

Masukan dan Keluaran

Bahasa C

Baharuddin Aziz

Semester I 2019/2020

Tujuan Perkuliahan

- Mengetahui konsep masukan dan keluaran
- Memahami penerapan masukan dan keluaran pada bahasa C

Pendahuluan

Struktur Dasar Program

Pada dasarnya, program komputer hanya terdiri dari tiga bagian:

- **Masukan (*input*):** suatu data yang dimasukkan ke dalam program.
 - biasa dari perangkat masukan: *keyboard*, *mouse*, kamera, *mic*, dll.
- **Proses:** langkah-langkah dilakukan program untuk menghasilkan *output*.
- **Keluaran (*output*):** informasi yang dihasilkan setelah dilakukan proses.
 - biasanya ditampilkan ke layar komputer.



Gambar: Struktur dasar program komputer

Fungsi Keluaran

Fungsi Keluaran

Ada dua macam fungsi yang bisa digunakan untuk menampilkan *output*:

1. Fungsi `printf()`
2. Fungsi `puts()`

Fungsi `printf()` [1/2]

- Fungsi `printf()`: fungsi untuk menampilkan *output* ke layar komputer.
- Fungsi ini terdapat pada *library* `stdio.h`.
 - Harus menuliskan `#include <stdio.h>` di bagian atas program agar dapat menggunakan fungsi ini.



Gambar: Struktur dasar fungsi `printf()`

Fungsi `printf()` [2/2]

Perhatikan:

- **"format"**: sebuah teks (*string*) untuk ditampilkan.
- tanda **...** berisi sebuah variabel atau nilai untuk ditampilkan berdasarkan format yang diberikan pada teks "format".



Gambar: Struktur dasar fungsi `printf()`

Contoh Program: output.c

```
#include <stdio.h>

int main(){
    printf("Hello, ini teks output\n");
    printf("Nama saya %s\n", "Budi");
    printf("Usia saya %d\n", 18);
    return 0;
}
```

- Fungsi `printf()` memakai simbol `%s`, `%d`, `\n` untuk format teks.
 - `%s` : menampilkan nilai **string**;
 - `%d` : menampilkan nilai **bilangan bulat**;
 - `\n` : membuat **baris baru**.
- Selain tiga simbol tersebut, masih banyak lagi simbol yang lain.

Simbol Format Teks (Umum)

Simbol	Arti atau Fungsi
<code>%c</code>	untuk menampilkan karakter
<code>%s</code>	untuk menampilkan teks (<i>string</i>)
<code>%d</code> atau <code>%i</code>	untuk menampilkan bilangan bulat
<code>%f</code>	untuk menampilkan bilangan pecahan
<code>%o</code>	untuk menampilkan bilangan oktal
<code>%x</code>	untuk menampilkan bilangan heksadesimal
<code>\t</code>	untuk menampilkan tab

Fungsi `puts()`

- Fungsi `puts()`: fungsi untuk menampilkan *output*.
 - Fungsi ini juga terdapat pada library `stdio.h`.
- Perbedaanya dengan `printf()`, fungsi `puts()` **tidak** memerlukan format.
- Fungsi `puts()` akan **selalu** membuat baris baru.
 - Tidak perlu menggunakan simbol `\n` seperti pada `printf()`.

<https://www.petanikode.com>



```
puts("Hello World");
```

teks untuk ditampilkan

Gambar: Struktur dasar fungsi `puts()`

Contoh program: output1.c

```
#include <stdio.h>

int main() {
    puts("Hello World");
    puts("Nama Saya Budi");
    return 0;
}
```

```
#include <stdio.h>

int main(){
    printf("Hello, ini adalah teks output\n");
    printf("Nama saya %s\n", "Budi");
    printf("Usia saya %d\n", 18);
    return 0;
}
```

Contoh program: output.c

Fungsi `putchar ()`

Selain fungsi `puts ()`, terdapat juga fungsi `putchar ()` yang khusus untuk menampilkan satu karakter saja.

Contoh program: output2.c

```
#include <stdio.h>

int main() {
    putchar('A');
    return 0;
}
```

Output

A

Pilih `printf()` atau `puts()`

- Pertanyaan:
Fungsi manakah yang harus dipakai?
- Jawabannya, tergantung...
Tergantung dari teks apa yang akan ditampilkan.
- Apabila ingin menampilkan **teks** dan **nilai** atau variabel, sebaiknya menggunakan `printf()` karena dapat membuat format.
- Jika hanya teks/kalimat **biasa** (tidak melibatkan variabel), pakai `puts()`.

Fungsi Masukan

Fungsi Masukan

Pada bahasa C, terdapat beberapa fungsi untuk mengambil *input*.

- *input* yang akan dibahas adalah *input* dari keyboard.

Input juga bisa dari file, kamera, mikrofon, dan sebagainya.

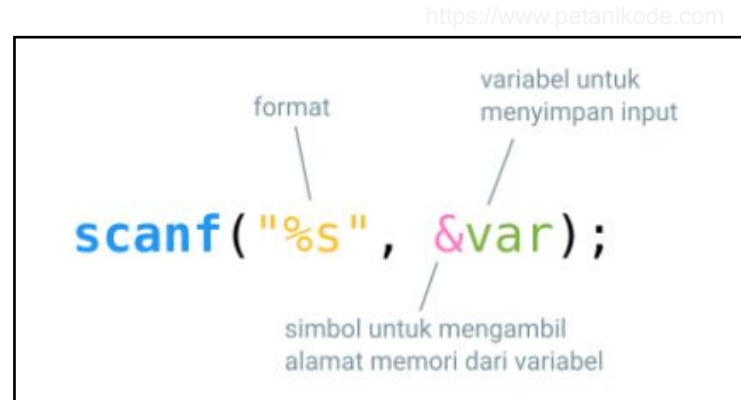
- untuk sekarang fokus membahas yang paling mudah.

Beberapa fungsi untuk mengambil *input* dari keyboard:

1. Fungsi `scanf ()`
2. Fungsi `gets ()`
3. Fungsi `fgets ()`

Fungsi scanf ()

- Fungsi **scanf ()** : fungsi untuk mengambil input dari keyboard.
 - Memiliki format seperti fungsi `printf ()` .
 - Format yang diberikan tergantung jenis tipe data apa yang ingin kita ambil.
 - Misalnya mau mengambil angka, maka kita bisa pakai `%d` atau `%i`.



Gambar: Struktur dasar fungsi `scanf()`

Contoh Program: `input.c`

```
#include <stdio.h>

int main () {
    // membuat variabel
    char name[20], web_address[30];

    printf("Nama: ");
    scanf("%s", &name);
    printf("Alamat web: ");
    scanf("%s", &web_address);

    printf("\n-----\n");
    printf("\nMasukan yang dibaca:\n");
    printf("Nama      : %s\n", name);
    printf("Alamat Web: %s\n", web_address);

    return 0;
}
```

- *Coba input:*
`Indonesia`
dan
`Indonesia Raya`
- *Bagaimana hasilnya?*
 - *Sama?*
 - *Berbeda?*
 - *Jika berbeda mengapa?*

Contoh Program: `input.c`

```
#include <stdio.h>

int main () {
    // membuat variabel
    char name[20], web_address[30];

    printf("Nama: ");
    scanf("%[^\\n]s", name);
    printf("Alamat web: ");
    scanf("%s", &web_address);

    printf("\\n-----\\n");
    printf("\\nMasukan yang dibaca:\\n")
    printf("Nama      : %s\\n", name);
    printf("Alamat Web: %s\\n", web_address);

    return 0;
}
```

- `[]` adalah karakter scanset.
- `[^\\n]` memberitahu jika masukan bukan *newline* ('\\n'), maka diambil sebagai masukan.

Fungsi `scanf()` dan simbol `&`

- Dalam menggunakan `scanf()` dianjurkan menggunakan simbol `&` sebelum nama variabel.
- Simbol `&` berfungsi mengambil **alamat memori** dari suatu variabel.
- Fungsi `scanf()` membutuhkan tempat untuk menyimpan nilai yang akan dimasukkan.
- Oleh karena itu, diberikan simbol `&` di depan nama variabel untuk menentukan alamat memori yang akan digunakan oleh `scanf()`.

Contoh Program: input1.c

```
#include <stdio.h>

void main(){
    int a, b, c;

    printf("Inputkan nilai a: ");
    scanf("%i", &a);
    printf("Inputkan nilai b: ");
    scanf("%i", &b);
    c = a + b;
    printf("Hasil a + b: %i", c);
}
```

```
#include <stdio.h>

int main () {
    // membuat variabel
    char name[20], web_address[30];

    printf("Nama: ");
    scanf("%[^\\n]s", name);
    printf("Alamat web: ");
    scanf("%s", &web_address);

    printf("\\n-----\\n");
    printf("\\nMasukan yang dibaca:\\n")
    printf("Nama      : %s\\n", name);
    printf("Alamat Web: %s\\n", web_address);

    return 0;
}
```

Contoh Program: input.c

Fungsi `gets ()`

- Fungsi `gets ()` : fungsi untuk mengambil *input* dalam satu baris.
- Fungsi `gets ()` tidak memerlukan format seperti `scanf ()`.



Gambar: Struktur dasar fungsi `gets()`

Contoh Program: input2.c

```
#include <stdio.h>

int main () {
    char name[50], web_address[50];

    printf("Nama: ");
    gets(name);

    printf("Alamat web: ");
    gets(web_address);
    printf("Nama: %s\n", name);
    printf("Alamat web: %s\n", web_address);

    return(0);
}
```

```
#include <stdio.h>

int main () {
    // membuat variabel
    char name[20], web_address[30];

    printf("Nama: ");
    scanf("%[^\n]s", name);
    printf("Alamat web: ");
    scanf("%s", &web_address);

    printf("\n-----\n");
    printf("\nMasukan yang dibaca:\n");
    printf("Nama      : %s\n", name);
    printf("Alamat Web: %s\n", web_address);

    return 0;
}
```

Contoh Program: input.c

Fungsi `fgets()`

Fungsi `fgets()` lebih aman dibandingkan `gets()`, karena kita bisa menentukan ukuran **buffer** dan **sumber** inputan.

- **ukuran buffer**: batas ukuran panjang *string* yang diinputkan.
- **stdin**: sumber inputan dari *keyboard*.



Gambar: Struktur dasar fungsi `fgets()`

Contoh Program: input3.c

```
#include <stdio.h>

int main(){
    char name[50], email[50];

    printf("Nama: ");
    fgets(name, sizeof(name), stdin);
    printf("Email: ");
    fgets(email, sizeof(email), stdin);
    printf("\n-----\n");
    printf("Nama anda: %s", name);
    printf("Alamat email: %s", email);

    return 0;
}
```

```
#include <stdio.h>

int main () {
    // membuat variabel
    char name[20], web_address[30];

    printf("Nama: ");
    scanf("%[^\n]s", name);
    printf("Alamat web: ");
    scanf("%s", &web_address);

    printf("\n-----\n");
    printf("\nMasukan yang dibaca:\n");
    printf("Nama      : %s\n", name);
    printf("Alamat Web: %s\n", web_address);

    return 0;
}
```

Contoh Program: input.c

Pertanyaan?