

MODUL 5**FUNGSI DAN PROSEDUR****A. TUJUAN**

Setelah menyelesaikan modul ini diharapkan mahasiswa akan mampu:

1. Mahasiswa mampu memahami fungsi dan prosedur.
2. Mahasiswa mampu mendeklarasikan dan mendefinisikan fungsi dan prosedur.
3. Mahasiswa mampu menyelesaikan masalah menggunakan fungsi dan procedure
4. Mahasiswa mampu memahami metode fungsi dan prosedur
5. Mahasiswa mampu memahami dan membuat rekursif

B. PETUNJUK

1. Awali setiap aktivitas anda dengan doa, agar anda lancar dalam belajar
2. Kerjakan tugas-tugas praktikum dengan baik, jujur, dan sabar
3. Tanyakan kepada instruktur apabila ada hal-hal yang kurang jelas

C. DASAR TEORI**1. Pengertian Fungsi dan Prosedur**

Prosedur/Fungsi adalah suatu program terpisah dalam blok sendiri yang berfungsi sebagai subprogram (program bagian). Tujuan pembuatan fungsi atau prosedur adalah

1. Program menjadi terstruktur
2. Dapat mengurangi duplikasi kode
3. Fungsi/procedure dapat di panggil berulang kali sesuai dengan kebutuhan

Pada umumnya fungsi atau procedure memerlukan masukan yang di sebut parameter atau argument. Parameter atau argument dapat lebih dari satu atau tidak sama sekali. Untuk memberi nama pada fungsi atau procedure aturannya sama dengan membuat variable. Bentuk umum penulisannya adalah

```
typeDataFungsi namaFungsi(parameter){
    //isi fungsi
}
```

2. Perbedaan fungsi dan procedure

Prosedur :

Prosedur dalam Bahasa C++ diawali dengan tipe data "void" yang artinya kosong. Artinya procedure tidak memiliki tipe data dan tidak memiliki nilai kembalian

Contoh prosedur menampilkan angka 1 sampai batas yang di tentukan :

```
void luaspersegi panjang(int batas){
    for(int i = 1; i < batas; i++) { cout << i << endl; }
}
```

Fungsi :

Fungsi dalam Bahasa C++ diawali dengan sebuah tipe data seperti int, Boolean, float, dll dan fungsi memiliki sebuah nilai kembalian yang di tandai dengan “return”.

Contoh fungsi menghitung luas persegi panjang :

```
int luaspersegi(int panjang, int lebar){
    int hasil = panjang * lebar;
    return hasil;
}
```

3. Prototype fungsi dan prosedur

Sebuah program C++ dapat terdiri dari banyak fungsi. Salah satu fungsi tersebut bernama main(). Pada umumnya fungsi di letakkan sebelum main. Jika fungsi yang lain di tuliskan setelah fungsi main(), sebelum fungsi main harus ditambahkan prototype function agar fungsi tersebut dapat dikenal. Hal tersebut tidak berlaku jika penulisannya dilakukan sebelum fungsi main().

```
#include <iostream>
using namespace std;

//prototype fungsi
void hitung(int angka);
int luas(int p, int l);

int main(){
    //isi fungsi main
}

//inisialisasi prosedur hitung
void hitung(int angka){
    cout << angka * 3;
}

//inisialisasi fungsi luas
int luas(int p, int l){
    int hasil = p * l;
    return hasil;
}
```

4. Fungsi rekursif

Rekursif adalah memanggil dirinya sendiri, artinya fungsi tersebut di panggil pada bagian tubuh fungsi atau isi fungsi. Hampir sama seperti perulangan, fungsi rekursif juga terdapat batasan kapan rekursif tersebut akan berhenti. Jika tidak maka rekursif tersebut tidak akan berhenti(infinitie loop).

Contoh fungsi mencari factorial n

```
Int faktorial(int n){
    //batasan / bestcase
    If(n == 1){
        return 1;
    }

    return n*faktorial(n+1);
}
```

D. LATIHAN

1. Latihan 1

```
1  #include <iostream>
2  using namespace std;
3
4  void membuatsegitiga(int tinggi){
5
6      for(int i = 0; i < tinggi; i++){
7          int h = 2 * i + 1;
8          for(int j = 0; j < i; j++) { cout << " "; }
9          for(int j = h; j < tinggi*2; j++) { cout << "*"; }
10         cout << endl;
11     }
12 }
13
14 int main(){
15     int n = 3;
16     cout << "membuat segitiga terbalik" << endl;
17     cout << "masukkan tinggi segitiga : " << endl;
18     cin >> n;
19     cout << endl;
20
21     membuatsegitiga(n); // memanggil prosedur membuatsegitiga
22
23 }
```

2. Latihan 2

```
1  #include <iostream>
2  using namespace std;
3
4  //variable global
5  const double pi = 3.14;
6  double volume;
7
8  double volume_tabung(int r, int tinggi){
9      volume = pi * r * r * tinggi;
10     return volume;
11 }
12
13 int main() {
14     double t, r;
15     cout << "Menghitung volume tabung" << endl;
16     cout << "masukkan jari jari tabung : ";
17     cin >> t; cout << endl;
18     cout << "masukkan tinggi tabung : ";
19     cin >> r; cout << endl;
20
21     cout << "volume tabung = " << volume_tabung(t,r) << endl;
22 }
```

3. Latihan 3

```
1  #include <iostream>
2  using namespace std;
3
4  long fibo(long angka){
5      if(angka == 1 || angka == 2) { return 1; }
6      return fibo(angka-1) + fibo(angka-2);
7  }
8
9  int main() {
10     long a;
11
12     cout << "menghitung bilangan fibonanci" << endl;
13     cout << "masukan angka : ";
14     cin >> a; cout << endl;
15
16     cout << "fibo " << a << " = " << fibo(a);
17 }
```

E. TUGAS PRAKTIKUM

1. Terdapat 3 inputan yaitu sisi 1, sisi 2, sisi 3. Tentukan dari ketiga sisi tersebut dapat terbentuk segitiga atau tidak. Jika tidak bisa keluarkan tidak bisa, jika bisa, tentukan apakah segitiga siku – siku, segitiga sembarang, atau segitiga sama kaki.
2. Buatlah fungsi untuk mencari hasil pangkat dari suatu bilangan dengan menggunakan rekursif