

Modul I

ARRAY

Pertemuan kali ini kita akan kembali membahas materi yang sudah diberikan di algoritma dan pemrograman, yaitu array. Secara singkat, array adalah suatu tipe data terstruktur yang berupa sejumlah data sejenis (bertipe data sama) yang jumlahnya tetap dan diberi suatu nama tertentu.

Array dapat berupa array 1 dimensi, 2 dimensi, bahkan n-dimensi.

DEKLARASI

```
tipe_data nama_var_array [ukuran];
```

tipe_data	:	menyatakan jenis tipe data elemen larik (int, char, float, dll)
nama_var_array	:	menyatakan nama variabel yang dipakai.
ukuran	:	menunjukkan jumlah maksimal elemen larik.

Contoh :

```
Int nilai[6];
```

INISIALISASI

Menginisialisasi array sama dengan memberikan nilai awal array pada saat didefinisikan.

```
int nilai[6] = {8,7,5,6,4,3};
```

Contoh diatas berarti berarti anda memesan tempat di memori komputer sebanyak 6 tempat dengan indeks dari 0-5, dimana indeks ke-0 bernilai 8, ke-1 bernilai 7, dst, dan dimana semua elemennya bertipe data integer.

PENGAKSESAN

```
nama_var_array [indeks];
```

Pengisian dan pengambilan nilai pada indeks tertentu dapat dilakukan dengan mengeset nilai atau menampilkan nilai pada indeks yang dimaksud. Pengaksesan elemen array dapat dilakukan berurutan atau random berdasarkan indeks tertentu secara langsung.

Contoh pengisian langsung saat deklarasi:

```
#include <stdio.h>

void main ()
{ int billy [] = {16, 2, 77, 40, 12071};
  int n, result=0;
  for ( n=0 ; n<5 ; n++ )
  {
    result += billy[n];
  }
  printf("%d", result);
}
```

Contoh pengaksesan dan pengisian langsung ke tiap elemen dari array:

```
#include <stdio.h>
#include <conio.h>
```

Modul I

```
void main ()
{
    int A [5]={20,9,1986,200,13},n,edit;
    clrscr();
    printf("Data yang lama\n");
    for (n=0;n<5;n++)
    {
        printf("%i ",A[n]);
    }
    printf("\nData yang baru : \n");
    A[0]=4;
    A[1]=2;
    A[2]=1;
    A[3]=3;
    A[4]=5;
    for (n=0;n<5;n++)
    {
        printf("%i ",A[n]);
    }
}
```

Contoh penghapusan data(elemen) pada array:

```
#include <stdio.h>
#include <conio.h>

void main ()
{ int A [5]={20,9,1986,200,13},n,hapus;
  clrscr();
  printf("Data yang lama\n");
  for (n=0;n<5;n++)
  {
      printf("%i ",A[n]);
  }
  printf("data yang ingin dihapus : ");
  scanf("%i",&hapus);
  printf("\nData yang baru : \n");
  for (n=hapus-1;n<5-1;n++)
  {
      A[n]=A[n+1];
  }
  for (n=0;n<4;n++)
  {
      printf("%i ",A[n]);
  }
}
```

LATIHAN TERBIMBING ARRAY

1. Penghapusan data pada array.

Ketentuan : data yang akan dihapus adalah data pada indeks ke-0 (data paling depan).

Source code :

Modul I

```
#include <stdio.h>
#include <conio.h>

void main() {
    int data[]={1,6,2,9,12,87,43,11};
    int n=8;

    do{
        //menampilkan data
        for(int i=0;i<n;i++)
            printf("%i ",data[i]);

        getch();
        printf("\n");

        // untuk menghapus data
        for(i=0;i<n-1;i++)
            data[i]=data[i+1];

        // jumlah data berkurang
        n=n-1;

        printf("\n");
        if(n==0) printf("Data kosong!!!");
    }while(n!=-1);
}
```

Cobalah program di atas.

Apakah yang terjadi?

Bagaimana jika penghapusan terjadi pada data yang berada di indeks terakhir? Bagian program yang mana yang harus diubah?

Modul I

STRUCT

- Bentuk struktur data yang dapat menyimpan variabel-variabel dalam 1 nama, namun memiliki tipe data yang berbeda ataupun sama. Variable-variabel tersebut memiliki kaitan satu sama yang lain.

Bentuk umum :

```
typedef struct nama_struct{
tipe_data <nama_var>;
tipe_data <nama_var>;
....
};
```

DEKLARASI

Ada 2 cara pendeklarasian struct, yaitu :

Deklarasi 1:

```
typedef struct Mahasiswa {
char NIM[8];
char nama[50];
float ipk;
};
```

Deklarasi 2 :

```
struct {
char NIM[8];
char nama[50];
float ipk;
} mhs;
```

Contoh struct:

```
#include <stdio.h>
#include <iostream.h>

void main()
{
    struct orang
    {
        char nama[40];
        short umur;
    }saya;
    printf("nama : ");
    cin.getline(saya.nama,40);
    printf("umur :" );
    scanf("%i",&saya.umur);
    printf("%s berumur %i",saya.nama,saya.umur);
}
```

ARRAY OF STRUCT

Apabila hendak menggunakan 1 struct untuk beberapa kali, ada 2 cara :

1. Deklarasi manual

Contoh :

Modul I

```
#include <stdio.h>
typedef struct Mahasiswa {
    char NIM[8];
    char nama[50];
    float ipk;
};
void main()
{
    Mahasiswa a,b,c;
    .....
    .....
    .....
}
```

artinya struct mahasiswa digunakan untuk 3 variabel, yaitu a,b,c

2. Array of struct

Contoh :

```
#include <stdio.h>
typedef struct Mahasiswa {
    char NIM[8];
    char nama[50];
    float ipk;
};
void main()
{
    Mahasiswa mhs[3];
    .....
    .....
    .....
}
```

artinya struct mahasiswa dapat digunakan untuk tiga variabel mhs, yaitu mhs[0], mhs[1], dan mhs[2].

Contoh lainnya :

```
#include <stdio.h>
#include <iostream.h>
#include <conio.h>
typedef struct orang
{
    char nama[30];
    short umur;
};
void main()

{
    orang saya[5];
    int i,x;
    for(i=0;i<=4;i++)
    {
        printf("nama ke-%i : ",i+1);
        cin.getline(saya[i].nama,30);
        printf("umur ke-%i : ",i+1);
        scanf("%i",saya[i].umur);
        printf("%s berumur %i",saya[i].nama,saya[i].umur);
    }
    for(x=0;x<=4;x++)
    {
        printf("nama %s berumur %d",saya[x].nama,saya[x].umur);
    }
}
```

Modul I

```
}  
}
```

LATIHAN TERBIMBING STRUCT

Buatlah struct untuk buku dengan deklarasi manual.

Ketentuan :

Yang harus disimpan adalah judul buku, tahun terbit dan harga buku.

Source code :

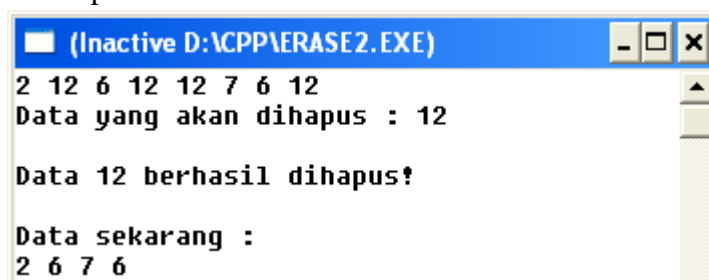
```
#include <stdio.h>  
#include <conio.h>  
  
typedef struct buku{  
    char judul[15];  
    int tahun_terbit;  
    int harga;  
};  
  
void main(){  
  
    buku book;  
    printf("Judul buku : ");scanf("%s",&book.judul);  
    printf("Tahun terbit : ");scanf("%i",&book.tahun_terbit);  
    printf("Harga : ");scanf("%i",&book.harga);  
  
    printf("\nJudul buku : %s\n",book.judul);  
    printf("Tahun terbit : %i\n",book.tahun_terbit);  
    printf("Harga : %i",book.harga);  
}
```

LATIHAN - LATIHAN MANDIRI DI KELAS

1. Program penghapusan data dengan inputan berupa angka yang ingin dihapus oleh user.

Ketentuan :

- Semua data yang sesuai dengan inputan user akan terhapus.
- Bonus jika terdapat counter untuk menghitung berapa data yang terhapus.
- Capture :



```
(Inactive D:\CPP\ERASE2.EXE)  
2 12 6 12 12 7 6 12  
Data yang akan dihapus : 12  
  
Data 12 berhasil dihapus!  
  
Data sekarang :  
2 6 7 6
```

Modul I

2. Program untuk melakukan update data.

Ketentuan :

- Terdapat dua inputan, yaitu inputan data yang akan diubah dan data baru (data pengganti)
- Semua data yang sesuai dengan inputan user akan diupdate nilainya.

3. Program untuk melakukan penambahan data.

Ketentuan :

- Penambahan data dapat dilakukan di mana saja.
- Inputan dari user berupa :
 - Nilai yang akan ditambahkan
 - Indeks ke berapa yang dituju
- Setelah penambahan, maka jumlah data akan bertambah dan posisi data akan bergeser sesuai dengan penambahan yang telah dilakukan.

4. Buatlah struct untuk data lagu yang berisi tentang judul lagu, penyanyi, tahun produksi, nomor track dan kode album.

Ketentuan :

- program ini akan memiliki dua buah struct, yaitu struct lagu dan struct kodeRBT.
- Jumlah data yang diinputkan dinamis (maks. 20 lagu)

LATIHAN – LATIHAN MANDIRI DI RUMAH / TES

1. Buatlah menu add, edit, view dan delete data menggunakan array.

Note : operasi-operasi tersebut dapat dilakukan pada data dan indeks mana saja. (Inputan data dan indeks dinamis).

2. Buatlah dengan menggunakan struct dan array 1 dimensi : record peminjaman buku di perpustakaan.

Data yang akan ditampilkan sebagai output adalah :

- Nama
- NIM
- Tanggal peminjaman (dd/mm/yyyy)
- Kode buku, dengan format **nomor rak-kategori buku** :
 - Nomor rak (inputan terserah)
 - Kategori : R (Referensi) atau U (Umum)
 - Contoh : 1234-R

Gunakan tiga struct untuk kasus ini!

Gabungkan soal 1 dan 2 di atas sehingga menghasilkan program berisi struct yang dapat melakukan fungsi add, edit, view dan delete

Modul I

Contoh tampilan program :

```

Berapa data yang anda inputkan? 1
Data ke-1:
  Judul lagu : Bersamamu
  Penyanyi : Vierra
  Tahun Produksi : 2009
  Nomor Track : 3
  Kode Album : MFL

Data lagu yang anda miliki :
  Data 1:
    Penyanyi : Vierra
    Judul lagu : Bersamamu
    Kode RBT : 3-MFL
    Tahun : 2009

```

<Asisten-asisten dapat memodifikasi atau mengembangkan soal-soal di atas sesuai dengan kelasnya>

END