

FUNGSI

OBJEKTIF :

1. Mahasiswa Mampu Memahami Tentang Fungsi pada Bahasa C.
 2. Mahasiswa Mampu Memahami Tentang Fungsi Argument (Local Global Variabel).
-

1.1 FUNGSI

Function adalah sub-program yang bisa digunakan kembali baik di dalam program itu sendiri, maupun di program yang lain. Terdapat beberapa keuntungan jika menggunakan Function, diantaranya :

1. Modularity artinya kita dapat mendefinisikan functions yang berbeda, dan kita dapat memanggil functions tertentu jika kita membutuhkannya
2. Reusable artinya dengan menggunakan function akan menghemat ruang yang banyak dan akan lebih mudah untuk dibaca.
3. Functions mudah untuk diimplementasikan pada program

Function terbagi menjadi dua jenis, yaitu Built-in functions dan User defined functions.

1. Built-in functions : Sistem menyediakan function ini dan menyimpannya dalam library jadi bisa disebut juga sebagai library functions. contohnya `scanf()`, `printf()`, `strcpy()` dan masih banyak lagi. untuk menggunakannya hanya perlu digunakan include pada header yang sesuai.
2. User defined functions : Adalah functions yang didefinisikan oleh user pada saat menuliskan program

Sintaks untuk mendeklarasikan function adalah sebagai berikut :

`tipedata namafungsi(parameter/variabel yang digunakan)`

Contoh :

```
int tambah()
```

Sintaks untuk mendefinisikan function adalah sebagai berikut :

```
returntipe_namafungsi(function argumen)
{
    statement pada function
}
```

Contoh :

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

int main()
{
    int jawaban;
    jawaban = tambah();
    printf("Hasil penjumlahan adalah %d", jawaban);
    return 0;
}

int tambah()
{
    int num1=10;
    int num2=5;

    return num1+num2;
}
```

1.2 FUNGSI ARGUMEN (LOCAL GLOBAL VARIABEL)

Terdapat dua cara untuk memanggil function argument atau parameter, yaitu dengan dipanggil berdasarkan value dan dipanggil berdasarkan reference.

1. Dipanggil Berdasarkan Value

Jika dipanggil berdasarkan value maka parameter aktual melewati function. Kemudian area memori baru dibuat untuk parameter tertentu dan hanya dapat digunakan di dalam function itu. Selain itu Parameter aktual tidak dapat diubah.

2. Dipanggil Berdasarkan Reference

Jika dipanggil berdasarkan reference maka akan function akan menyalin variabel alamat, lalu diteruskan ke function sebagai parameter. Kemudian menggunakan operator alamat yang merupakan simbol (&). Operator alamat digunakan dalam parameter function yang dipanggil dan tentu saja perubahan function mencerminkan perubahan nilai aslinya.

Contoh :

```
//pemanggilan berdasarkan value
#include<stdio.h>
#include<stdlib.h>

//deklarasi function
int tambah(int num1, int num2);

int main()
{
    int jawaban;
    int num1=10;
    int num2=5;
    jawaban = tambah(num1, num2);
```

```
        printf("Hasil penjumlahan adalah %d", jawaban);
        return 0;
    }

    //definisi function
    int tambah(int a, int b)
    {
        return a+b;
    }
```

Contoh :

```
//panggilan berdasarkan reference
#include<stdio.h>
#include<stdlib.h>
//deklarasi function
int tambah(int *num1, int *num2);
int main()
{
    int jawaban;
    int num1=10;
    int num2=5;
    jawaban = tambah(&num1, &num2);
    printf("Hasil penjumlahan adalah %d", jawaban);
    printf("\nhasil dari nilai a adalah %d", num1);
    return 0;
}

//definisi function
int tambah(int *a, int *b)
{
    *a=6;
    return *a+*b; }
```

RANGKUMAN

1. Function adalah sub-program yang bisa digunakan kembali baik di dalam program itu sendiri, maupun di program yang lain.
2. Terdapat dua cara untuk memanggil function argument atau parameter, yaitu dengan dipanggil berdasarkan value dan dipanggil berdasarkan reference.

TUNTUNAN LATIHAN

Berikut ini diberikan contoh program untuk memutlakkan suatu nilai :

- (a) Mendeklarasikan pre-processor include diikuti dengan file judul atau header file. Caranya mengetikkan langsung pada lembar kerja, yaitu :

```
1 #include<stdio.h>
2
```

- (b) Mendeklarasikan fungsi absolut dengan tipe data double. Caranya mengetikkan langsung pada lembar kerja, yaitu :

```
3 double absolut();
4
```

- (c) Mendeklarasikan fungsi main. Caranya mengetikkan langsung pada lembar kerja, yaitu :

```
5 main()
6 {
```

- (d) Mendeklarasikan variabel nilai dengan tipe data float dan memberi nilai pada variabel tersebut. Caranya mengetikkan langsung pada lembar kerja, yaitu :

```
7 float nilai;
8 nilai = -123.45;
```

- (e) Membuat perintah untuk mencetak output diikuti dengan pemanggilan fungsi absolut untuk variabel nilai. Caranya mengetikkan langsung pada lembar kerja, yaitu :

```
9 printf("%7.2f nilai mutlaknya adalah %7.2f\n", nilai,
10 absolut(nilai));
11
12 }
13
```

- (f) Membuat fungsi absolut dengan tipe data double. Caranya mengetikkan langsung pada lembar kerja, yaitu :

```
14 //fungsi untuk memutlakkan suatu nilai negatif
15 double absolut(double x) //definisi fungsi
16 {
17     if(x<0) x=-x;
18     return(x);
19 }
20
```

LATIHAN

1. Buatlah program fungsi untuk menghitung factorial berdasarkan nilai yang diinputkan !