**TUGAS PRAKTIKUM 15**

**DASAR FUNGSI**



**Nama : Devafilla Rizqy Santosa**

**Kelas : 1 D4 Teknik Informatika A**

**NRP : 3120600026**

1. Tugas Pendahuluan.
	1. A. Listing

#include <stdio.h>

#include <stdlib.h>

int terbesar(int,int);

main()

{

int bil\_1, bil\_2, maks;

 printf("Input bilangan pertama = ");

 scanf("%d", &bil\_1);

 printf("Input bilangan kedua = ");

 scanf("%d", &bil\_2);

 maks = terbesar(bil\_1, bil\_2);

 printf("\nBil %d adalah bil terbesar", maks);

}

int terbesar(int input\_1, int input\_2)

{

 if(input\_1 < input\_2)

 return input\_2;

 else

 return input\_1;

}



1. Output



1. Analisa

Sebuah program yang dapat menentukan 2 bilangan terbesar dari inputan dengan prototype. Dengan menggunakan fungsi tukar() sebagai deklarasi fungsi lalu menggunakan return value untuk dapat menampilkan bilangan terbesar.

* 1. A. Listing

#include <stdio.h>

#include <stdlib.h>

int permu (int,int);

int komb (int, int);

int faktor (int);

main()

{

int n, r, rumus\_perm, rumus\_komb;

 printf("\nMasukkan nilai n \t = ");

 scanf("%d", &n);

 printf("\nMasukkan nilai r \t = ");

 scanf("%d", &r);

 rumus\_perm = permu (n, r);

 rumus\_komb = komb (n, r);

 printf("\nHasil permutasi antara %d dengan %d adalah%d", n, r, rumus\_perm);

 printf("\nHasil kombinasi antara %d dengan %d adalah%d", n, r, rumus\_komb);

}

int faktor (int bil)

{

 int i, m = 1;

 for(i=m; i>0; i--)

 m = m \* i;

 return m;

}

 int permu (int nilai\_n, int nilai\_r)

{

 return faktor(nilai\_n) / faktor(nilai\_n - nilai\_r);

}

 int komb(int nilai\_n, int nilai\_r)

{

 return faktor(nilai\_n) / ( faktor(nilai\_r) \* faktor(nilai\_n - nilai\_r) );

}



1. Output



1. Analisa

Sebuah program yang dapat menghitung permutasi dan kombinasi. Menggunakan 5 deklarasi fungsi untuk menghitung setiap factorial dan nanti di fungsi terakhir digunakan untuk menghitung permutasi dan kombinasi.

* 1. A. Listing

#include <stdio.h>

#include <stdlib.h>

void naik(int \*n)

{

 (\*n)+=2;

 printf("Nilai fungsi = %d", \*n);

}

int main(void)

{

 int bil;

 printf("Input bil bulat: ");

 scanf("%d",&bil);

 puts("menampilkan hasil awal");

 printf("Nilai awal = %d\t",bil);

 puts("memanggil fungsi naik");

 naik(&bil);

 puts("menampilkan hasil akhir");

 printf("Nilai akhir = %d\t",bil);

 return 0;

}



B. Output



C.Analisa

Sebuah program yang menambahkan 2 angka dari 2 bilangan input dengan menggunakan pass by reference. Deklarasi fungsi menggunakan naik(int \*a, int \*b) lalu dalam fungsi setelah inputan digunakan naik(&a,&b) untuk dapat menyimpannya di pointer ( \* ) . Di pendeklarasi fungsi setelah \*a+=2 artinya nilai sudah di tambah 2 dan alamatnya sama dengan a yang di main, maka nantinya dapat langsung ditampilkan di main tanpa menggunakan return.

* 1. A. Listing

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

#define phi 3.142857f;

float BC (float,float);

float radian(float);

main()

{

 float sudut\_BAC,sisi\_AB,sisi\_BC;

 printf("\nMasukkan besaran sudut BAC\t = ");

 scanf("%f", &sudut\_BAC);

 printf("\nMasukkan panjang sisi AB\t = ");

 scanf("%f", &sisi\_AB);

 sisi\_BC = BC(sudut\_BAC,sisi\_AB);

 printf("\nPanjang sisi BC adalah %.2f meter", sisi\_BC);

}

 float BC(float nilai\_sudut\_BAC,float nilai\_sisi\_AB)

{

 return tan (radian(nilai\_sudut\_BAC)) \* nilai\_sisi\_AB;

}

 float radian(float derajat)

{

 return (derajat / 180.0f) \* phi;

}



B. Output



C.Analisa

Sebuah program yang dapat menghitung Panjang dari sebuah segitiga siku siku dengan diketahui ABC siku di B dan diketahui AB dan sudut di titik A menghitung panjang BC. Untuk deklarasi fungsi digunakan float rad( float , float) , dengan menggunakan tan = depan / samping, disini menggunakan radian untuk nantinya bisa dijadikan nilai tan. Lalu bc = tan(rad) \* ab untuk menentukan panjangnya , dan hasil akan menggunakan return by value

1. Pass By Value dan Pass By Reference merupakan metode dalam mengirimkan parameter ke dalam sebuah fungsi, dalam java disebut method . Pass By Value melewatkan parameter berdasarkan nilainya.
2. Passing by value adalah teknik bagaimana melewatkan suatu argument tanpa melakukan perubahan nilai setelah argument dikembalikan. Passing by refrence adalah teknik melewatkan argument dalam bentuk alamat suatu nilai. Jika terjadi perubahan nilai argumen, maka perubahan dibawa sampai argumen selesai dilewatkan.