

BAB III

PERINTAH INPUT OUTPUT

3.1. Memasukkan Data

Dalam bahasa C proses memasukkannya data bisa menggunakan beberapa fungsi pustaka yang telah tersedia. Beberapa fungsi pustaka yang bisa digunakan adalah:

`scanf()`

- Fungsi pustaka `scanf()` digunakan untuk menginput data berupa data numerik, karakter dan string secara terformat.
- Hal-hal yang perlu diperhatikan dalam pemakaian fungsi `scanf()` :
 - ✓ Fungsi `scanf()` memakai penentu format
 - ✓ Fungsi `scanf()` memberi pergantian baris secara otomatis
 - ✓ Fungsi `scanf()` tidak memerlukan penentu lebar field
 - ✓ Variabelnya harus menggunakan operator alamat &
- Kode penentu format :
 - ✓ `%c` : Membaca sebuah karakter
 - ✓ `%s` : Membaca sebuah string
 - ✓ `%i, %d` : Membaca sebuah bilangan bulat (integer)
 - ✓ `%f, %e` : Membaca sebuah bilangan pecahan (real)
 - ✓ `%o` : membaca sebuah bilangan octal
 - ✓ `%x` : Membaca sebuah bilangan heksadesimal
 - ✓ `%u` : Membaca sebuah bilangan tak bertanda

Contoh Program :

```
/* Program memasukan input dengan beberapa tipe data */  
#include <stdio.h>  
#include <conio.h>  
void main() {  
    int jumlah;  
    char huruf, nim[10];
```

```
float nilai;
clrscr();
printf("Masukkan sebuah bilangan bulat : ");
scanf("%d", &jumlah); /* membaca sebuah bilangan bulat */
printf("Masukkan sebuah karakter : ");
scanf("%c", &huruf); /* membaca sebuah karakter */
printf("Masukkan nim Anda : ");
scanf("%s", &nim); /* membaca sebuah string */
printf("Masukkan sebuah bilangan pecahan : ");
scanf("%f", &nilai); /* membaca sebuah bilangan float */
printf("\n Nilai variable yang Anda masukkan adalah :\n");
printf("jumlah = %d\n", jumlah);
printf("huruf = %c\n", huruf);
printf("nim = %s\n", nim);
printf("nilai = %f\n", nilai);
getch();
}
```

gets()

- Fungsi gets() digunakan untuk memasukkan data bertipe karakter dan tidak dapat digunakan untuk memasukkan data numerik.
- Harus diakhiri dengan penekanan tombol enter
- Cursor secara otomatis akan pindah baris
- Tidak memerlukan penentu format

Contoh Program :

```
/* Program input tipe data karakter/string */
#include <stdio.h>
#include <conio.h>
void main() {
    char nama[20];
    clrscr();
    printf("Masukkan nama Anda : ");
    gets(nama);
}
```

```
printf("Hello, NamaAndaadalah %s", nama);  
getch();  
}
```

getchar()

- Fungsi `getchar()` digunakan untuk membaca data yang bertipe karakter
- Harus diakhiri dengan penekanan tombol enter
- Karakter yang dimasukkan terlihat pada layar
- Pergantian baris secara otomatis

getch() dan getche()

- Fungsi `getch()` dan `getche()` digunakan untuk membaca data karakter.
- Karakter yang dimasukkan tidak perlu diakhiri dengan penekanan tombol enter.
- Tidak memberikan efek pergantian baris secara otomatis
- Jika menggunakan fungsi `getch()` karakter yang dimasukkan tidak akan ditampilkan pada layar sehingga sering digunakan untuk meminta inputan berupa password.
- Sedangkan pada `getche()` karakter yang dimasukkan akan ditampilkan pada layar.

Contoh Program :

```
#include "stdio.h"  
#include "conio.h"  
void main() {  
    char huruf1, huruf2;  
    printf("Masukkan sebuah karakter : ");  
    huruf1 = getch(); // karakter yang dimasukkan akan terlihat di layar  
    printf("\nKarakter yang Anda masukkan adalah %c\n", huruf1);  
    printf("\nMasukkan sebuah karakter lagi : ");  
    huruf2 = getch(); // karakter yang dimasukkan tidak terlihat di layar  
    printf("\nKarakter yang Anda masukkan adalah : %c, huruf2);  
    getch();  
}
```

CATATAN :

Jika terdapat beberapa proses input (memasukkan data) sekaligus, maka sebaiknya ditambahkan fungsi `fflush(stdin)`; setelah fungsi `scanf()`. Fungsi `fflush(stdin)` berfungsi menghapus buffer di dalam alat I/O.

3.2. Menampilkan Data

Adalah menampilkan isi data didalam suatu variabel atau nilai tanpa penampung variabel.

Menampilkan data ke layar monitor

- Menggunakan fungsi `printf()`, `puts()`, dan `putchar()`.
- Fungsi `printf()` digunakan untuk menampilkan semua jenis data (numeric dan karakter)
- Fungsi `puts()` digunakan untuk menampilkan data string dan secara otomatis akan diakhiri dengan perpindahan baris.
- Fungsi `putchar()` digunakan untuk menampilkan sebuah karakter.

Mengatur tampilan bilangan pecahan (float)

Bentuk umum :

```
printf("%m.nf", argument);  
m : menyatakan panjang range  
n : menyatakan jumlah digit di belakang koma.  
argument : nilai atau variable yang akan ditampilkan.
```

Contoh :

```
printf("%5.2f", nilai);  
artinya variable nilai akan ditampilkan sebanyak 5 digit dengan 2 digit di belakang koma.
```

Contoh Program 1;

```
/* Program untuk menampilkan data berupa bilangan pecahan */  
#include <stdio.h>  
#include <conio.h>  
void main(){  
    float nilai;  
    clrscr();  
    puts("Masukkan nilai Anda :"); scanf("%f", &nilai);  
    printf("Anda memperoleh nilai %5.2f", nilai);  
    printf("Apakah Anda telah puas mendapat nilai %6.4f ?", nilai);  
    getch();  
}
```

Contoh Program 2;

```
/* Program untuk menampilkan data berupa bilangan integer dan string */  
#include <stdio.h>  
#include <conio.h>  
void main() {  
    int umur;  
    char nama[20];  
    clrscr();  
    puts("Masukkan nama Anda :"); gets(nama);  
    puts("Masukkan umur Anda :"); scanf("%d", &umur);  
    printf("Nama Anda : %s \n", nama); //tipe data string  
    printf("Umur Anda : %d \n", umur); //tipe data integer  
    getch();  
}
```

Menampilkan data ke printer

- Untuk menampilkan data ke printer dapat menggunakan fungsi `fprintf()`, `fputs()` dan `fputc()`.
- Fungsi `fprintf()` digunakan untuk mencetak semua jenis tipe data ke printer dan secara otomatis memberikan efek perpindahan baris.
- Fungsi `fputs()` digunakan untuk mencetak tipe data string ke printer
- Fungsi `fputc()` digunakan untuk mencetak tipe data karakter ke printer

Contoh program :

```
#include <stdio.h>
#include <conio.h>
void main() {
    fprintf(stdprn, "Hallo, Saya akan tercetak di printer");
    fputs(stdprn, "Saya juga akan tercetak di printer");
}
```