STRUKTUR

SRI SUPATMI,S.KOM

>> STRUKTUR

- Struktur dalam bahasa C sama dengan record dalam bahasa Pascal.
- Struktur adalah kumpulan elemen – elemen data yang digabung menjadi satu kesatuan.
- Masing – masing elemen data pada record disebut filed, dimana field pada struktur dapat memiliki tipe data yang berbeda.
- Deklarasi Struktur :

```
struct nama_stuktur {
        tipe_data1 field1;
        tipe_data2 field2;
        . . .
        tipe_datan fieldn;
      } nama_variabel;
```

Hal yang diperhatikan pada struktur diatas :

- Deklarasi struktur hanya mendefinisikan bentuk data, tidak mendeklarasikan variable. Untuk mendeklarasikannya maka harus menambahkan nama variable strukturnya.
- Bagian dari struktur (field) adalah sebuah statement, sehingga harus diakhiri dengan titik koma (;).
- Contoh Deklarasi Struktur :

```
struct data_mhs {
        char nama[20];
        char nim[20];
        float ipk;
       } mahasiswa;
```

>> CONTOH PROGRAM

```
#include<stdio.h>
#include<conio.h>
void main()
// menghitung luas segitiga
{
struct  {
      float alas,tinggi;
      float luas;
      } segitiga ;
 clrscr();
puts(">>>> program menghitung luas segitiga <<<<");
printf("input alas :");
scanf("%f",&segitiga.alas);
printf("input tinggi : ");
scanf("%f",&segitiga.tinggi);

segitiga.luas=(segitiga.alas*segitiga.tinggi)/2;

printf("Luas segitiga = %2.2f\n",segitiga.luas);
getch();
}
```

>> contoh 2

```
#include<stdio.h>
#include<conio.h>
main()
{
// meninputkan data mahasiswa
struct mhs {
                        char nama[20];
                        char nim[20];
                        float ipk;
                        }; mhs mahasiswa;
clrscr();
printf("nama : "); gets(mahasiswa.nama);
printf("nim  : "); scanf("%s",&mahasiswa.nim);
printf("ipk  : "); scanf("%f",&mahasiswa.ipk);

printf("data mahasiswa \n");
printf("nama : %s \n",mahasiswa.nama);
printf("nim  : %s \n",mahasiswa.nim);
printf("ipk  : %f \n",mahasiswa.ipk);
}
```

>> penerapan array dan struktur

- Penggunaan struktur sering dikaitkan dengan array yang membentuk suatu data yang terstruktur.

BENTUK UMUM	CONTOH DEKLARASI
struct nama_stuktur { tipe_data1 field1; tipe_data2 field2; . . . tipe_datan fieldn; } ; struct <nama_struct> <nama_variabel>[n_array]	**struct** tanggal_lahir{ int tanggal; int bulan; int tahun; } ; Struct tanggal_lahir date[10]

>> contoh program

```
#include<stdio.h>
#include<conio.h>
main(){
struct mhs {                    char nama[20];
                                char nim[20];
                                float ipk;
                                }; mhs mahasiswa;
int jml, i, total=0;
clrscr();
printf("jumlah mahasiswa : "); scanf("%d", &jml);
for(i=1; i<=jml; i++) {
printf("nama : "); scanf("%s",&mahasiswa.nama);
printf("nim  : "); scanf("%s",&mahasiswa.nim);
printf("ipk  : "); scanf("%f",&mahasiswa.ipk);
if(mahasiswa.ipk <= 2)
                { total++;                      }
     }
printf("data mahasiswa \n");
for(i=1; i<=jml; i++)
     {
printf("nama : %s \n",mahasiswa.nama);
printf("nim  : %s \n",mahasiswa.nim);
printf("ipk  : %.2f \n",mahasiswa.ipk);
     }
printf("jumlah mahasiswa ipk <=2 adalah : %d",total);
}
```