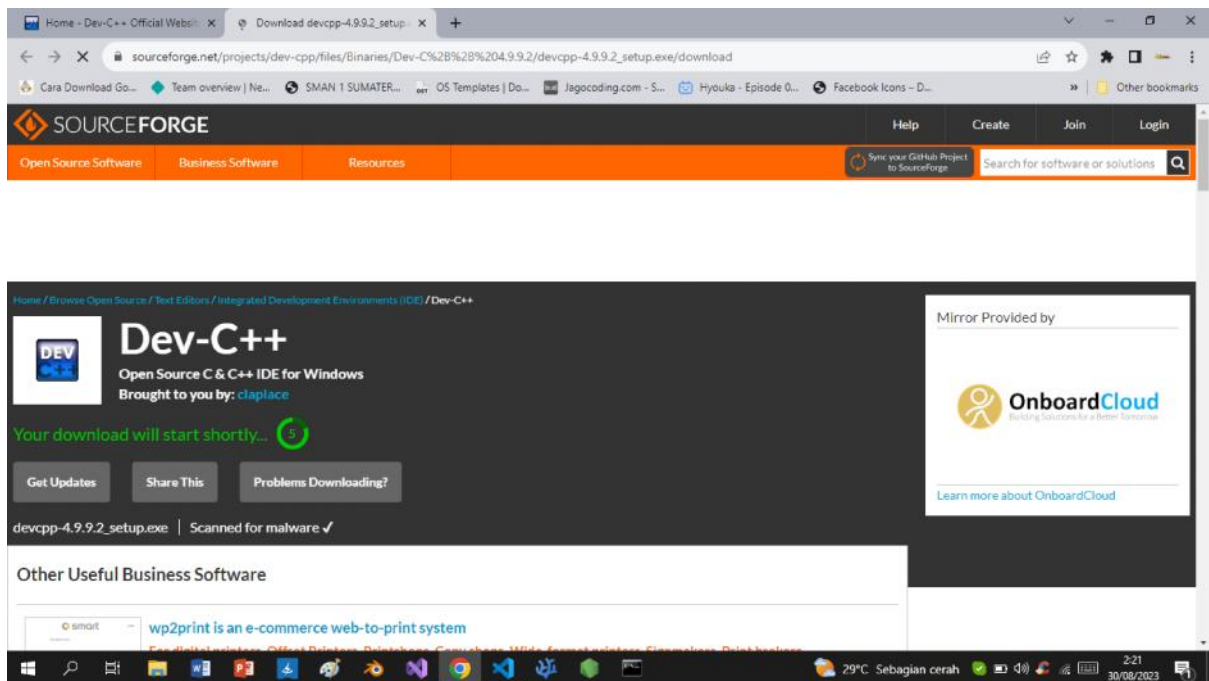
#### 1. Buka website installer Dev C



#### 2. Lalu scroll kebawah dan klik tombol download original Dev C++ 5

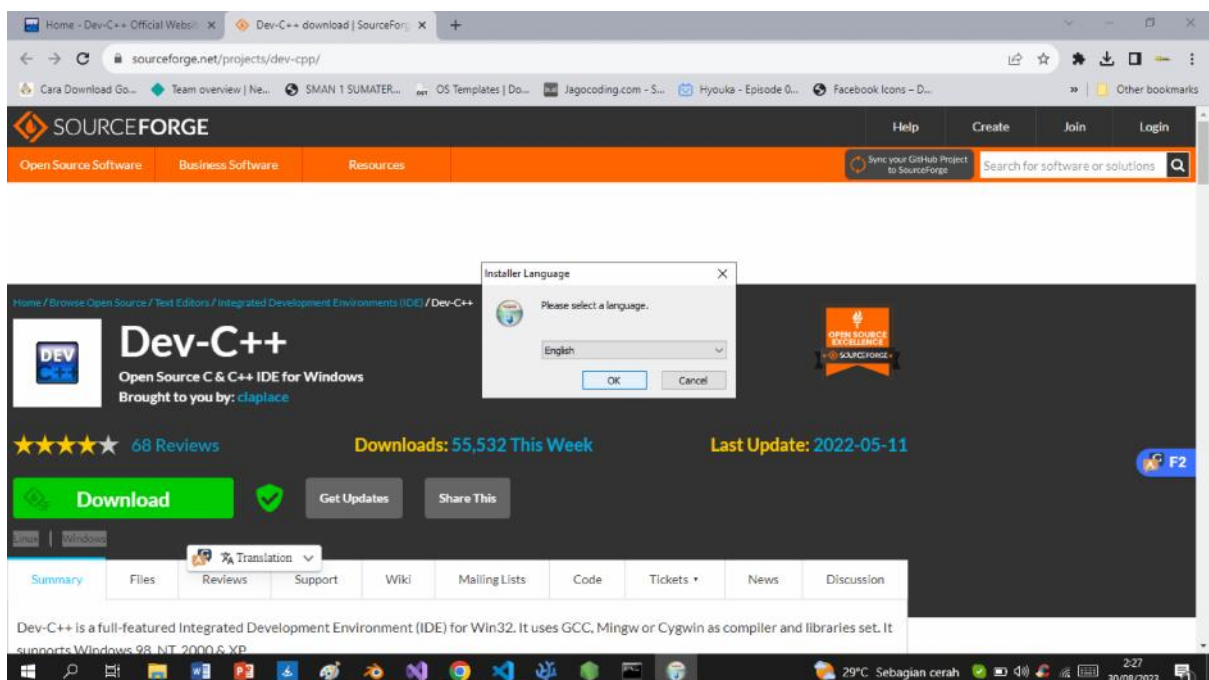


- Secara otomatis nanti akan diarahkan ke source forge dan file installer akan terdownload otomatis dalam 5 detik

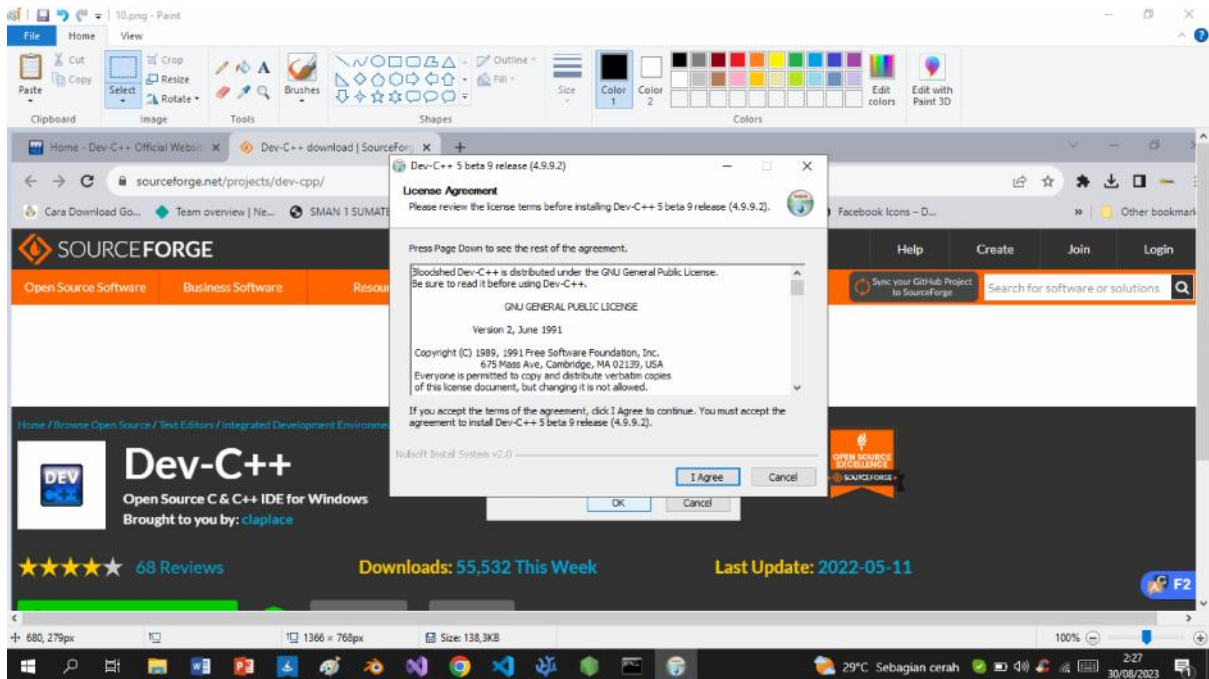


## Menyinstal

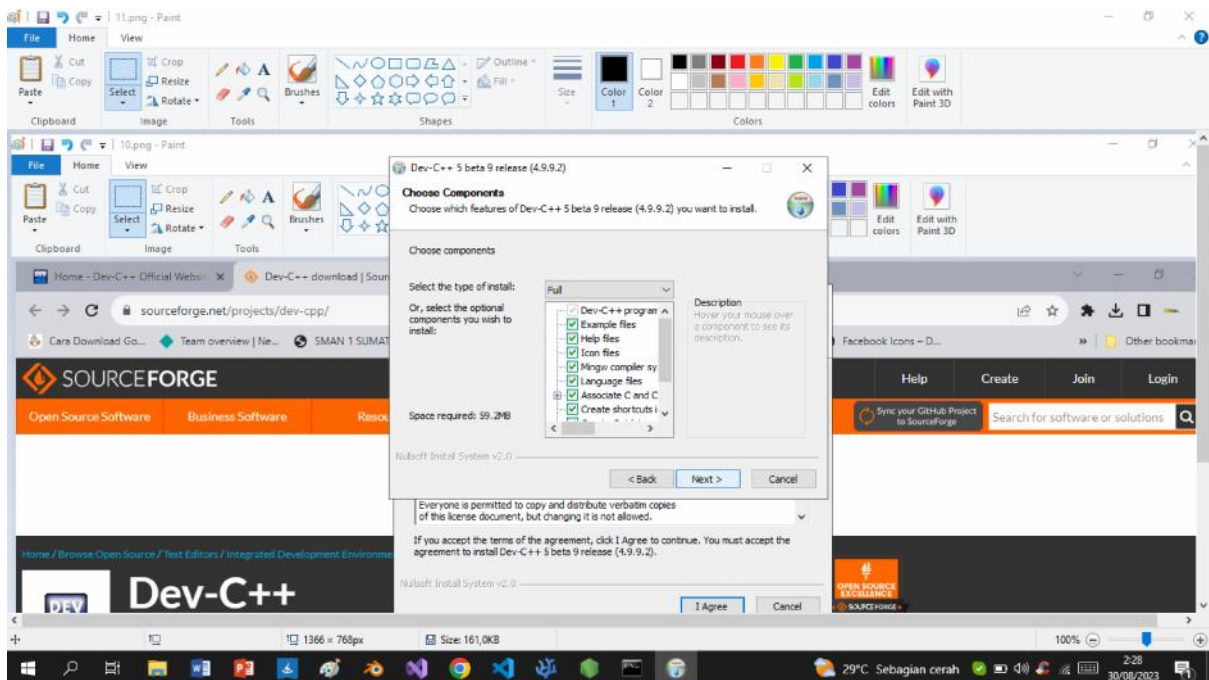
- Buka installer yang terdownload pada langkah pertama, lalu pilih bahasa yang ingin digunakan



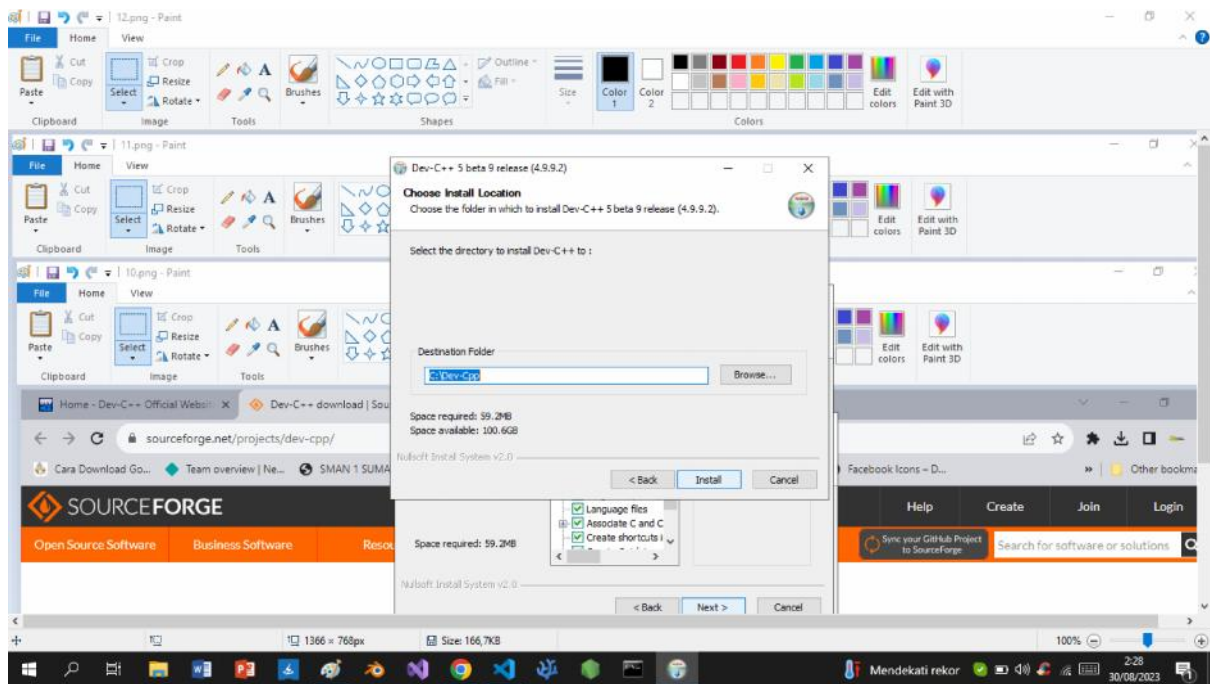
- Pada license agreement klik I Agree



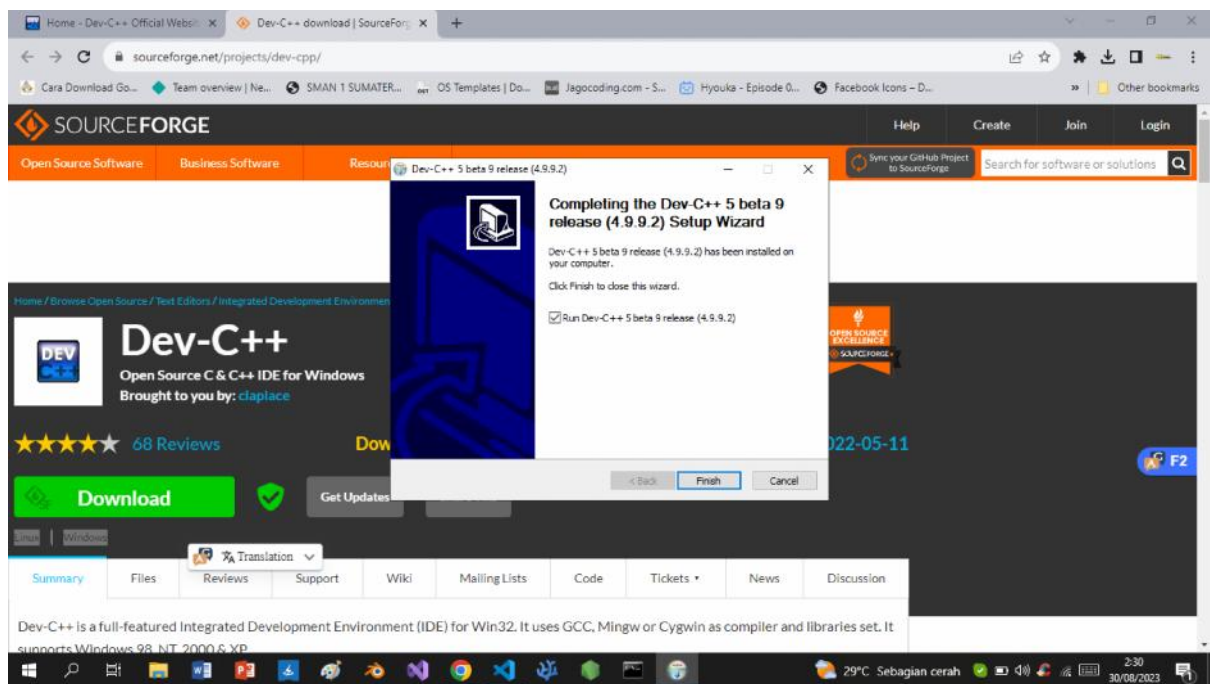
3. Biarkan pada settingan default lalu klik next



4. Pilih lokasi penyimpanan aplikasi, lalu klik install dan tunggu hingga proses instalasi selesai

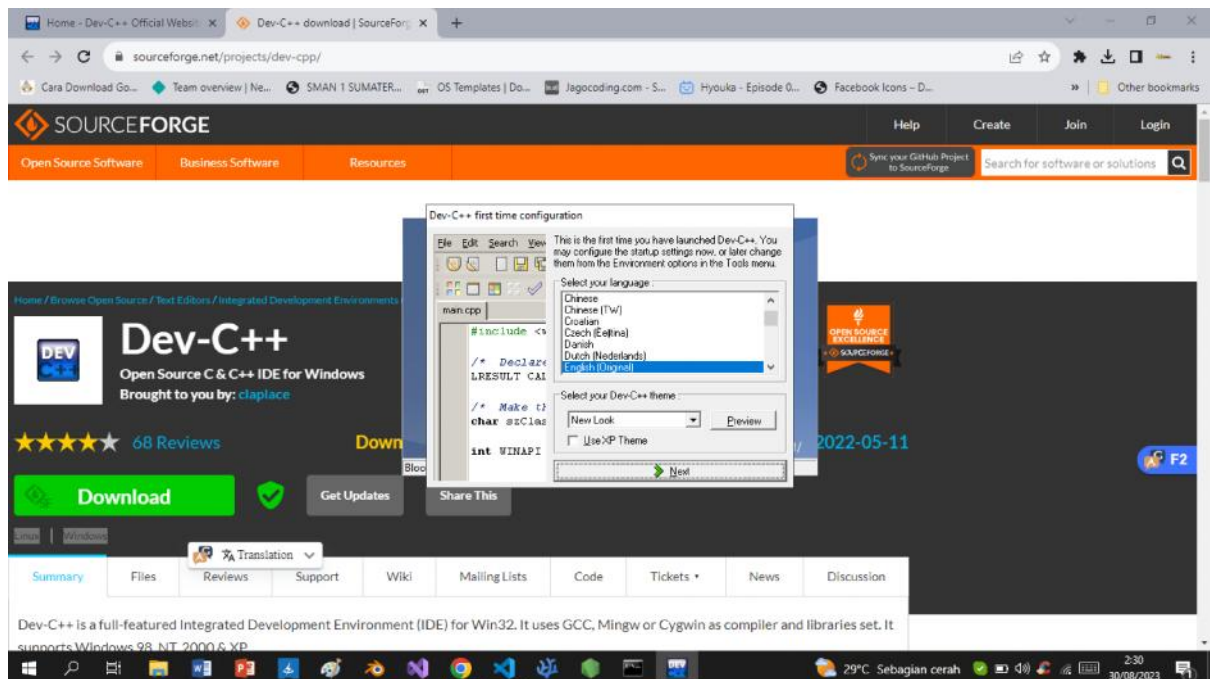


## 5. Setelah selesai klik finish

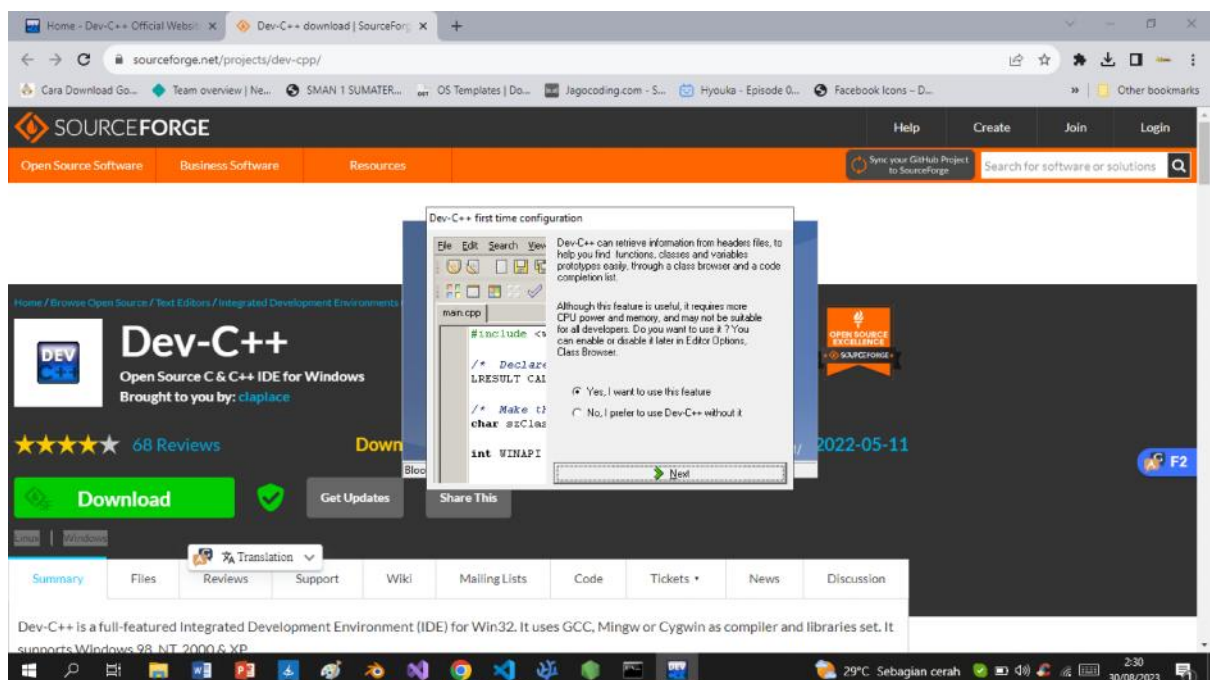




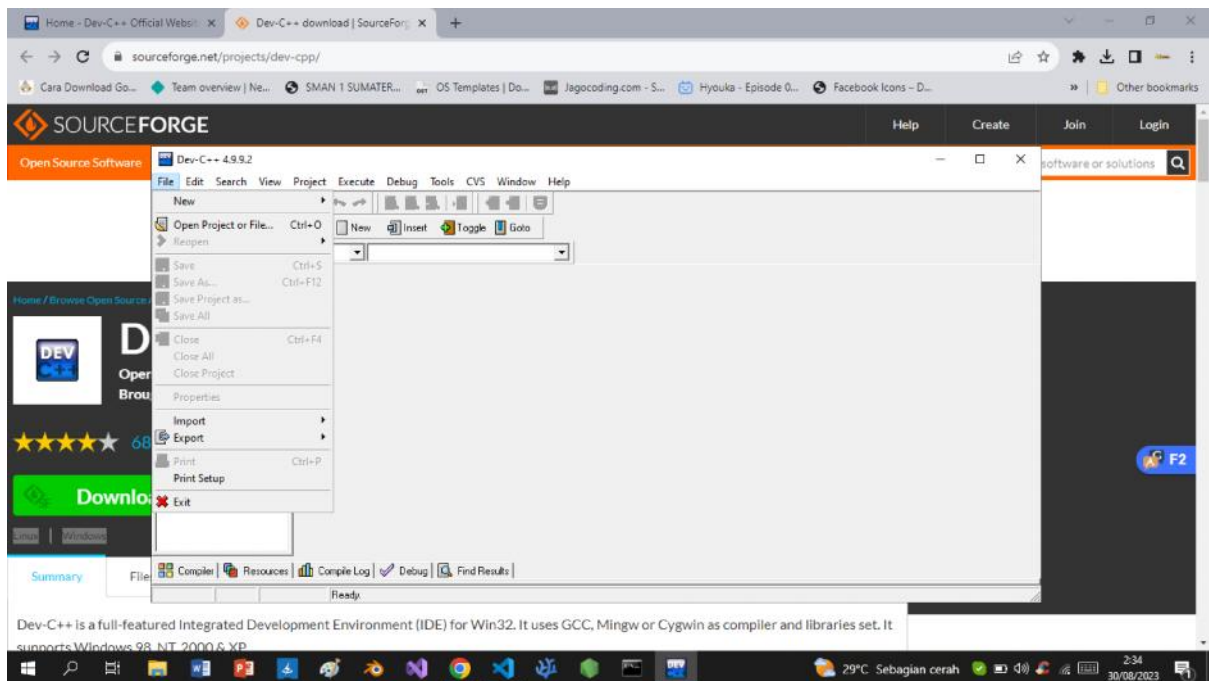
6. Untuk mengkonfigurasi aplikasi, jalankan aplikasi untuk pertama kali, lalu pilih Bahasa yang ingin di gunakan lalu klik next



7. Next saja terus hingga loading konfigurasi selesai



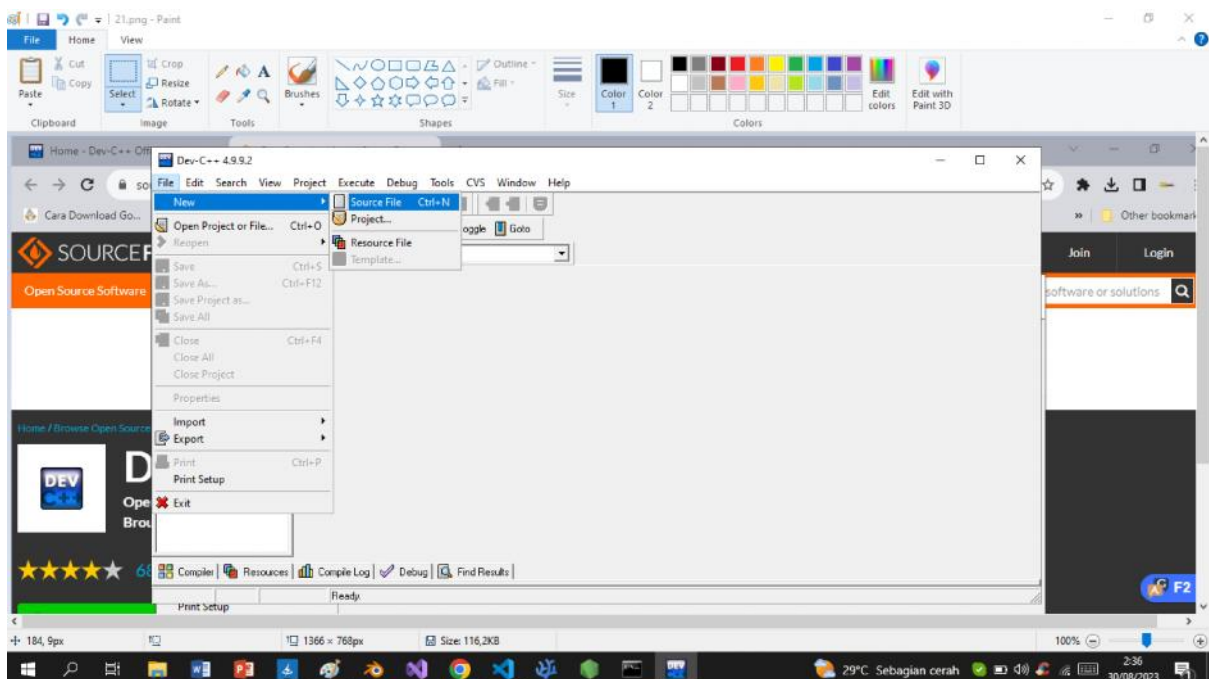
## 8. Aplikasi siap digunakan



## Contoh Penggunaan

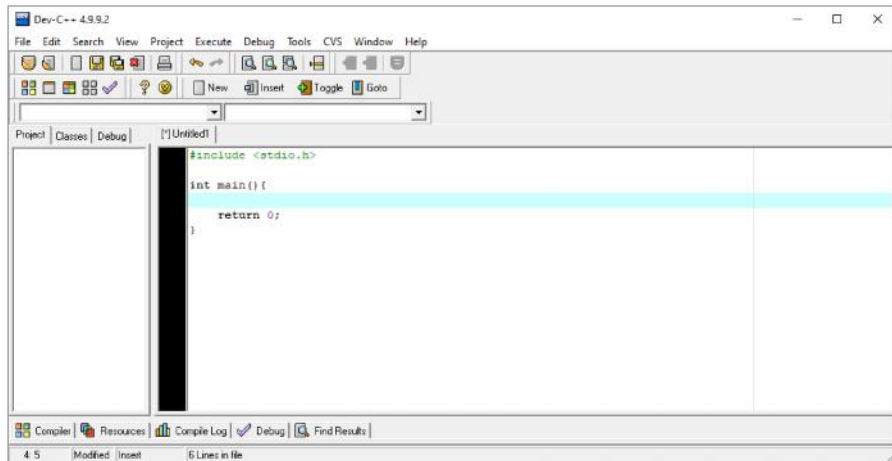
Pada contoh penggunaan, saya akan menerapkan penggunaan Bahasa C untuk menampilkan Hello World ke layer

### 1. Klik menu file -> new -> source file



## 2. Lalu masukkan struktur dasar Bahasa C

Blender



The screenshot shows the Dev-C++ 4.9.9.2 IDE. The main window displays the following C code:

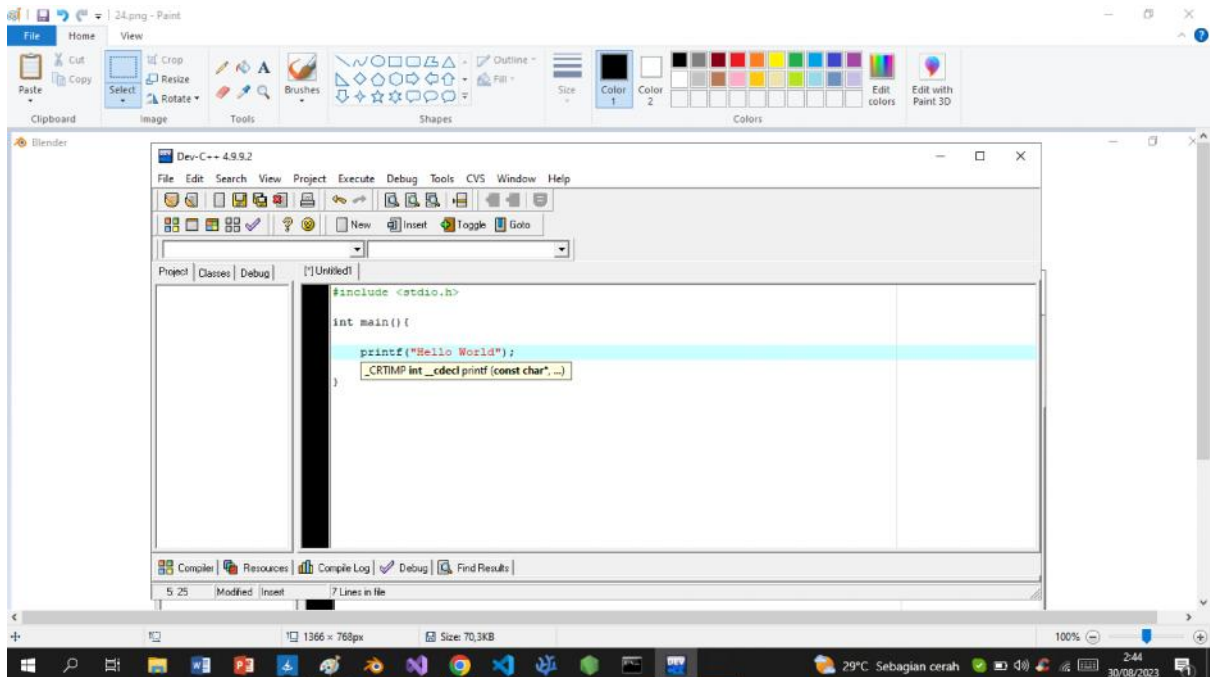
```
#include <stdio.h>

int main() {

    return 0;
}
```

The status bar at the bottom indicates "4 5 Modified Insert 6 Lines in file".

## 3. Lalu masukkan syntax untuk menampilkan teks



The screenshot shows the Dev-C++ 4.9.9.2 IDE with the following C code:

```
#include <stdio.h>

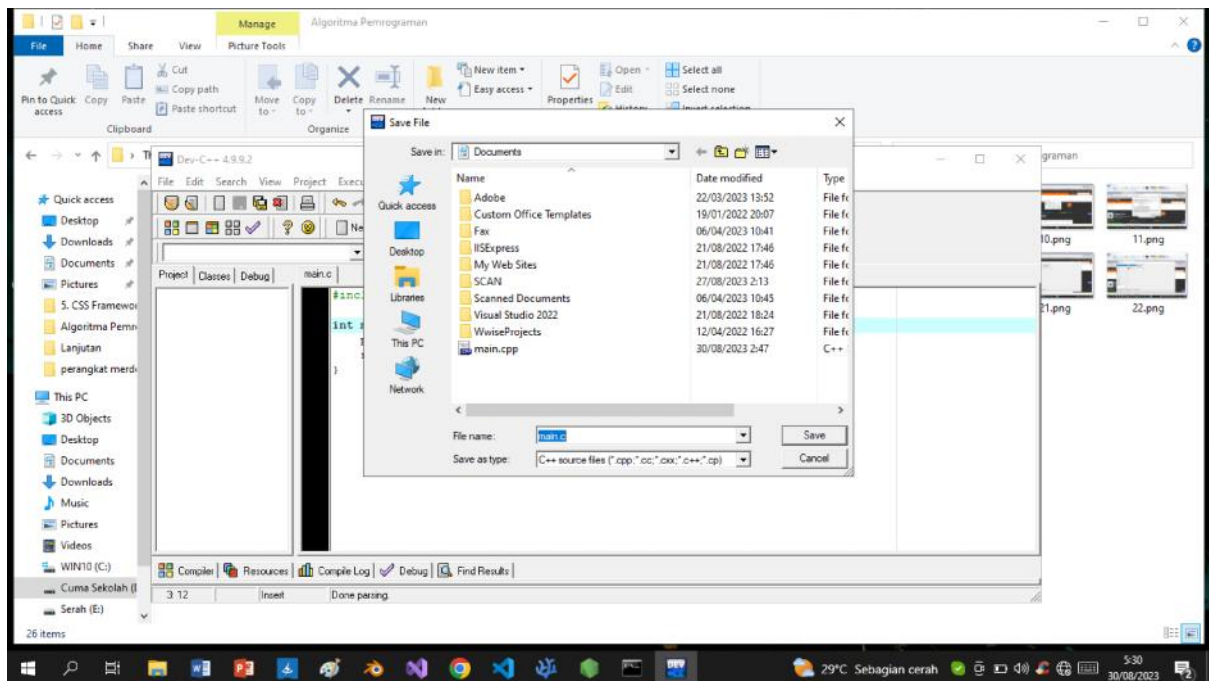
int main() {

    printf("Hello World");
    _CRTIMP int __cdecl printf(const char*, ...)
}
```

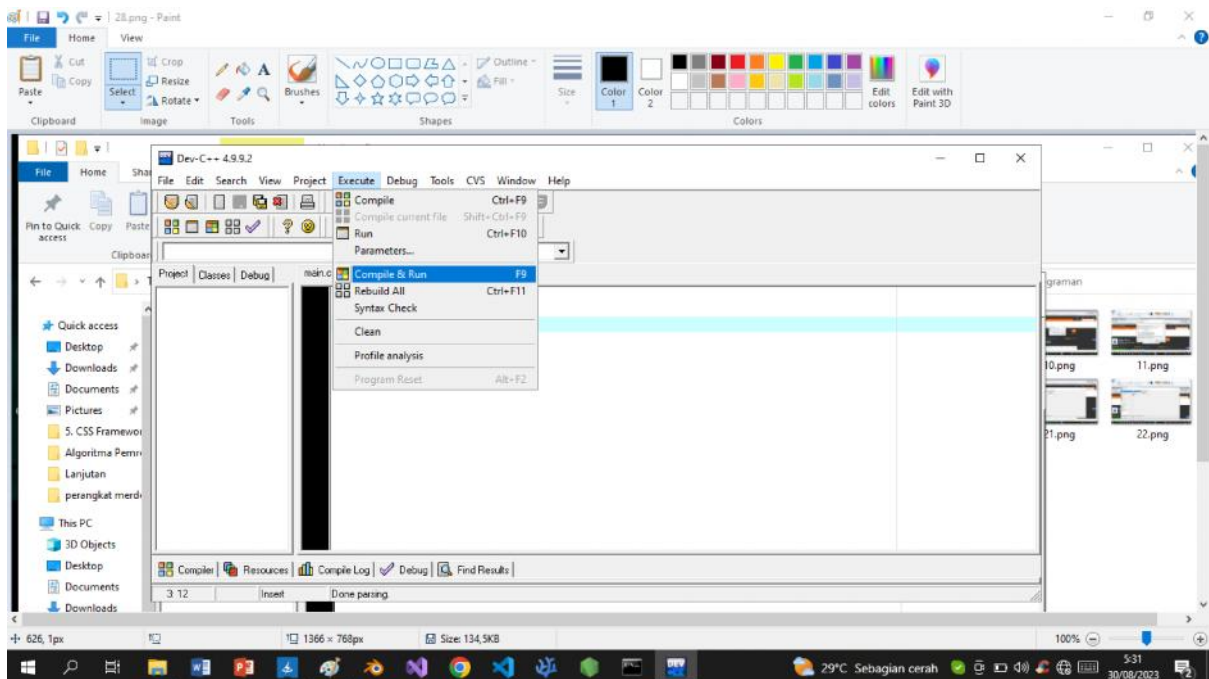
The status bar at the bottom indicates "5 25 Modified Insert 7 Lines in file".



4. Lalu klik ctrl + s untuk memunculkan menu save. Lalu beri nama dengan ekstensi .c



5. Lalu klik execute -> compile & run untuk menjalankan program



## 6. Program berhasil dijalankan

