

PRAKTIKUM PEMROGRAMAN KOMPUTER
MINGGU 9
Array

A. Tujuan praktikum:

- Mahasiswa mampu membuat program sederhana dengan menggunakan array

B. Dasar Teori

I. LATAR BELAKANG ARRAY

- Sejauh ini cuplikan program yang kita pelajari masih sangat terbatas, karena statement assignment hanya berupa pemberian satu nilai pada satu variabel
- Padahal sering kali kita perlu untuk meng-assign ataupun memanipulasi banyak nilai ke sekelompok variabel
- Sebagai contoh jika anda membuat program yang membaca 6000 nilai percobaan untuk dihitung nilai rata-rata-nya, akan sangat lucu jika anda menghitungnya dengan cara berikut: $\text{rata_rata} = (x_1 + x_2 + x_3 + x_4 + x_5 + \dots + x_{6000}) / 6000$.
- Ekspresi matematis berikut akan lebih baik :
$$\bar{x} = \frac{1}{n} \sum_{k=1}^n x_k$$

II. ARRAY DALAM BAHASA C

- Definisi : Kumpulan nilai dengan tipe data yang sama yang menggunakan nama sama.
- Deklarasi Array :
 - Tipe data elemen array
 - Nama array
 - Jumlah elemen array
- Contoh :

```
int val[ 200 ];           //declaration
val[ 12 ] = 5;           //assignment
```
- Jika sebuah array y memiliki n elemen, maka:
 - Elemen pertama adalah : y[0]
 - Elemen terakhir adalah : y[n-1]
- Contoh :
 - Short y[4], memiliki elemen y[0], y[1], y[2], y[3]
 - Tetapi y[4], bukan merupakan elemen array ini.
- Array dapat dibedakan menjadi :
 - Array berdimensi satu (1D), berpadanan dengan vektor di Matematika
 - Array berdimensi dua (2D), berpadanan dengan matriks di Matematika
 - Array berdimensi banyak
- Contoh array integer 1D bernama grade yg memiliki 5 elemen,
grade[0] = 1, grade[1] = 3, grade[2] = 5, grade[3] = 7, grade[4] = 9 dideklarasikan sbb:

| | | | | | |
|--------|---|---|---|---|---|
| grade | 1 | 3 | 5 | 7 | 9 |
| indeks | 0 | 1 | 2 | 3 | 4 |

- int grade[5] = {1,3,5,7,9};

III. INISIALISASI ARRAY

Saat mendeklarasikan sebuah array yang bersifat lokal (misalnya dalam suatu fungsi), jika tidak ditentukan, elemennya tidak akan diisi dengan sesuatu nilai sampai ada statemen untuk mengisinya. Tetapi elemen-elemen dari array yang bersifat global secara otomatis akan diinisialisasi dengan nilai nol.

Kita dapat memberikan nilai awal pada masing-masing elemen array dengan cara menyebutkan nilai-nilainya diikuti tanda kurung kurawal { }. Sebagai contoh:

```
int grade [5] = {1, 3, 5, 7, 9};
```

Deklarasi tersebut akan menciptakan array seperti berikut:



Banyaknya nilai di antara tanda kurung { } tidak boleh melebihi banyaknya elemen yang telah dideklarasikan. Sebagai contoh, pada contoh array **grade** di atas telah dideklarasikan memiliki 5 elemen dan pada daftar nilai awal di dalam tanda kurung { } kita tentukan 5 nilai masing-masing untuk satu elemen.

Saat menginisialisasi sebuah array, nilai jumlah elemen dalam tanda kurung siku [] boleh dikosongkan. Dalam kasus ini kompiler akan mengasumsikan ukuran array sebanyak nilai yang dimasukkan dalam tanda kurung { }:

```
int grade [] = {1, 3, 5, 7, 9};
```

IV. MENGAKSES NILAI SEBUAH ARRAY

Kita dapat mengakses nilai elemen array secara individual seperti variabel normal untuk membaca atau mengubahnya. Formatnya:

nama_array[index]

Sebagai contoh, untuk menyimpan nilai 5 ke dalam elemen **grade** pada posisi ketiga kita dapat menulis statemen berikut:

```
grade[2] = 5;
```

dan untuk mengambil nilai elemen ketiga dari array **grade** untuk disimpan ke variabel **item** kita dapat menulis:

```
item = grade[2];
```

Catatan: Pada contoh di atas, elemen ketiga dari array **grade** memiliki nomor indeks 2 karena penomoran dimulai dari 0.

CONTOH 1

Program berikut mencatat data-data suhu harian dalam sepekan, kemudian menampilkan kembali seluruh data yang telah dimasukkan dan nilai rata-ratanya.

```
void main(void) {
    int i, suhu[7];
    float rerata, jml;
    clrscr();
    cout<<"Program mencatat suhu udara harian\n";
    cout<<"Masukkan data-data suhu udara dalam sepekan!\n";

    // input data
    for (i=1; i <= 7; i++)
    {
        cout<<"Suhu hari ke- "<< i;
        cin>>suhu[i-1];
    }

    // menjumlahkan elemen array
```

```

jml = 0;
for (i=0; i <= 6; i++)
{
    jml = jml + suhu[i];
}

// menghitung suhu rata-rata
rerata = jml / 7;

// menampilkan kembali data-data yg telah dicatat dan rata-
ratanya
cout<<"Data-data suhu yang telah dicatat adalah:\n";
for (i=0; i <= 6; i++)
    cout<< suhu[i]<<endl;
cout<<"Suhu rata-rata dalam sepekan = "<< rerata<<endl;
getch();
}

```

CONTOH 2

Program berikut mencatat 10 data nilai mata kuliah yang diperoleh seorang mahasiswa, kemudian menampilkan kembali seluruh data yang telah dimasukkan. Nilai matakuliah disimpan dalam array bertipe **char**.

```

void main(void) {
    int i;
    char nilai[10];
    clrscr();
    cout<<"Program mencatat data nilai matakuliah\n";
    cout<<"Masukkan data-data nilai (huruf A-E) dari 10 mahasiswa\n";

    // input data
    for (i=1; i <= 10; i++)
    {
        cout<<"Nilai matakuliah ke- "<< i<<";
        cin>>nilai[i-1];
    }

    // menampilkan kembali data-data yg telah dicatat
    cout<<"Data-data nilai yang telah dicatat adalah:\n";
    for (i=0; i <= 9; i++)
        cout<< nilai[i]<<" ";
    getch();
}

```

materi diambil dari : http://lecturer.eepis-its.edu/~tessy/lecturenotes/kp1/Array_dalam_BahasaC_bahan4.ppt

C. LATIHAN

Untuk setiap soal di bawah ini, buatlah algoritmanya terlebih dahulu.

1. Buat program untuk memasukkan dan menampilkan n bilangan.

Contoh :

Masukkan banyak bilangan : 4

Masukkan bilangan 1 : 30

Masukkan bilangan 2 : 20

Masukkan bilangan 3 : 10

Masukkan bilangan 4 : 40

Bilangan yang anda masukkan adalah :

Bilangan 1 : 30

Bilangan 2 : 20

Bilangan 3 : 10

Bilangan 4 : 40

2. Buat program untuk memasukkan bilangan sebanyak n (masukan user). Kemudian program akan meminta user untuk mencari bilangan, dan program akan menampilkan bilangan tersebut ditemukan pada posisi indeks tertentu.

Contoh :

Masukkan banyak bilangan (maks 10) : 4

Masukkan bilangan 1 : 30

Masukkan bilangan 2 : 20

Masukkan bilangan 3 : 10

Masukkan bilangan 4 : 40

Bilangan berapa yang anda cari ? 10

Bilangan 10 ada di posisi ke-3

Mau cari lagi ? Y

Bilangan berapa yang anda cari ? 50

Bilangan 50 tidak ada.

Mau cari lagi ? T

Terima kasih

3. Buat program untuk memasukkan bilangan sebanyak n (masukan user) (seperti no.2). Kemudian program akan menampilkan data bilangan yang memiliki nilai lebih besar dari rata-rata seluruh data tersebut beserta urutan berapa.

Contoh :

Masukkan banyak bilangan (maks 10) : 4

Masukkan bilangan 1 : 30

Masukkan bilangan 2 : 20

Masukkan bilangan 3 : 10

Masukkan bilangan 4 : 40

Bilangan yang lebih besar dari rata-rata (25) adalah :

30 (urutan 1)

40 (urutan 4)

Terima kasih

4. Buat program untuk memasukkan bilangan sebanyak n (masukan user) (seperti no.2). Kemudian program akan menampilkan bilangan terbesar dan terkecil dari yang telah dimasukkan user..

Contoh :

Masukkan banyak bilangan (maks 10) : 6
Masukkan bilangan 1 : 30
Masukkan bilangan 2 : 20
Masukkan bilangan 3 : 10
Masukkan bilangan 4 : 40
Masukkan bilangan 5 : 99
Masukkan bilangan 6 : 56

Bilangan terbesar : 99

Bilangan terkecil : 10

Terima kasih

5. Buat program untuk memasukkan bilangan sebanyak n (masukan user) (seperti no.2). Kemudian program akan menampilkan data bilangan tersebut dengan urutan berkebalikan.

6. Buatlah program untuk memasukkan 10 data dengan nilai antara 10 - 70, kemudian menampilkan diagram bintang dari data-data tsb.

Contoh:

Masukkan data ke-1: 20
Masukkan data ke-2: 25
Masukkan data ke-3: 60
Masukkan data ke-4: 30
Masukkan data ke-5: 50
Masukkan data ke-6: 30
Masukkan data ke-7: 27
Masukkan data ke-8: 65
Masukkan data ke-9: 48
Masukkan data ke-10: 15

```
20: *****
25: *****
60: *****
30: *****
50: *****
30: *****
27: *****
45: *****
48: *****
15: *****
```

7. Buat program untuk menampilkan sebuah deret. Dengan aturan :
 $\text{bilangan}[n] = \text{bilangan}[n-1] + \text{bilangan}[n-2];$

Bilangan pertama dan kedua sudah diset awal bernilai 1. User diminta memasukkan jumlah deret yang akan ditampilkan.

Contoh :

Masukkan jumlah deret : 8

Deret adalah : 1, 1, 2, 3, 5, 7, 9, 11

C. Laporan Praktikum

Format Laporan Praktikum

1. Tujuan Praktikum (5%)

2. Untuk setiap soal pada latihan , kerjakan algoritma (30%), capture program , beserta output. (30%)
3. Analisa setiap program latihan (secara keseluruhan program melakukan apa / apa ide dasar, perintah yang digunakan dan mengapa perintah tersebut dilakukan) (30%)
4. Daftar Pustaka (5%) => buku yang anda pakai untuk belajar

Catatan

- **Laporan terlambat setelah 1 minggu tidak akan diberi nilai.**
- **Keterlambatan akan diberi nilai sebenarnya dan pengurangan 5 untuk setiap harinya.**
- **Laporan yang sama atau ditengarai sebagai hasil penjiplakan/plagiat maka tidak akan diberi nilai.**

@ You are nothing if you don't do something @