

Array 1 Dimensi

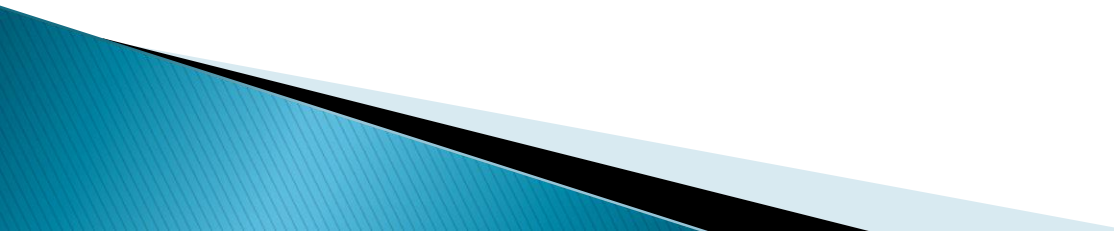
Pemograman C++

Array

Array digunakan untuk:

- ❑ Menyimpan data–data yang diinputkan masing– masing kedalam memory komputer secara bersebelahan/homogen
- ❑ Ukuran atau jumlah elemen maksimum array telah diketahui dari awal yaitu ketika array dibuat.
- ❑ Sekali ukuran array ditentukan maka tidak dapat diubah. Ukuran array adalah bilangan bulat positif.
- ❑ Array harus diberi nama sebagai identifikasi
- ❑ Contoh kasus yang membutuhkan array:
 - ✓ Daftar pegawai perusahaan tertentu
 - ✓ Daftar nilai tes dari suatu matakuliah tertentu
 - ✓ Daftar customer dan nomor teleponnya

Array

- ❑ Di dalam C dan pemrograman yang lain, terdapat suatu fasilitas untuk menyimpan data-data yang bertipe data sama dengan suatu nama tertentu = ARRAY/LARIK
 - ❑ Array adalah suatu tipe data terstruktur yang berupa sejumlah data sejenis (bertipe data sama) yang jumlahnya tetap dan diberi suatu nama tertentu.
 - ❑ Elemen-elemen array tersusun secara sekuensial di dalam memori sehingga memiliki alamat yang berdekatan/bersebelahan.
 - ❑ Array dapat berupa array 1 dimensi, 2 dimensi, bahkan n- dimensi.
 - ❑ Elemen-elemen array bertipe data sama tapi bisa bernilai sama atau berbeda-beda.
- 

Bentuk Array dalam Memory (int)

0	1	2	3	4	5	6	7	indeks
8	10	6	-2	11	7	1	100	value
ffea	ffeb	ffec	ffed	ffef	fffa	fffb	fffc	alamat

Array (2)

- ❑ Elemen–elemen array dapat diakses oleh program menggunakan suatu indeks tertentu
- ❑ Pengaksesan elemen array dapat dilakukan berurutan atau random berdasarkan indeks tertentu secara langsung.
- ❑ Cara mengaksesnya adalah dengan menyebutkan nama array dan indeksinya. Indeks array dimulai dari 0 sampai dengan $n-1$ (n adalah ukuran array).
- ❑ Dalam C, tidak terdapat error handling terhadap batasan nilai indeks, apakah indeks tersebut berada di dalam indeks array yang sudah didefinisikan atau belum.
- ❑ Hal ini merupakan tanggung jawab programmer.
- ❑ Di bahasa pemrograman lain: array index out of bounds exception

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.

Tipe data sejenis

- Ada indeks yang teratur dan berurutan

- Bersifat statis, harus diketahui ukurannya terlebih dahulu

Argumen Array

/*Array sebagai argumen input*/

```
int get_max(const int list[], int n)
{
    int i, cur_large;
    cur_large=list[0];
    for(i=1;i<n;++i)
        if (list[i] > cur_large)
            cur_large=list[i];
    return(cur_large);
}
```

Argumen Array

/*Array sebagai argumen input dan array sebagai argumen output*/

```
void add_array( const double ar1[], const
double ar2[], double arsum[], int n )
{
int i;
for(i=1;i<n;++i)
arsum[i]=ar1[i]+ar2[i];
}
```


Contoh dan Arti

```
char huruf[9];  
int umur[10];  
int kondisi[2] = {0,1}  
int arr_dinamis[] = {1,2,3}
```

- `char huruf[9]` berarti akan memesan tempat di memori komputer sebanyak 9 tempat dengan indeks dari 0-8, dimana semua elemennya bertipe data karakter semuanya. Kalau satu karakter berukuran 1 byte, berarti membutuhkan memori sebesar 9 byte.
- `int umur[10]`: berarti akan memesan tempat di memori komputer sebanyak 10 tempat dengan indeks dari 0-9, dimana semua elemennya bertipe data integer semuanya. Kalau satu integer berukuran 4 bytes, berarti membutuhkan memori sebesar $4 \times 10 = 20$ bytes.

Array

- ▶ ☐ Array : kumpulan data yang bertipe sama.
- ▶ ☐ Contoh deklarasi, inisialisasi, dan penggunaan array bisa dilihat pada contoh potongan program berikut :
- ▶ `typedef enum {Senin, Selasa, Rabu, Kamis, Jumat} class_day_t;`
- ▶ `...`
- ▶ `char answer[]={‘T’, ‘F’, ‘T’, ‘F’, ‘F’, ‘T’, ‘T’, ‘F’, ‘F’, ‘F’};`
- ▶ `int score[5], i;`
- ▶ `...`
- ▶ `i=1;`
- ▶ `printf(“\n %d %d”, i, score[i]);`
- ▶ `printf(“\n %d”, score[i]+1);`
- ▶ `printf(“\n %d”, score[i+1]);`
- ▶ `printf(“\n %d”, score[i*2]);`
- ▶ `scanf(“%d”, &score[i]);`
- ▶ `X[i+1]=X[i+2];`
- ▶ `....`

Contoh dalam C

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

int main()
{ int nilai[5], x;
  clrscr();

  printf("Memasukkan nilai :\n");
  for(x=0;x<5;x++)
  {
    printf("Nilai Angka : "); scanf("%d",&nilai[x]);
  }
  printf("\n");

  printf("Membaca nilai :\n");
  for(x=0;x<5;x++)
  {
    printf("Nilai Angka : %d",nilai[x]);
  }

  getch();
}
```



```
Command Prompt (2) - tc
Memasukkan nilai :
Nilai Angka 4
Nilai Angka 7
Nilai Angka 3
Nilai Angka 9
Nilai Angka 6

Membaca nilai :
Nilai Angka 4
Nilai Angka 7
Nilai Angka 3
Nilai Angka 9
Nilai Angka 6
```

Contoh (2)

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

int main(){
    int bil[7],i;
    printf("elemen 1 ? ");scanf("%d",&bil[0]);
    bil[1] = 5;
    bil[2] = bil[1] + 20;
    for(i=4;i<7;i++) bil[i] = i*10;
    bil[3] = bil[bil[1]];
    for(i=0;i<7;i++) printf("bil[%d] = %d dan alamatnya:
%X\n",i,bil[i],&bil[i]);
    getch();
    return 0;
}
```



```
elemen 1 ? 33
bil[0] = 33 dan alamatnya: 22FF50
bil[1] = 5 dan alamatnya: 22FF54
bil[2] = 25 dan alamatnya: 22FF58
bil[3] = 50 dan alamatnya: 22FF5C
bil[4] = 40 dan alamatnya: 22FF60
bil[5] = 50 dan alamatnya: 22FF64
bil[6] = 60 dan alamatnya: 22FF68
```

Array

- Array adalah suatu variabel yang dapat menyimpan beberapa item data yang tipenya sama
- Deklarasi array :

Format :

```
<type_data> <nama_variabel>[exp1][exp2][...]
```

Contoh :

```
int bulan[12];  
float harga[50];  
char nama[20];
```

- Indeks awal array dimulai dari 0 sehingga variabel bulan di atas terdiri dari bulan[0] s/d bulan[11].
- Nilai array dapat langsung diinisial seperti contoh berikut :

```
int A[5] = {15, 34, 32, 12, 11};
```

artinya : mendeklarasikan A sebagai variabel array sekaligus memberikan nilai 15 ke A[0], 34 ke A[1], 32 ke A[2], 12 ke A[3], 11 ke A[4].

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

```
void main()
```

```
{
```

```
    int a;
```

```
    int b[4];
```

```
    for (a=0;a<4;a++)
```

```
    {
```

```
        printf("Masukkan nilai B[%d] :
```

```
“,a);scanf("%d",&b[a]);
```

```
    }
```

```
    for (a=0;a<4;a++)
```

```
    {
```

```
        printf("\nNilai B[%d} adalah : %d",a,b[a]);
```

```
    }
```

```
    getch();
```

```
}
```

Array untuk mencari bilangan terbesar dari sederet bilangan

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

void main()
{
    int data[10];
    int elemen,a;
    int maks=0;
    clrscr();

    //entri 10 data
    for(elemen=0;elemen<=9;elemen++)
    {
        cout<<"Data ke -
"<<elemen;cout<<" = ";
        cin>>data[elemen];
    }
```

```
//tampilkan data setelah entri

for(elemen=0;elemen<=9;elemen
++)
{
    if (maks<data[elemen])
    {
        maks=data[elemen];
        a=elemen;
    }
    cout<<"\nData Terbesar adalah
data yang ke - "<<a<<" =
"<<maks;
    getch();
}
```

Contoh program untuk menguji array dan looping dengan for :

```
/* Nama File : acak.c */
#include <stdio.h>
#include <stdlib.h>

void main()
{
    int acak[5][5][5];
    int i,j,k;

    /*Harga awal untuk fungsi random*/
    srand(1234);

    for (i=0;i<5;i++)
    {
        for (j=0;j<5;j++)
        {
            for (k=0;k<5;k++)
            {
                acak[i][j][k]=rand();
                printf("\n Acak[%d][%d][%d]\n
                    = %d",i,j,k,acak[i][j][k]);
            }
        }
    }
}
```

Latihan : ubah kontrol pengulangan **for** pada kode program di atas dengan **while** !

Array dan fungsi

- Pada fungsi sebagai argumen formal array hanya diberi tanda [] saja tanpa diberi indeks, sedangkan sebagai argumen aktual array yang disebut namanya saja tanpa [].

Contoh :

...

//called function

```
int maksimum(int d[],maks)
```

```
{
```

```
...
```

```
}
```

```
void main()
```

```
{
```

```
int dat[10];
```

```
...
```

//Calling function

```
Hasil=maksimum(dat,n);
```

```
}
```

Latihan : Lengkapi kode program tersebut yang tujuannya mencari nilai maksimum dari sekelompok nilai yang dimasukkan melalui keyboard

Array dan pointer

- Nama array adalah alamat memory (pointer) pertama dari elemen array tersebut. Jika A adalah array berdimensi satu maka alamat memory dari elemen pertama adalah &A[0] atau A saja. Sehingga &A[i] sama dengan (A+i). Maka A[i] pun sama dengan *(A+i)
- Berikut contoh program tentang bagaimana mengakses data array menggunakan konsep pointer :

```
/* Nama file :p_array.c */

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

void main()
{
    int Nilai[5]={55,75,60,90,80};

    clrscr();
    for(int i=0;i<5;i++)
    {
        //akses dengan indeks array
        printf("\nNilai[%d] = %d",i,Nilai[i]);
        //akses dengan pointer
        printf("\n*(Nilai+%d)= %u",i,*(Nilai+i));
    }
    getch()
}
```

String

- String adalah array dari tipe character yang diakhiri dengan null character (\0)
- Data konstan string ditulis dengan diapit oleh tanda petik dua ("...string...")
- Deklarasi tipe string :

Format :

```
char <nama_variabel>[exp]
```

atau

```
char <nama_variabel>[]={inisial}
```

atau

```
char *<nama_variabel>
```

Contoh :

```
char nama[];  
char *alamat;
```

- Pemberian nilai inisial ke string dapat dilakukan dengan :

```
char nama[]="Hanif";
```

Atau

```
char nama[]={ 'H', 'a', 'n', 'i', 'f', '\0' };
```

- Deklarasi `char *nama` menunjukkan bahwa string adalah pointer, atau dapat diakses dengan cara-cara yang sama dengan pointer atau array
- Contoh program mengakses string dalam function :

```
/*Nama file : p_string.c */  
  
#include <stdio.h>  
#include <string.h>  
#include <conio.h>  
  
void ubah_nama(char *nama)  
{  
    nama=strcat(nama, " Disa");  
}  
  
void main()  
{  
    char nm[30];  
    clrscr();  
    printf("\nMasukkan nama anda :"); gets(nm);  
    ubah_nama(nm);  
    printf("\nNama anda sekarang : %s", nm);  
    getch();  
}
```

Structure

- Struktur yaitu suatu variabel yang dapat menyimpan sejumlah item dari tipe data yang berbeda.
- Struktur pada C sama dengan record pada pascal
- Deklarasi Struktur :

Format :

```
struct <nama_struktur>
{
    member1
    member2
    member3
    ...
};
```

- Setelah struktur dideklarasikan, selanjutnya dapat dideklarasikan variabel jenis struktur :

```
<nama_struktur> <nama_variabel>
```

Contoh :

```
struct mahasiswa
{
    char nim[10];
    char nama[25];
    int umur;
};

mahasiswa amir, amat, tuti;
```

- Boleh juga digabung langsung seperti berikut :

```
struc mahasiswa
{
    char nim[10];
    char nama[25];
    int umur;
} amir, amat, tuti;
```

- Dari deklarasi tersebut berarti variabel amir memiliki 3 elemen yaitu :

amir.nim, amir.nama dan amir.umur

- Variabel struktur dapat langsung diinisialisasi seperti berikut :

```
mahasiswa amir ={"20019823","Amir mahmud",19};
```

yang sama dengan :

```
amir.nim="20019823";
amir.nama="Amir mahmud";
amir.umur=19;
```

Tugas/Latihan :

Buatlah program data mahasiswa yang memiliki tiga operasi yaitu : entri, tampil dan edit data. Item datanya terdiri dari : nim,nama, alamat,tempat & tanggal lahir. Bila perlu gunakan menu sederhana untuk memilih operasi.

Anda perlu membaca referensi lain untuk menyelesaikan tugas ini !

Soal

- ▶ Buatlah program untuk menyalin isi array A ke array B dengan jumlah elemen yang sama.

Ilustrasinya:

Array: int A[5];

Indeks	0	1	2	3	4
--------	---	---	---	---	---

Nilai	5	10	6	0	4
-------	---	----	---	---	---

Array: int B[5];

Indeks	0	1	2	3	4
--------	---	---	---	---	---

Nilai	--	--	--	--	--
-------	----	----	----	----	----

Kemudian array B akan berisi sama dengan isi array A:

Array: int B[5];

Indeks	0	1	2	3	4
--------	---	---	---	---	---

Nilai	5	10	6	0	4
-------	---	----	---	---	---

Soal

Kembangkan soal tadi, dimana array B akan berisi elemen–elemen array A tapi dengan posisi terbalik, data pertama di array A akan berada di data terakhir di array B!

Array: int B[5];

Indeks	0	1	2	3	4
--------	---	---	---	---	---

Nilai	4	0	6	10	5
-------	---	---	---	----	---