

# Praktikum Dasar Pemrograman

## Pengenalan Bahasa C

2023

# 1 Tujuan

- Mahasiswa dapat membuat proyek di dalam IDE
- Mahasiswa dapat menunjukkan pengetahuan mereka tentang struktur program dalam bahasa C
- Mahasiswa dapat menunjukkan pengetahuan mereka tentang tipe data dalam bahasa C
- Mahasiswa dapat menunjukkan pengetahuan mereka tentang operator dalam bahasa C
- Mahasiswa mampu menggunakan fungsi untuk membaca masukan dari keyboard
- Mahasiswa mampu menggunakan fungsi untuk mencetak teks di layar

## 2 Mengenal Bahasa C

Bahasa C dikembangkan oleh Dennis M. Ritchie dan Brian W. Kernighan pada awal tahun 1970. Terdapat beberapa standar untuk bahasa C, standar di sini dapat diartikan sebagai guideline dalam menuliskan bahasa C. Beberapa Standar yang ada:

1. Definisi Kernighan & Ritchie (K&R)
2. ANSI-C (X-3.159 -1989-)
3. Definisi AT&T (untuk superset C, C++), dan
4. GNU Coding Standards.

Aplikasi atau kegunaan Bahasa C

1. Membuat sistem operasi dan program-program sistem.
2. Pemrograman yang "dekat" ke perangkat keras (misalnya untuk kontrol peralatan).
3. Membuat tool kit.
4. Menulis program aplikasi.

## 3 IDE (Integrated Development Environment)

IDE adalah singkatan dari "Integrated Development Environment" dalam bahasa Inggris. Dalam Bahasa Indonesia, IDE dapat diterjemahkan menjadi "Lingkungan Pengembangan Terintegrasi" atau "Ruang Kerja Pengembangan Terpadu." IDE adalah sebuah perangkat lunak yang dirancang untuk membantu pengembang perangkat lunak dalam proses pengembangan, pengkodean, dan pengujian aplikasi komputer.

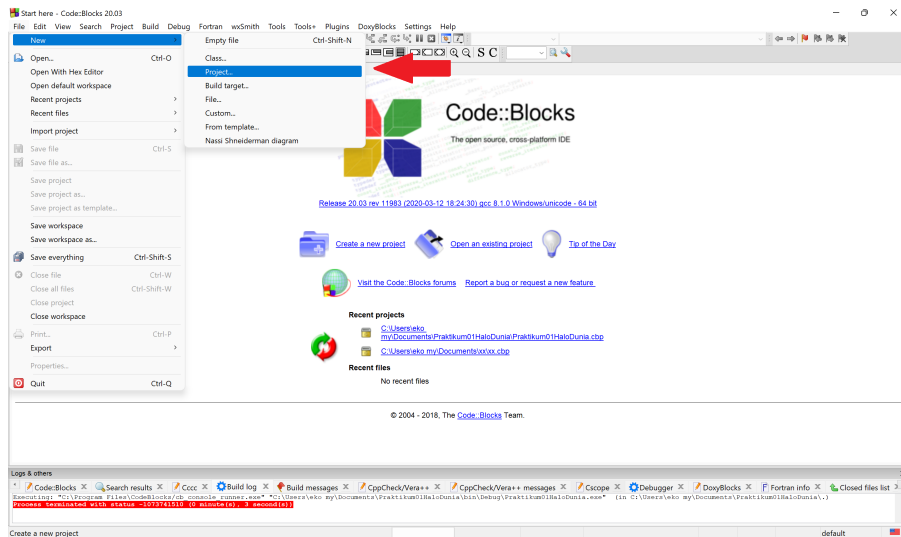
Berikut ini adalah daftar aplikasi IDE bahasa C yang dapat digunakan.

- CodeBlocks
- DevC++

## 4 Membuat proyek baru pada IDE Code::Blocks

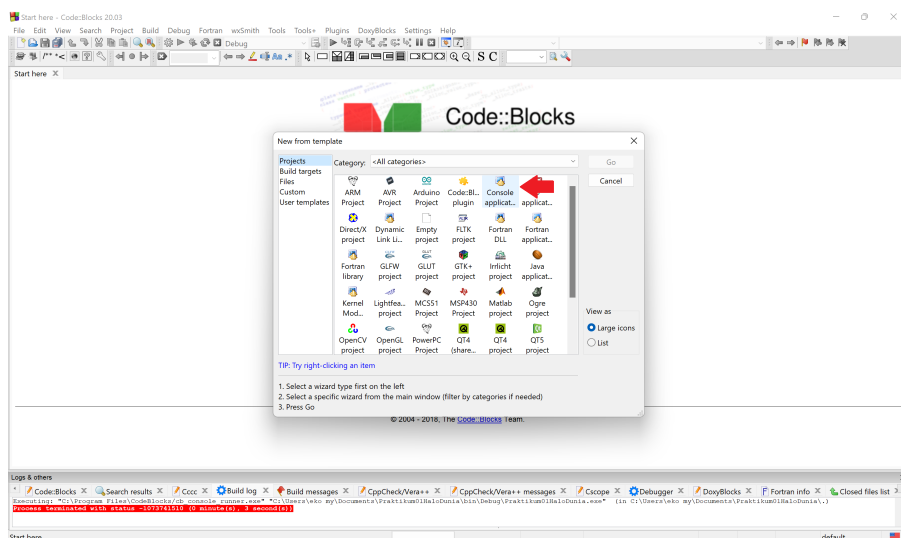
### 4.1 Langkah untuk membuat proyek baru

1. Go to File > New > Project



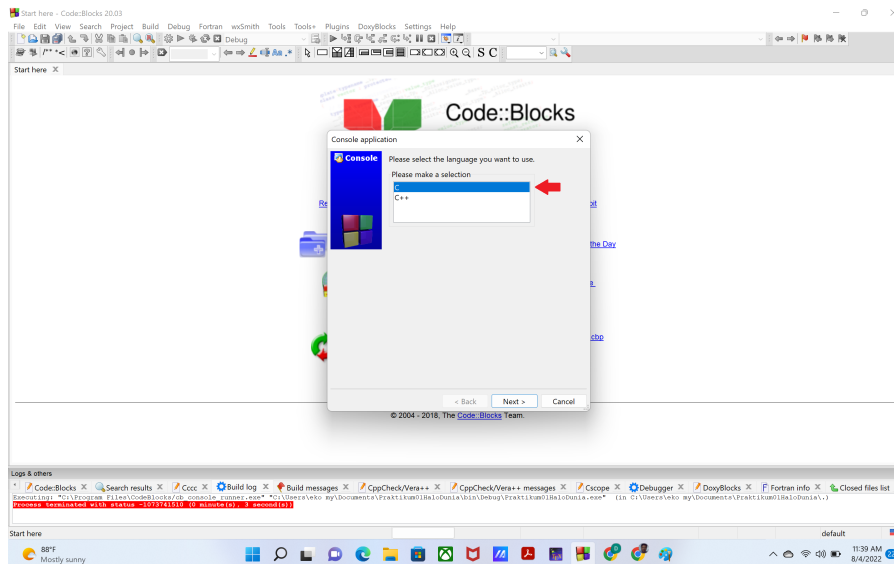
Gambar 1

2. Klik Console Application



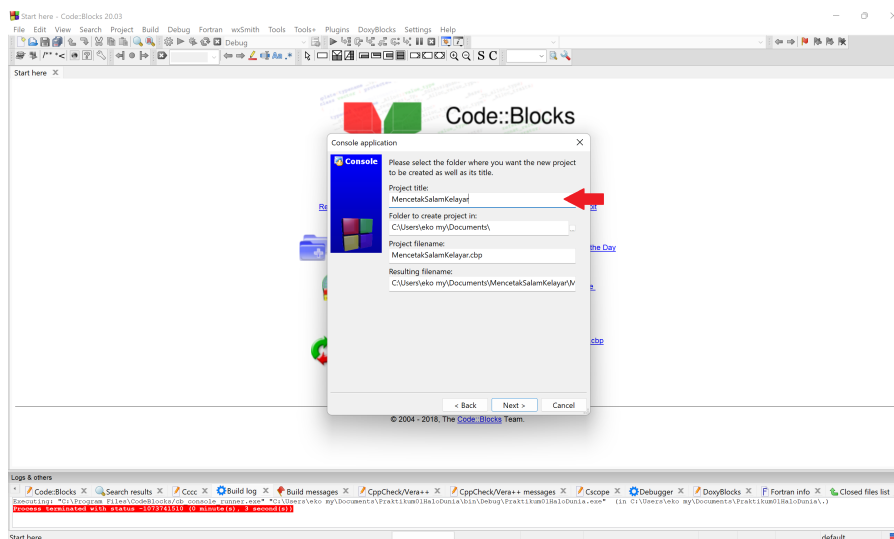
Gambar 2

3. Pilih C sebagai bahasa Pemrograman



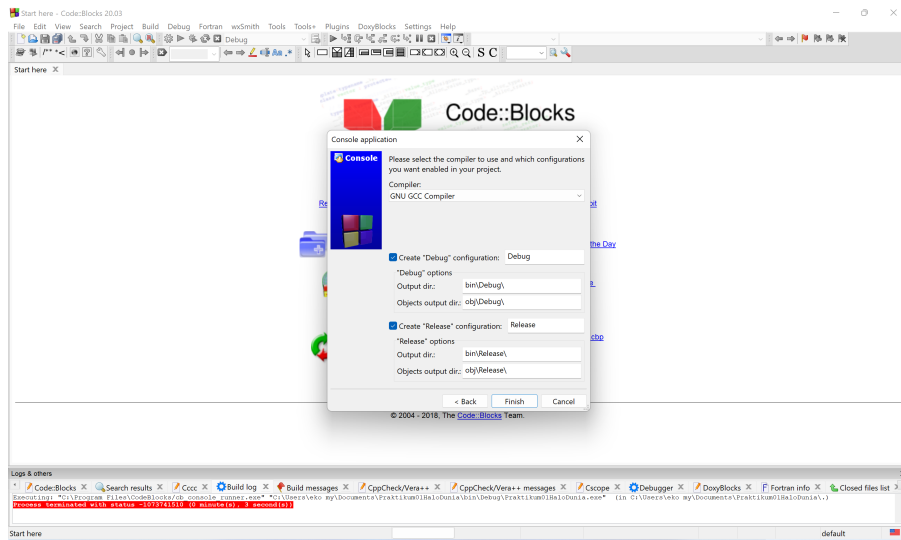
**Gambar 3**

4. Berikan nama ke project



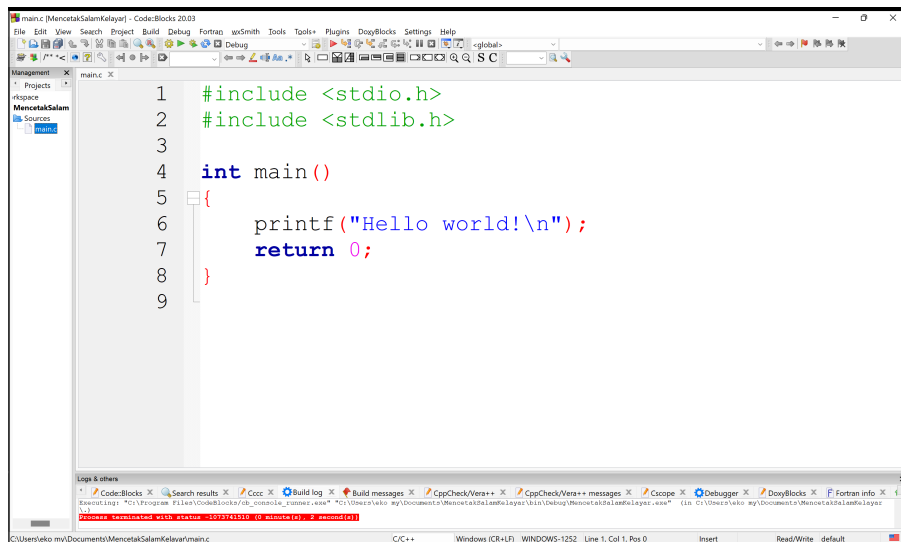
**Gambar 4**

5. Pilih compiler (gcc), pilih direktori untuk menyimpan, dan klik save.



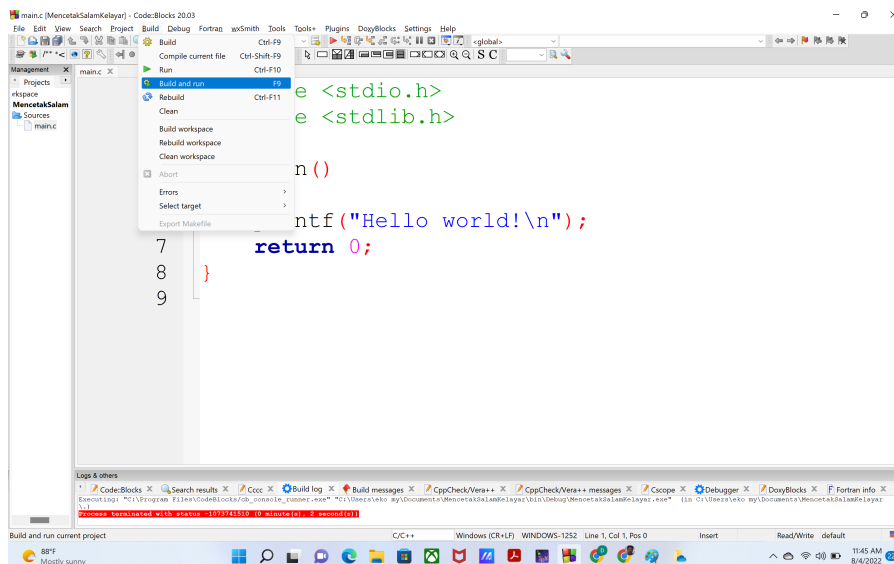
**Gambar 5**

6. Ketikan kode pada 6 ke Code::Blocks



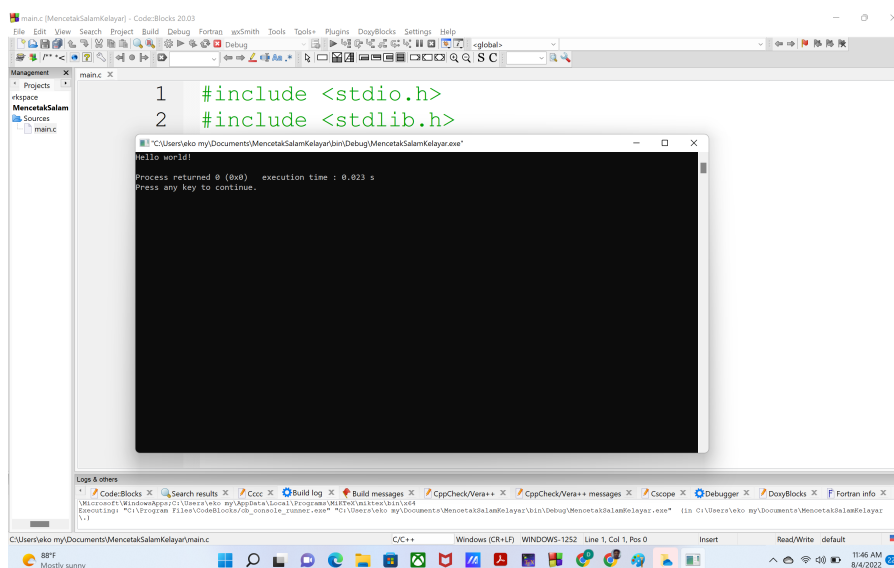
**Gambar 6**

7. Klik Build—>Build and Run atau tekan F9



Gambar 7

8. Keluaran dari program dapat dilihat di console



Gambar 8

## 4.2 Tugas Pendahuluan

1. Buatlah proyek dengan nama HaloDunia dan tulis program seperti gambar 6 ubahlah Hello World! dengan kata lain!

## 5 Struktur Bahasa Pemrograman C

Listing 1: Contoh program sederhana dalam bahasa C

```
1 #include <stdio.h>
2
3 int main()
4 {
```

```

5 //printing to screen
6 printf("Halo Dunia");
7 return 0;
8 }

```

Kode pada listing 1 adalah program mudah untuk mencetak "Halo Dunia" ke layar. Berikut penjelasan yang dilakukan setiap baris kode pada program.

Baris 1 : `#include <stdio.h>`

file library header untuk fungsi input dan output seperti `printf()` (contoh digunakan di baris 6)

Baris 2 : Baris kosong.

Baris 3 : `int main()`

Fungsi main. Fungsi utama adalah fungsi pertama yang akan dijalankan ketika program dimulai.

Baris 4 : `{`

Permulaan dari `main()` fungsi code block.

Baris 5 : `//printing to screen`

Komen. Komen digunakan untuk menjelaskan program. Komen akan diabaikan oleh program, tetapi membantu pembaca.

Baris 6 : `printf("Halo Dunia");`

Print/mencetak "Halo Dunia" ke layar.

Baris 7 : `return 0;`

Mengembalikan `main()` fungsi (sebuah fungsi berakhir ketika dikembalikan/return)

Baris 8 : `}`

Menutup `main()` fungsi code block.

## 5.1 Tugas Pendahuluan

1. Cobalah untuk menukar baris 6 dan baris 7 di listing 1. Apa yang terjadi? Jelaskan!
2. Bagaimana jika `return 0;` diganti dengan `return 1;`?

## 6 Tipe Data dan Variabel

### 6.1 Tipe Data

Pada bahasa C, terdapat beberapa tipe data untuk merepresentasikan data yang berupa bilangan bulat, bilangan real, karakter, string, dan lain-lain. Berikut adalah beberapa tipe data pada C.

**Tabel 1:** Beberapa tipe data di C

Tipe Data	Ukuran	Rentang Nilai
Int (or signed int)	2 bytes	-32,768 to 32,767
unsigned int	2 bytes	0 to 65,535
Short int(or signed short int)	2 bytes	-32,768 to 32,767
Long(or signed short int)	4 bytes	-2,147,483,648 to 2,147,483,647
unsigned long	4 bytes	0 to 4,294,967,295
float	4 bytes	1.2E-38 to 3.4E+38
double	8 bytes	2.3E-308 to 1.7E+308
Long double	10 bytes	3.4E-4932 to 1.1E+4932
char(or signed char)	1 byte	-128 to 127
unsigned char	1 byte	0 to 255

Untuk menampilkan data pada layar, setiap tipe data memiliki format specifier yang dapat digunakan pada formatted string. Berikut adalah format specifier untuk beberapa tipe data.

**Tabel 2:** Format Specifier

Format Specifier	Tipe Data
%d or %i	int
%f	float
%lf	double
%c	char
%s	string

Masih ada lebih banyak tipe data dari pada yang dituliskan pada Tabel 1. Tipe-tipe data ini dan spesifikasinya bisa ditemukan dengan mudah di internet.

### 6.1.1 Modifier

Secara garis besar modifier merupakan sebuah kata, frasa atau klausa yang mengubah makna atau menjelaskan kata benda(noun) atau kata kerja (verb) dalam sebuah kalimat. Sedangkan modifier dalam bahasa pemrograman adalah sebuah kata kunci atau keyword yang digunakan untuk mengubah perilaku atau sifat suatu elemen dalam program, seperti variabel, fungsi, atau kelas. Kita menggunakan modifier untuk mengubah range dari tipe data dasar untuk menyesuaikan dengan keperluan pemrograman. Ada empat modifier, yaitu:

1. signed  
`int value = -10;` (Menggunakan tanda negatif pada variabel bertipe int yang secara default adalah signed integer.)
2. unsigned  
`unsigned int count = 100;` (Menggunakan variabel unsigned integer untuk menyimpan bilangan bulat positif tanpa tanda.)
3. long  
`long population = 7500000000;` (Menggunakan tipe data long untuk menyimpan nilai yang lebih besar daripada tipe data int.)



#### 4. short

`short temperature = 20;` (Menggunakan tipe data short untuk menghemat memori ketika kita tahu bahwa nilai yang akan disimpan akan relatif kecil.)

## 6.2 Variabel

Variabel merupakan tempat menyimpan data. Mendeklarasikan suatu variabel dapat dilakukan dengan cara sebagai berikut.

**Listing 2:** Deklarasi Variabel C

```
1 DataType VariableName;
```

### 6.2.1 Operator Aritmatika dan Penugasan (Assignment)

Operator penugasan dapat dilakukan pada variabel yang tidak mempunyai `const` atau merupakan variabel nilai-l. Namun operator aritmatika dapat menerima kedua variabel dengan `const` atau tidak (nilai-l dan nilai-r). Tabel di bawah menunjukkan beberapa operator aritmatika di C.

**Tabel 3:** Operator Aritmatika di C

Operator	Nama	Contoh
+	Penambahan	$x + y$
-	Pengurangan	$x - y$
*	Perkalian	$x * y$
/	Pembagian	$x / y$
%	Modulo	$x \% y$

Tabel di bawah menunjukkan beberapa operator penugasan.

**Tabel 4:** Operator Penugasan

Operator	Contoh	Arti yang sama
=	$x = 5$	$x = 5$
+=	$x += 3$	$x = x + 3$
-=	$x -= 3$	$x = x - 3$
*=	$x *= 3$	$x = x * 3$
/=	$x /= 3$	$x = x / 3$
%=	$x \% = 3$	$x = x \% 3$
&=	$x \& = 3$	$x = x \& 3$
=	$x  = 3$	$x = x   3$
^=	$x \wedge = 3$	$x = x \wedge 3$
>>=	$x >> = 3$	$x = x >> 3$
<<=	$x << = 3$	$x = x << 3$

Terdapat juga "singkatan" untuk beberapa operator penugasan seperti  $x+=1$  dan  $x-1$  yaitu `++` dan `--` yang masing-masing disebut kenaikan dan penurunan. Singkatan ini digunakan seperti berikut.

```
x++;
```

```
x--;
```

```
++x;  
--x;
```

### 6.2.2 Operator Bitwise

Bitwise adalah operator khusus untuk menangani operasi logika bilangan biner dalam bentuk bit. Bilangan biner sendiri merupakan jenis bilangan yang hanya terdiri dari 2 jenis angka, yakni 0 dan 1. Jika nilai asal yang dipakai bukan bilangan biner, akan dikonversi secara otomatis oleh compiler C menjadi bilangan biner. Misalnya 7 desimal = 0111 dalam bilangan biner.

**Tabel 5:** Operator Bitwise

Operator	Nama	Contoh	Biner	Hasil (biner)	Hasil (decimal)
&	AND	10 & 12	1010 & 1100	1000	8
	OR	10   12	1010   1100	1110	14
^	XOR	10 ^ 12	1010 ^ 1100	0110	6
~	NOT	~10	~1010	0101	-11 (two complement)
<<	Left shift	10 << 1	1010 << 1	10100	20
>>	Right shift	10 >> 1	1010 >> 1	101	5

**Catatan:** Terdapat beberapa operator dalam Bahasa C. Silahkan pelajari dengan mencari referensi secara mandiri.

## 6.3 Tugas Pendahuluan

**Listing 3:** Menggunakan operator penugasan pada variabel const

```
1 #include <stdio.h>  
2 int main()  
3 {  
4     //deklarasi variabel const  
5     const int x=0;  
6     x=1;  
7     return 0;  
8 }
```

Coba jalankan program di Listing 3, apa yang terjadi?

## 7 Output dan Input

### 7.1 printf()

printf digunakan untuk mencetak string ke output yang dilengkapi dengan format specifier yang dimulai dengan % pada string.

```
printf(const char *format,v1,v2,...,vn)
```

Format specifier untuk beberapa tipe data dapat dilihat pada Tabel 2

### Contoh 7.1.1 Mencetak teks ke layar.

**Listing 4:** Mencetak Tulisan "C Programming" Ke layar

```
1  #include <stdio.h>
2  int main()
3  {
4      // Mencetak teks yang ditulis dalam simbol "
5      printf("C Programming");
6      return 0;
7  }
8
```

- Seluruh program C harus berisi fungsi main() tempat program memulai menjalankan kode.
- Fungsi printf() adalah library untuk mengirim output yang telah diformat ke layar. Fungsi printf() mencetak string dalam tanda dua tanda petik.
- return 0; pernyataan di main() fungsi memberitahu program untuk keluar.

### Contoh 7.1.2 Mencetak integer.

```
1  #include <stdio.h>
2  int main()
3  {
4      int testInteger = 5;
5      printf("Number = %d", testInteger); // <- %d format string
6      return 0;
7  }
8
```

Pada contoh ini digunakan format specifier %d untuk mencetak tipe data int. %d pada tex akan digantikan oleh isi dari testInteger.

### Contoh 7.1.3 Keluaran bilangan real (float atau double)

- Base : menggunakan float tipe data.
- Height: menggunakan float tipe data.
- Area : menggunakan float tipe data.

$$Luas = \frac{1}{2} \times Alas \times Tinnnggi \quad (1)$$

```
1  #include <stdio.h>
2
3  int main()
4  {
5      // deklarasi variabel
6      float Base;
7      float Height;
8      float Area;
9      // inisialisasi nilai
10     Base = 10;
11     Height = 5;
12     // menghitung area
```

```

13     Area = 0.5*Base*Height;
14     // mencetak area ke layar
15     printf("Area = %f",Area);
16     return 0;
17 }
18
19

```

## Penjelasan

**Baris 6-8** Alas, Tinggi dan Luas bertipe data float untuk menyimpan data parameter luas segitiga.

**Baris 10 dan 11** Memberi nilai ke Variabel Alas=10 dan Tinggi=5

**Baris 13** Menghitung luas alas sesuai dengan persamaan 7.1

**Baris 15** Mencetak Luas ke layar dengan menggunakan perintah printf.

.

## 7.2 scanf

Fungsi scanf(const char \*format, ...) membaca input dengan format.

### 1. Syntax

```
scanf(const char *format, ...)
```

### 2. Parameter

Format string pada C yang terdiri dari satu atau lebih yang terdiri dari Karakter Whitespace, Karakter Non-whitespace dan Format specifiers.

### 3. Return Value

Ketika berhasil maka fungsi mengembalikan jumlah item dari argumen yang berhasil di baca.

**Contoh 7.2.4** Menghitung luas segitiga dengan dengan alas Alas dan tinggi Tinggi yang diinputkan dari keyboard.

```

1  #include <stdio.h>
2
3  int main()
4  {
5      float Alas ,Tinggi,Luas;
6
7      printf("Menghitung luas segitiga\n");
8      printf("\nMasukkan Alas= ");
9      scanf("%f",&Alas);
10     printf("\nMasukkan Tinggi=");
11     scanf("%f",&Tinggi);
12     Luas = 0.5*Alas *Tinggi;
13     printf("Luas Seigtiga = %.2f", Luas);
14     return 0;
15 }
16

```

```
Menghitung Luas Segitiga
Masukan Panjang Alas= 4
Masukan Tinggi =3
Luas Segitiga = 6.00

...Program finished with exit code 0
Press ENTER to exit console.
```

Gambar 9

**Baris 9** `scanf("%f",&Alas);` meminta masukan untuk alas segitiganya

**Baris 11** `scanf("%f",&Tinggi);` meminta masukan tinggi segitiganya

**Baris 13** `printf("Triangle Area = %.2f", Luas);`, `%.2f` menandakan bahwa hanya 2 angka di belakang koma(decimal point) yang perlu dicetak.

**Contoh 7.2.5** Program untuk memasukkan nama dan email.

Pada contoh ini dipelajari bagaimana cara menginputkan string atau text dari keyboard dan mencetak kelayar. Input dari contoh program ini ada dua yang terdiri dari `sName` dan `sAlamatEmail`. Oleh karena text berisi banyak karakter maka masing-masing variabel dideklarasikan sebagai kumpulan karakter dengan jumlah karakter untuk `sName`=20 dan `sAlamatEmail`=30.

```
1  #include <stdio.h>
2
3  int main ()
4  {
5      char sName[20], sEmail[30];
6
7      printf("Masukkan Nama: ");
8      scanf("%19s", sName);
9
10     printf("Masukkan Email: ");
11     scanf("%29s", sEmail);
12
13     printf("Nama : %s\n", sName);
14     printf("Email:%s", sEmail);
15     return(0);
16 }
17
```

## 7.3 Escape Sequence

Escape Sequence adalah urutan karakter yang digunakan untuk memformat output dan tidak ditampilkan ketika dicetak ke layar. Setiap karakter mempunyai fungsi tertentu.

Tabel 6: Escape Sequence

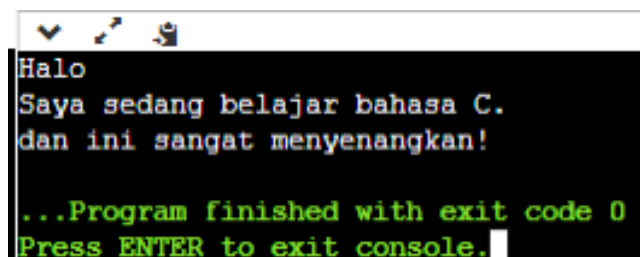
Escape sequence	Output berupa
\a	bell, alarm
\b	Backspace
\f	Ganti halaman
\n	Ganti baris
\r	Carriage return
\t	tab horisontal
\v	tab vertikal
\'	Petik tunggal
\"	Petik Ganda
\?	Tanda tanya
\\	Backslash

**Contoh 7.3.6** Mengubah baris dengan escape sequence \n.

```

1 #include <stdio.h>
2
3 int main()
4 {
5     printf("Halo \nSaya sedang belajar bahasa C.\ndan ini sangat menyenangkan!");
6     return 0;
7 }
8

```



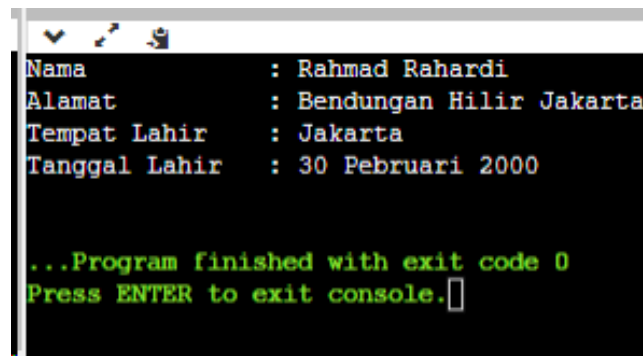
**Gambar 10**

**Contoh 7.3.7** Menggunakan escape sequence \t mengubah tab.

```

1 #include <stdio.h>
2 int main(void)
3 {
4     printf("Nama \t\t: Rahmad Rahardi\n");
5     printf("Alamat \t\t: Bendungan Hilir Jakarta\n");
6     printf("Tempat Lahir \t: Jakarta\n");
7     printf("Tanggal Lahir \t: 30 Pebruari 2000\n");
8
9     return (0);
10 }

```



```
Nama      : Rahmad Rahardi
Alamat    : Bendungan Hilir Jakarta
Tempat Lahir : Jakarta
Tanggal Lahir : 30 Pebruari 2000

...Program finished with exit code 0
Press ENTER to exit console.█
```

Gambar 11

## 7.4 Tugas Pendahuluan

1. Cobalah buat suatu program yang dapat menerima input berupa nama dan NRP kemudian menampilkannya pada layar.
2. Buat program yang meminta pengguna memasukkan suhu dalam Celsius dan kemudian mengonversinya ke Fahrenheit.

**Opsional:** Pelajari Git dan Github. Anda dapat memulai pembelajaran dari sumber berikut ini:

GitHub - <https://github.com>

Git - <https://git-scm.com/doc>