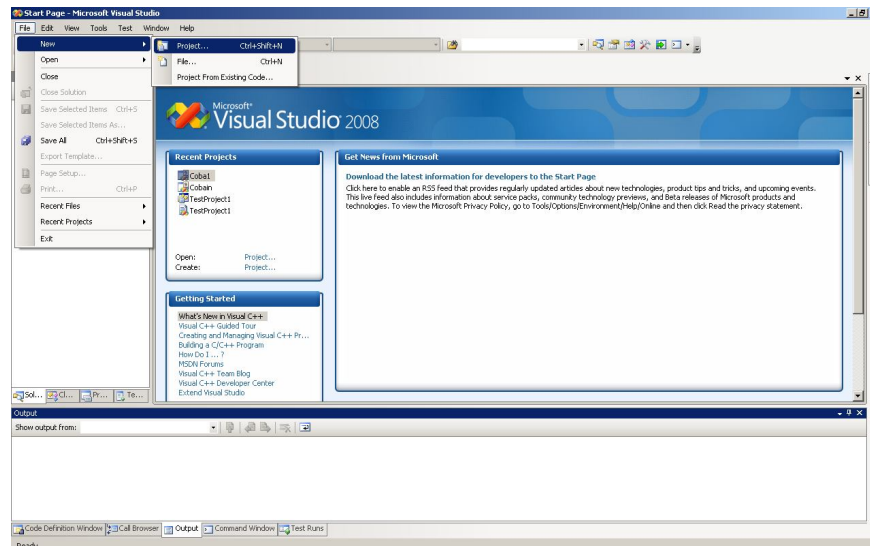


MODUL I MEMULAI BAHASA C

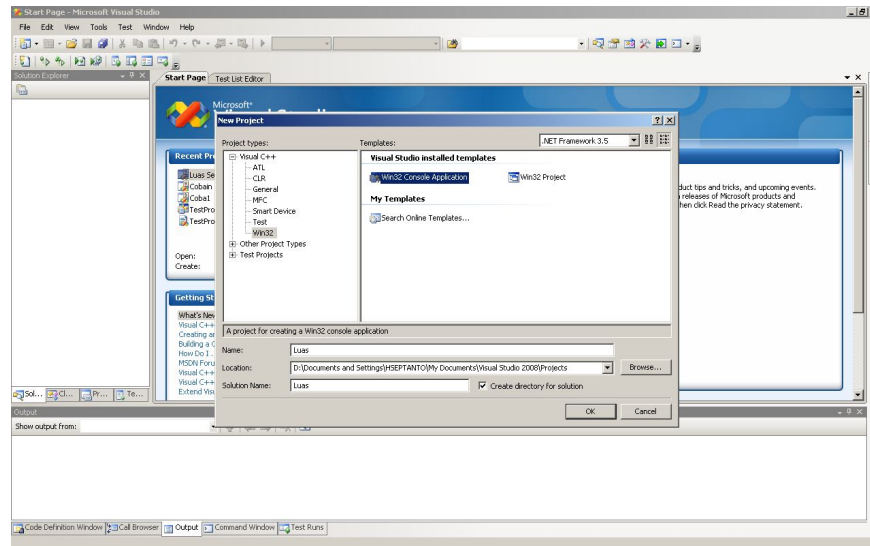
I.1. MEMBUAT PROGRAM C DENGAN VISUAL C++ 2008

Microsoft Visual C++ 2008 (MVC++ 2008) adalah IDE (*Integrated Development Environment*) yang digunakan dalam kuliah ini. Ikuti langkah-langkah berikut untuk memulai membuat program C dengan MVC++ 2008:

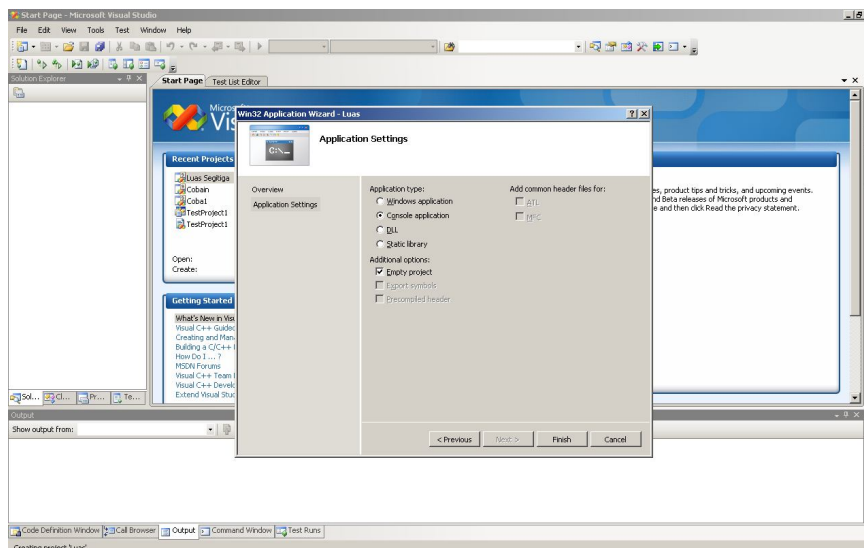
- a. Pada window **Start Page – Microsoft Visual Studio**, pilih **File→New→Project** atau tekan tombol **Ctrl+Shift+N**



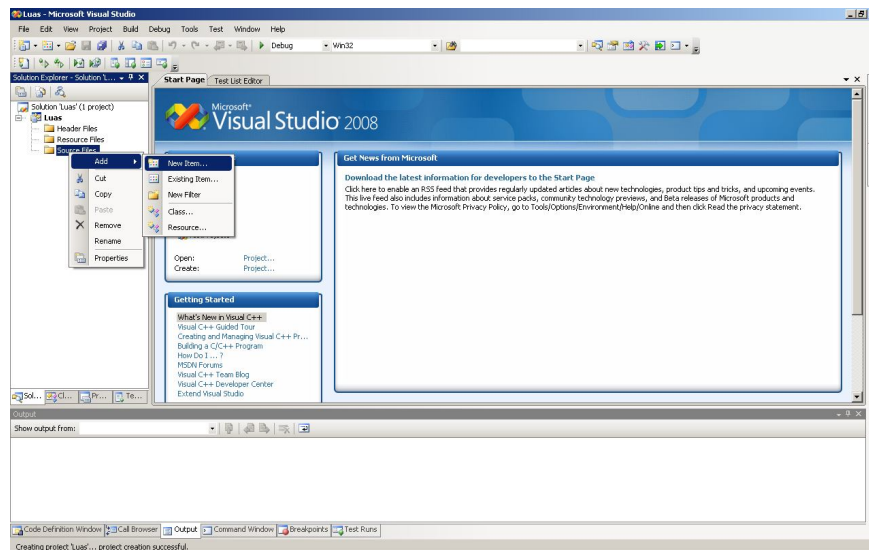
- b. Pada window **New Project**, pilih **Project types: Win32** dan pilih **Templates: Win32 Console Application**. Kemudian tuliskan beri nama project, misalnya, **Luas**. Tentukan pula lokasi untuk menyimpan file project. Akhiri dengan klik **OK**.



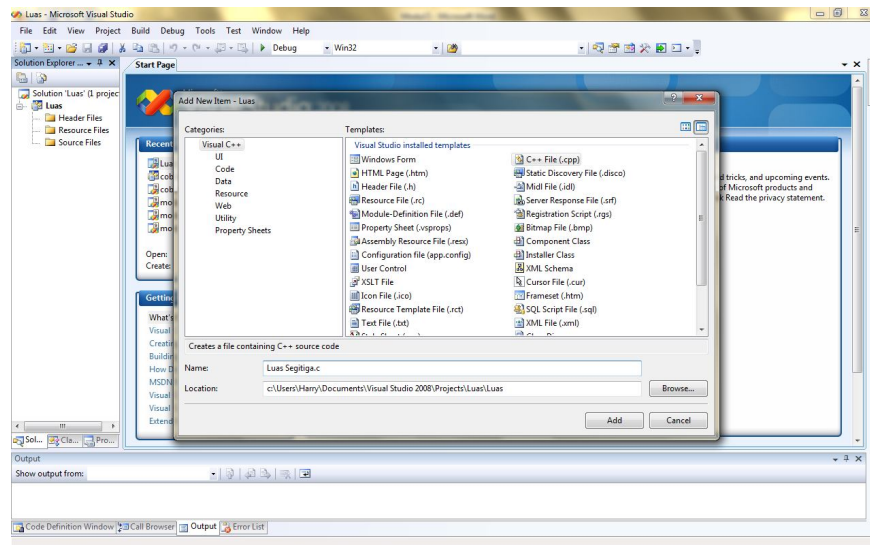
- c. Pada **Win32 Application Wizard**, klik **Next** atau pilih **Application Settings** → **Application type: Console application** dan hilangkan tanda centang pada **checkboxbox Additional options: Precompiled header**. Lalu, beri tanda centang pada **checkboxbox Additional options: Empty project**. Akhiri dengan klik **Finish**.



- d. Pada window **Luas – Microsoft Visual Studio**, klik kanan **Source Files**→pilih **Add**→**New Item...**

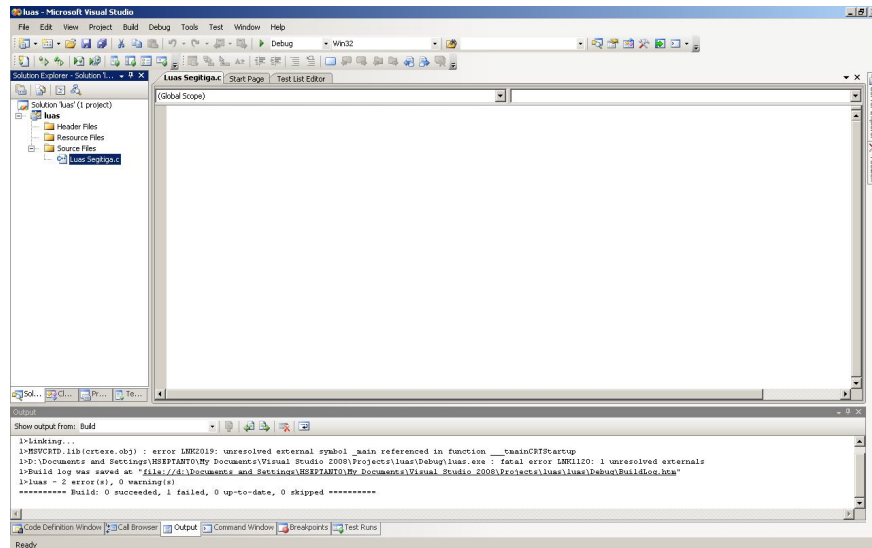


- e. Pada window **New Item – Luas** pilih **Categories: Visual C++** dan pilih **Templates: C++ File (.cpp)**. Beri nama, misalnya **Luas Segitiga.c** (Ket.: secara otomatis, nama yang diberikan memiliki extension **.c**, bukan **.cpp**). Akhiri dengan klik **Add**.



Modul I—Memulai Bahasa C

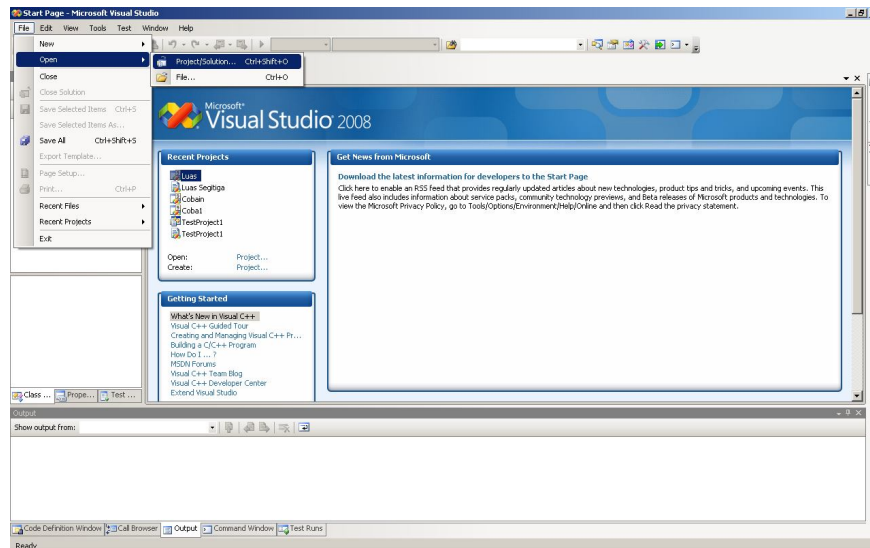
- f. Pada window **Luas – Microsoft Visual Studio**, klik ganda **Luas Segitiga.c** (tampil sebagai cabang dari Source Files).



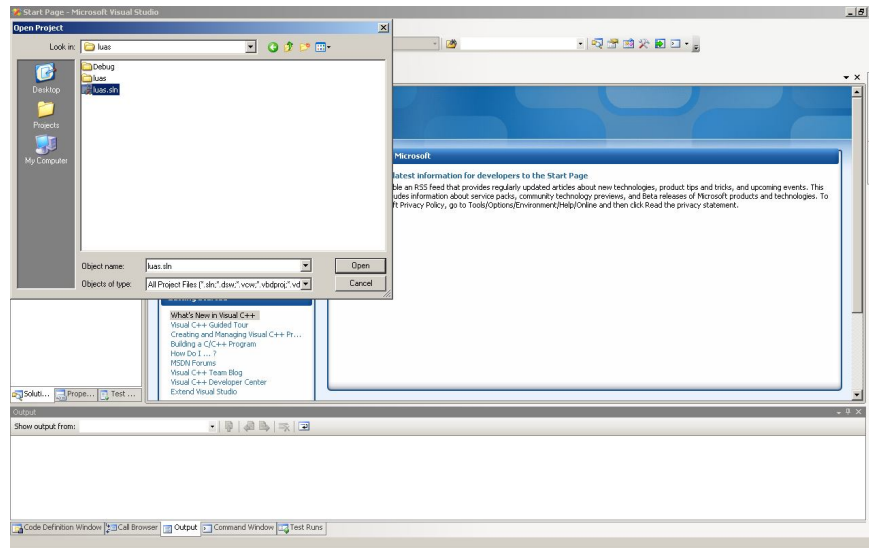
- g. Program C siap dibuat.

Memanggil File Program yang Sudah Dibuat

- h. Misalkan suatu program C telah dibuat. Langkah-langkah untuk memanggil kembali program tersebut, yaitu: pilih **File→Open→Project/ Solution...**



- i. Kemudian, cari file project yang telah dibuat (memiliki extension .sln)



I.2. KERANGKA DASAR BAHASA C

Kerangka program C diberikan sebagai berikut:

- Ada satu atau lebih fungsi, namun yang harus ada di dalam sebuah program C adalah fungsi `main()`
- Deklarasi variable
- Perintah (statement)
- Akses library
- Komentar

Perhatikan contoh program C sederhana berikut ini. Program C ini adalah program perhitungan luas segitiga.

Contoh Program 1

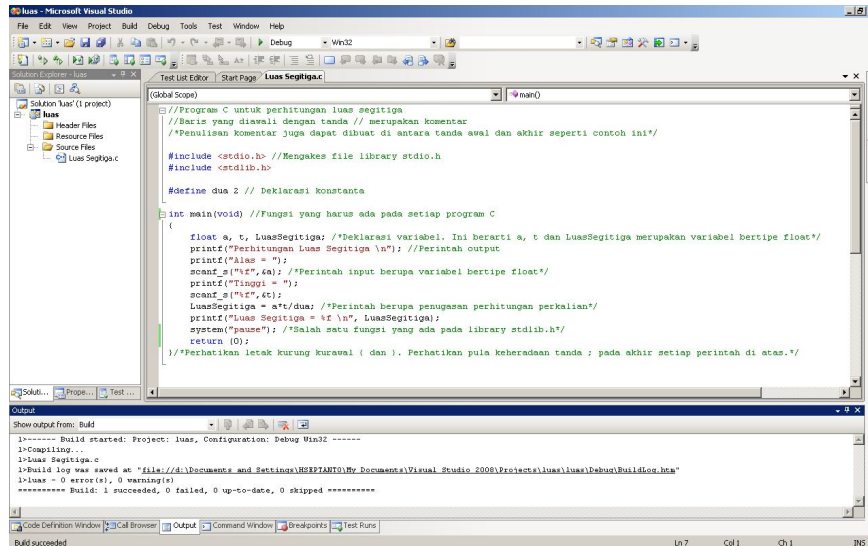
```
//Program C untuk perhitungan luas segitiga
//Baris yang diawali dengan tanda // merupakan komentar
/*Penulisan komentar juga dapat dibuat di antara tanda awal dan
akhir seperti contoh ini */

#include <stdio.h> //Mengakses file library stdio.h
#include <stdlib.h>

#define dua 2 //Deklarasi konstanta

int main(void) //Fungsi yang harus ada pada setiap program C
{
    float a, t, LuasSegitiga; /*Deklarasi variabel. Ini berarti
                                a, t dan LuasSegitiga merupakan
                                variabel bertipe float*/
    printf("Perhitungan Luas Segitiga \n"); //Perintah output
    printf("Alas = ");
    scanf_s("%f",&a); /*Perintah input berupa variabel bertipe
                        float*/
    printf("Tinggi = ");
    scanf_s("%f",&t);
    LuasSegitiga = a*t/dua; /*Perintah berupa penugasan
                            perhitungan perkalian*/
    printf("Luas Segitiga = %f \n", LuasSegitiga);
    system("pause"); /*Salah satu fungsi yang ada pada library
                      stdlib.h*/
    return (0); //Nilai balikan fungsi main() adalah 0
}/*Perhatikan letak kurung kurawal tutup dan kurung kurawal buka.
Perhatikan pula keberadaan tanda ; pada akhir setiap perintah
di atas.*/
```

Ketiklah contoh program C berikut. Lalu compile dengan cara menekan tombol Ctrl+F7. Pada bagian bawah window Microsoft Visual Studio, yaitu Output dari Build, akan terdapat keterangan kesalahan yang terjadi.



Kemudian jalankan (Start Debugging) dengan menekan tombol F5. Lakukan pengamatan.

I.3. TIPE DATA, VARIABEL DAN KONSTANTA

I.3.1 Tipe Data

Bahasa C mendukung beberapa tipe data. Masing-masing tipe data berkaitan dengan cakupan nilai dan besar memori yang digunakan. Tipe-tipe data tersebut diberikan pada tabel di bawah ini.

Tipe Data	Keterangan	Penggunaan Memori
int	Bilangan bulat	2 atau 4 byte (berbeda pada setiap compiler)
char	Karakter tunggal	1 byte
float	Meliputi bilangan desimal	4 byte
double	Bilangan decimal dengan tingkat ketelitian lebih besar daripada float	8 byte

Tipe-tipe data dasar di atas dapat pula digunakan bersamaan dengan keterangan-sifatnya (qualifier), yaitu short, long, signed dan unsigned. Contohnya, long int dan long double. Penggunaan *qualifier* ini juga menentukan cakupan nilai dan besar memori yang digunakan.

I.3.2 Variabel

Variabel adalah media untuk mendapatkan dan menyimpan data. Sedangkan melakukan deklarasi variable berarti menuliskan program yang menyatakan satu atau beberapa variabel beserta tipe data-nya dan diakhiri dengan tanda titik-koma ; .

Perhatikan kembali Contoh Program 1; pada contoh program di atas tertulis:

```
float a, t, LuasSegitiga;
```

Interpretasi potongan program tersebut adalah melakukan deklarasi variabel sehingga variabel `a`, `t` dan `LuasSegitiga` yang memiliki tipe data `float`. Pernyataan di atas dapat juga dikatakan bahwa `a`, `t` dan `LuasSegitiga` adalah variabel-variabel `float`.

I.3.3 Konstanta

Perhatikan bahwa pada Contoh Program 1:

```
#define dua 2 //Deklarasi konstanta
```

Interpretasi potongan program di atas adalah melakukan deklarasi konstanta dengan simbol-nama `dua` untuk menggantikan (substitusi) angka 2.

Selain untuk menggantikan angka, dengan deklarasi konstanta, karakter dan string (kumpulan karakter) juga dapat disubsitusi dengan suatu simbol-nama. Tentang hal ini akan dibahas lebih lanjut di dalam materi tentang Array.

I.4. FUNGSI `main()`

Fungsi `main()` adalah fungsi yang harus ada pada setiap program C. Dengan demikian simbol-nama `main` tidak dapat diubah dengan simbol-nama lain sehingga, misalnya, menjadi `nima()`. Selain itu juga, program C juga mengartikan huruf besar dan huruf kecil berbeda (*case-sensitive*). Misalnya, `h` berbedan dengan `H`. Dengan demikian, fungsi `main()` juga tidak bisa diubah namanya dengan `MAIN()`.

Perhatikan bahwa pada Contoh Program 1 tertulis potongan program seperti di bawah ini:

```
int main(void)
{
    .
    .
    .
    return (0);
}
```

Interpretasi potongan program di atas adalah sebagai berikut:

- Tipe data `int` pada `int main (void)` berarti nilai balikan (`return`) dari fungsi `main` bertipe integer.
- Perintah `return(0);` berarti nilai balikan dari fungsi `main` samadengan nol.
- Simbol `void` pada `int main (void)` berarti fungsi `main` tidak menerima data apapun sebelum program mulai dijalankan.

I.5. Fungsi `printf()` dan Fungsi `scanf()`

Fungsi `printf()` dan fungsi `scanf()` merupakan fungsi-fungsi yang terdapat pada file library/ file header standard `stdio.h`.

Fungsi `printf()` digunakan untuk menampilkan informasi/ data (output). Perhatikan potongan program pada Contoh Program 1 di bawah ini

```
printf("Perhitungan Luas Segitiga \n");
printf("Alas = ");
```

Interpretasi potongan program di atas adalah:

- Pada layar akan ditampilkan kalimat: Perhitungan Luas Segitiga
- Simbol `\n` berarti kalimat `Alas =` akan ditampilkan pada baris baru (*new line*)

Perhatikan kembali potongan program berikut:

```
printf("Luas Segitiga = %f \n", LuasSegitiga);
```

Interpretasi potongan program di atas adalah:

- Pada layar akan ditampilkan kalimat `Luas Segitiga =`.
- Simbol `%f` berarti kalimat tadi diikuti dengan dengan bilangan bertipe `float` yang nilainya bergantung dari nilai variabel `LuasSegitiga`.

Fungsi `scanf()` digunakan untuk meminta/ menerima data nilai dari user (input). Adapun fungsi `scanf()` memiliki kegunaan yang sama dengan `scanf_s()`. Fungsi `scanf_s()` merupakan fungsi pengganti `scanf()` yang disarankan oleh MVC++ karena alasan keamanan (*security*). Untuk saat ini dapat diabaikan. Perhatikan potongan program di bawah ini:

```
scanf("%f", &a);
```

Interpretasi potongan program di atas yaitu:

- Komputer meminta/ menerima nilai yang akan dikonversikan menjadi nilai bertipe `float` (ditandai dengan simbol `%f`) dari user.
- Simbol `&a` berarti variabel `a` memiliki nilai samadengan nilai input tadi.

Tabel di bawah ini (sumber: *Programming With C*, Schaum Outline) merupakan tabel karakter yang dapat digunakan untuk mengkonversikan nilai seperti pada contoh di atas.

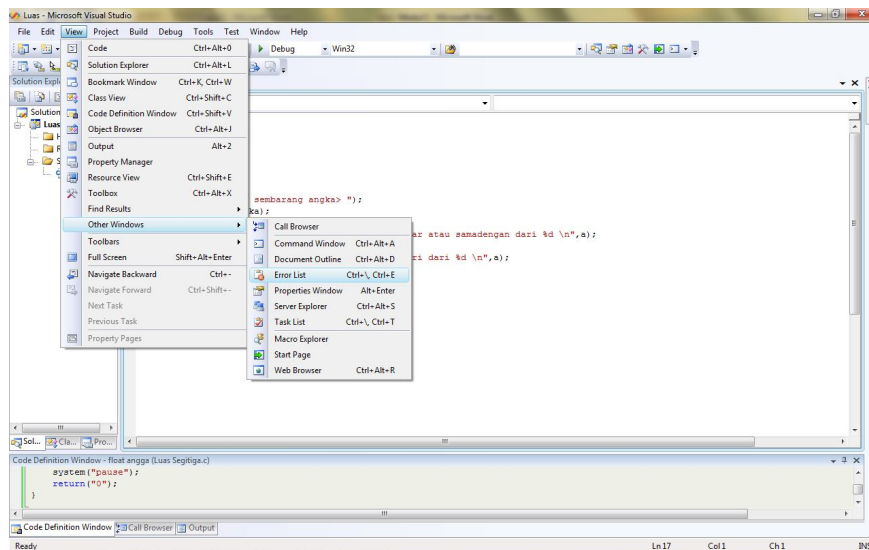
<i>Conversion Character</i>	<i>Meaning</i>
<code>c</code>	data item is a single character
<code>d</code>	data item is a decimal integer
<code>e</code>	data item is a floating-point value
<code>f</code>	data item is a floating-point value
<code>g</code>	data item is a floating-point value
<code>h</code>	data item is a short integer
<code>i</code>	data item is a decimal, hexadecimal or octal integer
<code>o</code>	data item is an octal integer
<code>s</code>	data item is a string followed by a whitespace character (the null character <code>\0</code> will automatically be added at the end)
<code>u</code>	data item is an unsigned decimal integer
<code>x</code>	data item is a hexadecimal integer
<code>[. . .]</code>	data item is a string which may include whitespace characters (see explanation below)

I.6. Menemukan Kesalahan

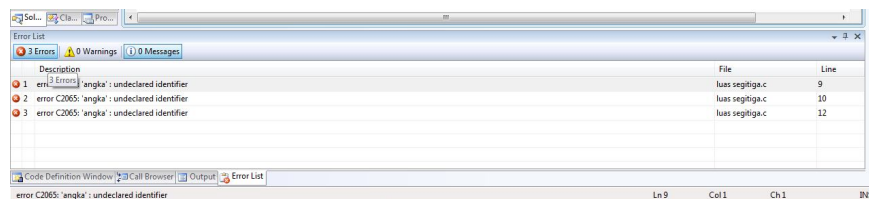
Setiap kompilasi dari program yang terdapat kesalahan, pada kolom **Output** akan ditampilkan keterangan gagal (*failed*) berikut: ===== Build: 0 succeeded, 1 failed, 0 up-to-date, 0 skipped =====

Adapun kesalahan yang terjadi sehingga menyebabkan kegagalan kompilasi dapat dilihat dengan mengikuti langkah-langkah berikut:

- a. Pilih **View**→**Other Windows**→**Error List**



- b. Sehingga pada bagian bawah window **Microsoft Visual Studio** ditampilkan keterangan error:



- c. Double-click pada list di bagian **Description** untuk mengetahui posisi kesalahan pada code.

I.7. Latihan

- a. Perhatikan potongan program berikut pada Contoh Program 1.:

```
printf("Perhitungan Luas Segitiga \n");
```

Hilangkan simbol `\n` pada potongan program di atas. Kemudian, compile dan jalankan. Lakukan pengamatan.

- b. Perhatikan Contoh Program 1. Hilangkan potongan program berikut:

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

Kemudian compile. Lakukan pengamatan.

- c. Perhatikan Contoh Program 1. Hilangkan secara bergantian dua potongan program berikut :

```
#include <stdlib.h>
```

dan

```
system("pause");
```

Kemudian compile dan jalankan. Lakukan pengamatan.

- d. Perhatikan Contoh Program 1. Ubahlah tipe data variabel yang pada deklarasi variabel semula adalah bertipe float menjadi bertipe integer. Kemudian compile dan jalankan. Lakukan pengamatan.

- e. Seperti pada No. d, namun kali ini simbol `%f` pada:

```
printf("Luas Segitiga = %f \n", LuasSegitiga);
```

diubah menjadi `%d`. Kemudian compile dan jalankan. Lakukan pengamatan.

- f. Buatlah program untuk perhitungan luas lingkaran, dengan $\pi = 3.14159$.

- g. Compile dan jalankan program berikut. Lakukan pengamatan dan tambahkan komentar-komentar pada program tersebut.

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

int main(void)
{
    int tgl, bln, thn;
    printf("Tanggal-Bulan-Tahun Hari Ini\n");
    printf("\n");
    printf("Ketik tanggal, bulan dan tahun, secara berurutan.");
    printf("Ket: Tekan SPACEBAR untuk memisahkan tanggal, bulan dan tahun. Lalu, akhiri dengan menekan ENTER \n ");
    scanf_s("%d %d %d",&tgl,&bln,&thn);
    printf("\n");
    printf("Hari ini adalah tanggal %d, bulan %d, tahun %d \n", tgl, bln, thn);
    system("pause");
    return (0);
}
```

- h. Kembali kepada Contoh Program 1; jalankan program dan berikan nilai Alas = 1.2 dan Tinggi = 1.3. Amati hasilnya.
- i. Seperti pada no. h, amati hasilnya setelah sebelumnya dilakukan perubahan pada bagian berikut:

```
printf("Luas Segitiga = %.1f \n", LuasSegitiga);
```

- j. Seperti pada no. h, amati hasilnya setelah sebelumnya dilakukan perubahan pada bagian berikut:

```
printf("Luas Segitiga = %.3f \n", LuasSegitiga);
```

Soal no k- n diberikan code yang memiliki kesalahan syntax (*syntax error*). Temukan dan perbaiki kesalahan tersebut. Kemudian jalankan dan laporkan hasilnya.

- k.

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

int main(void)
{
```

l.

```
printf("D4 Teknologi Media Digital \n");
system("pause");
return("0");
```

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

}

int b = 3;
printf("b = %f \n",b)
system("pause");
return("0");

}
```

m.

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

int Main(void)
{
    float ANGKA; /* Mendeklarasikan variabel ANGKA bertipe float
float
printf("Ketikan sembarang angka> ");
scanf("%f",&angka);
printf("Angka yang Anda ketikan = \n",angka);
system("pause");
return("0");
```

n.

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

int main(void)
{
    int a, b=3, c;
printf("Masukan angka> ");
scanf("%d",&a);
c = a+b;
printf("Setelah ditambah 3, yaitu %d + 3 = %d \n",
    &a, &c);
system("pause");
return("0");

}
```