

ARRAY

Array adalah sekelompok data sejenis yang disimpan ke dalam variabel dengan nama yang sama, dengan memberi indeks pada variabel untuk membedakan antara yang satu dengan yang lain.

VARIABEL ARRAY

nama_variabel[indeks]

ketentuan nama variabel array sama dengan nama variabel biasa.
indeks menunjukkan nomor dari variabel .

DEKLARASI VARIABEL ARRAY

BU : tipe nama_variabel[indeks];

Contoh : float bil[10];

deklarasi variabel array dengan nama bil yang akan menampung 10 data yang bertipe float. Indeks 10 menunjukkan variabel bil terdiri dari 10 elemen, dimana setiap elemen akan menampung sebuah data.

Indeks array dimulai dari nol(0) , sedang nomor elemen biasanya dimulai dari satu(1). Nomor elemen dapat dibuat sama dengan nomor indeks untuk mempermudah pembuatan program yaitu dengan memberi indeks satu lebih banyak dari jumlah data yang dibutuhkan, sehingga menjadi :

float bil[11]

INISIALISASI ARRAY 1 DIMENSI

Inisialisasi dapat dilakukan bersama dengan deklarasi atau tersendiri. Inisialisasi suatu array adalah dengan meletakkan elemen array di antara tanda kurung kurawal {}, antara elemen yang satu dengan lainnya dipisahkan koma.

int bil[2] = {4,1,8}

bil[0] = 4

bil[1] = 1

bil[2] = 8

AUTOMATIC ARRAY adalah Inisialisasi array dilakukan di dalam fungsi tertentu. Hanya compiler C yang berstandar ANSI C yang dapat menginisialisasikan automatic array.

Cara menginisialisasikan array dari compiler yg tidak mengikuti standar ANSI C:

1. Diinisialisasikan di luar fungsi sebagai variabel GLOBAL/EXTERNAL ARRAY.

int bil[2]={0,0,0};

main()

2. Diinisialisasikan didlm fungsi sebagai variabel LOKAL/STATIC ARRAY.

```

main()
{
    static int bil[2]={0,0,0};
    .....

```

Pada automatic array yang tidak diinisialisasikan , elemen array akan memiliki nilai yang tidak beraturan. Bila global & static array tidak diinisialisasi maka semua elemen array secara otomatis akan diberi nilai nol(0).

Contoh :

```

main()
{
    int y;
    int hitung=0;
    int x[0];
    for(y=0;y<5;y++)
    {
        hitung+=y;
        x[y]=hitung;
        printf("%3d - %3d\n",y,x[y]);
    }
}

```

OUTPUT:

```

0- 0
1- 1
2- 3
3- 6
4- 10

```

MENDEFINISIKAN JUMLAH ELEMEN ARRAY DALAM VARIABEL

Besarnya variabel indeks dapat ditentukan dengan menggunakan preprocessor directives #define

```

#define N 40
main()
{
    int no[N],gaji[N],gol[N],status[N],juman[N];

```

Bila besari indeks akan diubah menjadi 50, cukup diganti dengan #define N 50

ARRAY 2 DIMENSI

```

nama_variabel [indeks1][indeks2]

```

indeks1 : jumlah/nomor baris
indeks2 : jumlah/nomor kolom

Jumlah elemen yang dimiliki array 2 dimensi dapat ditentukan dari hasil perkalian indeks1 * indeks2

misal : array A[2][3] akan memiliki $2 \times 3 = 6$ elemen.

```
main()
{
    float bil [5] [5]
    .....
```

dapat dituliskan dengan #define

```
#define N 5
main()
{
    float bil [N] [N]
    .....
```

INISIALISASI ARRAY 2 DIMENSI

```
main()
{
    float bil[2] [3] =
    { { 1,2,3}, /*baris 0*/
      { 4,5,6}, /*baris 1*/
    }
}
```

```
elemen bil [0] [0] = 1
elemen bil [0] [1] = 2
elemen bil [0] [2] = 3
elemen bil [1] [0] = 4
elemen bil [1] [1] = 5
elemen bil [1] [2] = 6
```

Contoh :

```
main()
{
    int x[3][5];
    int y,z;
    int hitung=0;
    for(y=0;y<3;y++)

    {
        printf("y = %d\n",y);
        for(z=0;z<5;z++)
        {
            hitung+=z;
        }
    }
}
```

```

        x[y][z] = hitung;
        printf("%t/t%3d - %3d\n",z,x[y][z]);
    }
}

```

OUTPUT:

```

y = 0
0- 0
1- 1
2- 2
3- 6
4- 10
y = 1
0- 10
1- 11
2- 13
3- 16
4- 20
y = 2
0- 20
1- 21
2- 23
3- 26
4- 30

```

STRING dan ARRAY

1. Pada string terdapat karakter null(\0) di akhir string
2. String sudah pasti array, array belum tentu string

CONTOH - CONTOH :

1. array dengan pengisian input melalui keyboard

```

    baca_input()
    {
        float nilai[10];
        for(i=0;i<10;i++)
            scanf("%d",&nilai[i]);
    }

```

2. Fungsi yang mencetak isi array dari akhir ke awal

```

    cetak_array()
    {
        float nilai[10];
        for(i=9;i>=0;i--)
            scanf("%3f",nilai[i]);
    }

```

3. Menghitung rata - rata isi array nilai

```
rata_rata()
{
    float nilai[10],jum*rata;
    for(i=0,jum=0;i<=9;i++)
        jum+=nilai[i];
    rata=jum/i;
}
```

4. Mencari nilai terbesar

```
besar()
float temp,nilai[10];
{
    for(temp=nilai[0],i=1;i<=9;i++)
        if(nilai[i] > temp)
            temp=nilai[i];
}
return(temp)
```