

MODUL 7

FUNGSI

A. TUJUAN

Setelah mempelajari bab ini diharapkan mahasiswa akan mampu:

- Memahami fungsi
- Mendeklarasikan dan mendefinisikan fungsi
- Menyelesaikan masalah menggunakan fungsi
- Memahami metode pemanggilan fungsi
- Membuat fungsi rekursif

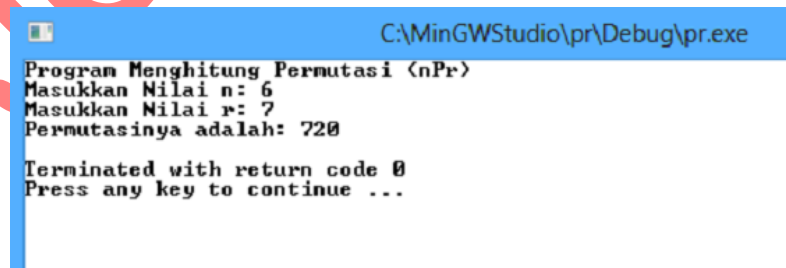
B. PETUNJUK

1. Awali setiap aktivitas anda dengan doa, agar anda lancer dalam belajar.
2. Pahami tujuan, dasar teori dan latihan- latihan praktikum dengan baik.
3. Kerjakan tugas-tugas praktikum dengan baik, jujur dan sabar.
4. Tanyakan kepada asisten praktikum apabila ada hal-hala yang kurang jelas.

C. ALAT dan BAHAN

- PC/Laptop yang memenuhi kapasitas untuk pemrograman Bahasa C++
- Software C++
- Modul/handout materi yang akan di praktikkan.

D. DASAR TEORI



```
C:\MinGWStudio\pr\Debug\pr.exe
Program Menghitung Permutasi (nPr)
Masukkan Nilai n: 6
Masukkan Nilai r: 7
Permutasinya adalah: 720
Terminated with return code 0
Press any key to continue ...
```

Dalam pembuatan program diatas terdapat suatu rumus permutasi. Agar program lebih efisien dipisahkan antara rumus dengan program utama, sehingga dalam program uatam hanya ada perintah memanggil rumus untuk memproses suatu variable. Hal ini dalam pemrograman disebut sebagai fungsi.Fungsi merupakan blok dari kode yang dirancang untuk melakukan tugas khusus. Tujuan pembuatan fungsi adalah:

- o Program menjadi terstruktur
- o Dapat mengurangi duplikasi kode
- o Fungsi dapat di panggil dari program atau fungsi yang lain

Pada umumnya fungsi memerlukan masukan yang disebut parameter atau argument. Hasil akhir fungsi akan berupa nilai(nilai balik fungsi). Bentuk umum sebuah fungsi adalah:

JOB SHEET TEUM

TypeFungsi namaFungsi (parameter)

```
{
//Tubuh fungsi
}
```

Misalkan: Dalam bahasa C++

```
void Ckali(int angka)
{
    int a=angka*angka;
    cout<<"Hasil Perkalian:"<<a<<endl;
}
```

Dalam bahasa Java:

```
public void Jkali(int angka)
{
    int a=angka*angka; system.out.println("Hasil Perkalian:" + a);
}
```

Dalam bahasa VB:

```
public Vkali(ByVal angka As Integer) As Integer
{
    Dim a = angka*angka Console.WriteLine("Hasil Perkalian:" & a)
}
```

TypeFungsi berfungsi untuk menentukan tipe keluaran fungsi yang dapat berupa salah satu tipe data C++ yang berlaku misalnya char atau int. Default tipe yang tidak disebutkan dianggap sebagai int.

Contoh:

```
Ckali()
{
    return(0);
}
```

PROTOTYPE FUNCTION

Sebuah program C++ dapat terdiri dari banyak fungsi. Salah satu fungsi tersebut harus bernama 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().

Misal:

```
#include <stdio.h>
//prototype function
void hitung(int angka, int bilangan);
void tulis(char);
int tampil(int angka[], char huruf);
//fungsi main
void main()
{
    int array[3]={1,2,3};
    char huruf='D';
    //memanggil fungsi
    hitung(2,3);
    tulis('A');
    tampil(array,huruf);
}
//inisialisasi fungsi hitung
void hitung(int a, int b)
{
    //Blok fungsi
    printf("Angka adalah: %d\n",a);
    printf("Angka adalah: %d\n",b);
}
//inisialisasi fungsi tulis
void tulis(char a)
{
    //blok fungsi
    printf("huruf adalah: %c\n",a);
}
//inisialisasi fungsi tampil
int tampil(int x[], char a) {
    //blok fungsi
    return(a);
}
```

Call by Value vs Call by Reference

Ada dua cara pemanggilan fungsi yaitu: (1) Call by Value (2) Call by Reference

(1) Call By Value

Pemanggilan fungsi dengan melewati nilai parameter variable dalam definisi fungsi.

Fungsi tersebut akan mengolah nilai yang dikirimkan kepadanya. Misalnya:

```
float purchase = 4000;
float tax = calcSalesTax(purchase);
//dapat juga merupakan suatu konstan
hitung(2, 3);
tulis(„A”);
```

(2) Call by Reference

Pemanggilan dengan referensi merupakan suatu usaha melewatkan alamat dari suatu variable ke dalam fungsi. Dengan perubahan dilakukan di dalam fungsi untuk mengubah variabel di luar fungsi. Misal:

A bernilai 20 □ X juga bernilai 20

B bernilai 30 □ Y juga bernilai 30

Selanjutnya nilai X dan Y diproses tetapi nilai A dan B akan tetap (tidak berubah). Metode Call By Reference menggunakan operator pointer „&” dan „*” untuk mengirimkan alamat variable ke suatu fungsi.

Fungsi Rekursif

Fungsi rekursif adalah fungsi yang memanggil dirinya sendiri artinya fungsi tersebut dipanggil di dalam tubuh fungsi itu sendiri. Parameter yang dilewatkan berubah sebanyak fungsi itu dipanggil.

E. LATIHAN

1. LAT1

Menjelaskan maksud program berikut:

```

#include <iostream.h>
void Rtukar (int *,int*);
void Vtukar (int,int);
main()
{
    int a,b;
    a=99; b=11;
    //call by reference
    cout<<"Nilai sebelum pemanggilan fungsi "<<endl;
    cout<<"Nilai a= " <<a<<" Nilai b= "<<b<<endl;
    Rtukar(&a,&b);
    cout<<"Nilai setelah pemanggilan fungsi Rtukar: "<<endl;
    cout <<"Nilai a="<<a<<" Nilai b="<<b<<endl<<endl<<endl;
    //kembalikan ke nilai semula
    a=99;
    b=11;
    //call by value
    cout<<"\nNilai sebelum pemanggilan fungsi : "<<endl;
    cout<<"Nilai a="<<a<<" Nilai b="<<b<<endl<<endl;
    Vtukar(a,b);
    cout<<"Nilai setelah pemanggilan fungsi Vtukar: "<<endl;
    cout<<"Nilai a=" <<a<<" Nilai b=" <<b<<endl;
    return 0;
}

void Rtukar (int *px,int *py)
{
    int temp;
    temp = *px;
    *px=*py;
    *py=temp;
    cout<<"Nilai pada fungsi dengan Call by Value: "<<endl;
    cout<<"Nilai px="<<*px<<" Nilai py="<<*py<<endl<<endl;
}

```

```

void Vtukar (int px,int py)
{
    int temp;

    temp = px;

    px=py;

    py=temp;

    cout<<"Nilai pada fungsi dengan Call by Value: "<<endl;

    cout<<" Nilai px="<<px<<" Nilai py="<<py<<endl<<endl;

}

```

2. LAT2

Mencari nilai terbesar dari suatu data

```

//fungsi
#include<stdio.h>
//deklarasi fungsi
int cari(int n1, int n2);
void prin (int n);

//main function
main()
{
    int i=5;
    int j=7;
    int k;
    k = cari(i,j);
    prin(k);
    return 0;
}
//<-- function definition here

int cari(int a, int b)
{
    if (a>b) return a;
    else return b;
}

void prin (int n)
{
    printf("Bilangan terbesar adalah: %d\n",n);
}

//<-- end function

```

3. LAT3

Menentukan pajak pembelian

```

#include <iostream.h>
//deklarasi fungsi
int calcSalesTax(float purchase, float taxRate=0.1);
//main function
main()
{
    float purchase;
    cout<<"Masukkan pembelian : ";
    cin>>purchase;

    float tax=calcSalesTax(purchase);
    cout<<"Pajak peabalian adalah : "<<tax<<endl;
    return 0;
}
int calcSalesTax(float purchase, float taxRate)
{
    return purchase * taxRate;
}

```

4. LAT4

Membedakan program call by reference dan call by value

```

/*aplikasi rekursif pada faktorial*/
#include <iostream.h>
int faktorial(int a)
{
    if(a==1)
        return 1;
    else
        return a*faktorial(a+1);
}

void main()
{
    int x;
    cout<<"Mencari Nilai Faktorial\n";
    cout<<"Masukkan nilai n=";
    cin>>x;
    cout<<"Faktorial "<<x<<"!="<<faktorial(x)<<endl;
}

```

5. LAT5

Program menghitung factorial dengan fungsi rekursif


```

/*aplikasi rekursif pada fibonacci*/
#include <iostream.h>

long fibo(long a)
{
    if(a==1 && a==2)
        return 1;
    else
        return fibo(a-1) + fibo(a-2);
}

void main()
{
    int x;
    cout<<"Mencari Nilai Fibonacci\n";
    cout<<"Masukkan nilai n=";
    cin>>x;
    cout<<"Fibonacci " <<x<<"=" <<fibo(x)<<endl;
}

```

F. TUGAS PRAKTIKUM

1. TGP1

Program fungsi untuk menampilkan permutasi nPr.

```

C:\MinGWStudio\permutasi\Debu
Program Menghitung Permutasi (nPr)
-----Ananda Putri Syaviri-----
-----130533608243-----
-----PTI'13 B-----
Masukkan Nilai n: 4
Masukkan Nilai r: 5
Permutasinya adalah: 24

Terminated with return code 0
Press any key to continue ...

```

2. TGP2

Program fungsi untuk menghitung nilai fibonaci dan factorial dari sebuah inputan yang dimasukkan

```

C:\MinGWStudio\fact_fibo\Deb
Program Menghitung Faktorial dan Fibonacci
-----Ananda Putri Syaviri-----
-----130533608243-----
-----PTI'13 B-----
Masukkan Nilai: 10
Hasil 10 Faktorial adalah 3628800
Fibonacci 10 = 55

Terminated with return code 0
Press any key to continue ...

```

G. TUGAS RUMAH

1. Buat Program untuk menentukan jumlah gaji bersih dari gaji pokok yang diinputkan ditambah dengan jumlah bonus juga diinputkan. Untuk mencari bonus rumusnya adalah jumlah bonus yang dimasukkan dikalikan dengan gaji pokok. Gaji bersih = gaji pokok + jumlah bonus. Petunjuk:

Gunakan 5 macam fungsi:

main()

InputData()

bonus()

gajibersih()

gajitotal()

2. Buatlah program bentuk MENU UTAMA yang berisi operasi aritmetika yang berisi penjumlahan, pengurangan, perkalian dan pembagian. MENU UTAMA tersebut berisi 5 pilihan yang tampil seperti berikut:

MENU UTAMA

1. Operasi Penjumlahan

2. Operasi Pengurangan

3. Operasi Perkalian

4. Operasi Pembagian

x. Keluar

Menu di atas akan diulang terus menerus dengan meminta pengguna supaya memasukkan kode. Jika kode yang di inputkan oleh pengguna selain 1,2,3,4 atau x, satu pesan muncul "Kode tidak sesuai!", sedangkan kode X pula akan menghasilkan pesan "Terima Kasih".