# MODUL I: Array

1.Materi

* + Array

2.Tujuan

* + Mahasiswa dapat memahami struktur Array
  + Mahasiswa dapat membuat Array sederhana

3. Teori

* + Variabel array adalah variabel yang dapat menyimpan beberapa item data sekaligus. Setiap item data lokasi penyimpanannya dibedakan oleh nomor indeks dalam kurung siku [ ]. Setiap item data pada array type dasarnya harus sama, misalnya sama-sama integer, sama-sama char, float dan seterusnya.
  + Dengan menggunakan array maka kebutuhan pengolahan data bertipe sama secara serentak dapat dilakukan lebih mudah dengan bantuan looping dan umumnya menggunakan looping for.
  + Pada praktikum kali ini akan dilakukan percobaan pengolahan data menggunakan array meliputi :
  + Cara Memasukkan/meng-inputkan data ke array
  + Cara mencetak data pada array
  + Men-total-kan data integer pada array
  + Me-rata-rata-kan data integer pada arrayLakukan percobaan dengan program-program berikut :

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

//Program Percobaan menggunakan tipe Array

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

#include <stdio.h>

#include <conio.h>

void main()

{

int a[5]; //mendeklarasikan variavel tipe array

int i;

a[0]=4; //memasukkan data ke variabel array indek ke-0

a[1]=7; //memasukkan data ke variabel array indek ke-1

a[2]=12; //memasukkan data ke variabel array indek ke-2

a[3]=9; //memasukkan data ke variabel array indek ke-3

a[4]=8; //memasukkan data ke variabel array indek ke-4

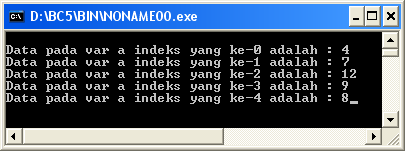
//mencetak data array dengan looping for

for(i=0;i<=4;i++)

printf("\nData pada var a indeks yang ke-%d adalah : %d",i,a[i]);

getch();

}



//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

//Program Percobaan menggunakan tipe Array

//Memasukkan data dengan scanf

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

#include <stdio.h>

#include <conio.h>

void main()

{

int a[5]; //mendeklarasikan variavel tipe array

int i;

//Masukkan data ke var array dengan memanfaatkan looping for

for(i=0;i<=4;i++)

{

printf("Masukkan data integer untuk a[%d] :",i);scanf("%d",&a[i]);

}

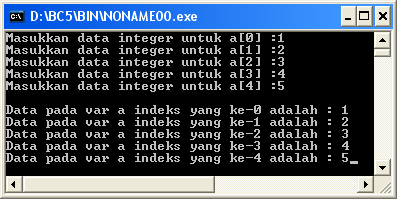
//mencetak data array dengan looping for

for(i=0;i<=4;i++)

printf("\nData pada var a indeks yang ke-%d adalah : %d",i,a[i]);

getch();

}



//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

//Program Percobaan menggunakan tipe Array

//Mentotalkan data pada array

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

#include <stdio.h>

#include <conio.h>

void main()

{

int a[5]; //mendeklarasikan variavel tipe array

int i,total;

//Masukkan data ke var array dengan memanfaatkan looping for

for(i=0;i<=4;i++)

{

printf("Masukkan data integer untuk a[%d] :",i);scanf("%d",&a[i]);

}

//Menghitung total pada array dengan looping for

total=0;

for(i=0;i<=4;i++)

{

total=total+a[i];

}

//mencetak data array dengan looping for

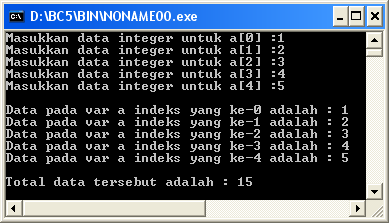
for(i=0;i<=4;i++)

printf("\nData pada var a indeks yang ke-%d adalah : %d",i,a[i]);

printf("\n\nTotal data tersebut adalah : %d",total);

getch();

}



## Latihan :

1. Berdasarkan program di atas buatlah program untuk me-rata-ratakan 5 bilangan integer dalam array !
2. Berdasarkan program no. 3 di atas, buatlah program untuk me-rata-ratakan 10 buah bilangan integer !
3. Berdasarkan program di atas, buatlah program me-rata-rata-kan sebanyak N buah bilangan, dimana nilai N dimasukkan terlebih dahulu sebelum data diinputkan dengan scanf !
4. Modifikasi program 3 di atas sehingga dengan 5 buah bilangan integer pada var a menggunakan penomoran indeks 1, 2, 3, 4, 5 bukan 0, 1, 2, 3, 4.

# MODUL II: Pointer

1.Materi

* + Pointer

2.Tujuan

* + Mahasiswa dapat menggunakan Pointer
  + Mahasiswa dapat menggunakan Pointer dengan argument fungsi call by reference
  + Memahami penggunaan pointer ke fungsi

3.Teori

**Pointer**

* Merupakan variabel yang fungsinya menyimpan alamat (address) variabel
* Fasilitas untuk mengakses suatu variabel dengan memakai adress

**Kegunaan**

* Calling by reference
* Mengembalikan lebih satu nilai dari satu fungsi
* Mengirimkan array dan string dari suatu fungsi ke fungsi lain
* Memudahkan dalam manipulasi array
* Manipulasi memory dalam system
* Membuat struktur data dinamis seperti linklist, tree, graph dan sebagainya

**Deklarasi pointer**

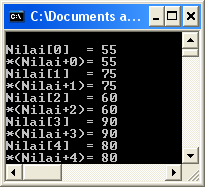
<tipe\_data> \*<nama\_variabel>

Contoh :

int \*ptrA;

**ptrA** adalah variabel pointer untuk tipe data integer

* 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 :

#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 ();

}

* **Coba program berikut dan amati untuk memahami kode pemrograman pointer :**

*/\**

*Nama File :pointer.c*

*\*/*

#include <stdio.h>

#include <conio.h>

void main()

{

int A;*//deklarasi var biasa*

int \*ptrA; *//deklarasi var pointer*

clrscr();

A=23;

ptrA=&A;

printf("\n A = %d",A); *//isi dari var A*

printf("\n&A = %u",&A); *//alamat memory var A*

printf("\n\*(&A) = %u",\*(&A)); *//isi di alamat pada var A*

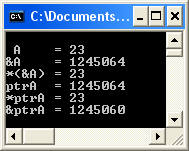
printf("\nptrA = %u",ptrA);*//isi dari var pointer ptrA*

printf("\n\*ptrA = %d",\*ptrA);*//isi di alamat pada var ptrA*

printf("\n&ptrA = %u",&ptrA);*//alamat memory var ptrA*

getch();

}



* Contoh program penerapan pointer untuk argumen fungsi call by reference :

*/\*Nama File : tukar2.c*

*Prinsip Call by reference*

*Oleh : \*/*

#include <stdio.h>

#include <conio.h>

void tukar(int \*a,int \*b)

{

int temp;

temp=\*a;

\*a=\*b;

\*b=temp;

}

void main()

{

clrscr();

int x=3, y=5;

printf("\nNilai x dan y semula");

printf("\nx = %d y = %d",x,y);

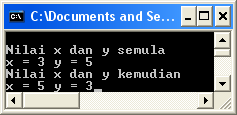
tukar(&x,&y);

printf("\nNilai x dan y kemudian");

printf("\nx = %d y = %d",x,y);

getch();

}



# MODUL III: String dan Password

1.Materi

* + String
  + Password

2.Tujuan

* + Mahasiswa dapatmelakukan Pengolahan datastring
  + Mahasiswa dapat menggunakan string dan fungsi input/output karakter
  + Mahasiswa dapat membuat password

3.Teori

* String adalah array dari tipe character yang diakhiri dengan null character (\0)
* Data konstan string ditulis dengan diapit oleh tanda petik dua (“…string…”)
* Character string konstan memiliki scope static yang berarti …………………………………………………………..
* 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>

void ubah\_nama(char \*nama)

{

nama=strcat(nama," Nasution");

}

void main()

{

char \*nm;

clrscr();

printf("\nMasukkan nama anda :");gets(nm);

ubah\_nama(nm);

printf("\nNama anda sekarang : %s",nm);

getch();

}

Cobakan program berikut :

#include <stdio.h>

#include <conio.h>

#include <string.h>

void main()

{

char username[25],password[12];

printf("User Name :");gets(username);

printf("Password :");gets(password);

if ((strcmp(username,"aku")==0) && (strcmp(password,"tif08")==0))

{

printf("Username dan password OK !!!\n");

}

else

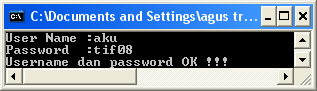
{

printf("Username Or password invalid !!!\n");

}

getch();

}

****

**Note :**

* + Perhatikan adanya penambahan header ***<string.h>***
  + Perhatikan bahwa untuk membandingkan dua string digunakan perintah ***strcmp(string1,string2),*** bukan operator “==” , jika strcmp() bernilai 0 ( strcmp()==0 ) berarti string tersebut sama.

Jalankan program dan coba isi ***username*** dengan : **aku** dan ***password*** dengan : **tif08**.

Coba juga isi dengan username dan password dengan data yang lain.

## Latihan :

1. Modifikasi program di atas sehingga dapat membedakan antara kesalahan pengisian username (***username invalid***) dan kesalahan pengisian password (***password invalid***) !
2. Tambahkan kode pada program di atas sehingga jika username atau password salah akan meminta mengulangi pengisi username dan password.

# MODUL IV: Sorting dan Array

1.Materi (lanjutan)

* + Sorting dan Array

2.Tujuan

* + Mahasiswa dapatmelakukan sorting data
  + Mahasiswa dapat menggunakan variable array untuk sorting

3.Teori

* Sorting berarti proses meng-urutkan. Ada dua jenis sorting yaitu ASCending (menaik) : urutan dari kecil ke besar, dan DESCending(menurun) : urutan dari besar ke kecil.
* Ada beberapa algoritma meng-urutkan, pada percobaan kali akan diberikan program sorting paling sederhana yang sering disebut dengan bubble sort. Perbedaan algoritma dalam hal ini bukan menentukan hasil sorting-nya tapi lebih menentukan efektivitas proses sorting apakah butuh waktu lama atau tidak. Algoritima sorting yang lain akan diberikan pada mata kuliah : struktur data.
* Pada percobaan kali ini yang paling utama untuk diperhatikan peserta praktikan adalah bagaimana variabel array digunakan untuk proses sorting tersebut.
* Selanjutnya lakukan percobaan dengan program-program berikut :

***//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\****

***//Program Percobaan menggunakan tipe Array***

***//Algoritma Descending Sorting***

***//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\****

#include <stdio.h>

#include <conio.h>

void main()

{

const N=10; ***//membuat konstanta, maksudnya N tidak boleh berubah(tetap=constant)***

int a[N]; ***//mendeklarasikan variavel tipe array***

int i,j,T;

***//Masukkan data ke var array looping for***

for(i=0;i<=N-1;i++)

{

printf("Masukkan data integer untuk a[%d] :",i);scanf("%d",&a[i]);

}

***//Data pada array Sebelum Sorting***

clrscr();

printf("\nData pada var a sebelum disorting :");

printf("\n===================================");

for(i=0;i<=N-1;i++)

{

printf("\nData pada var a[%d] : %d",i,a[i]);

}

printf("\n\nEnter dulu dong...!");getch();

***//proses sorting***

***//Perhatikan begaimana digunakan for dalam for (for bersarang = nested for)***

for(i=0;i<=N-2;i++)

{

for(j=i+1;j<=N-1;j++)

{

if (a[i]<=a[j])

{

T=a[i];

a[i]=a[j];

a[j]=T;

}

}

}

***//Data pada array Setelah Sorting***

printf("\n\nData pada var a setelah disorting :");

printf("\n===================================");

for(i=0;i<=N-1;i++)

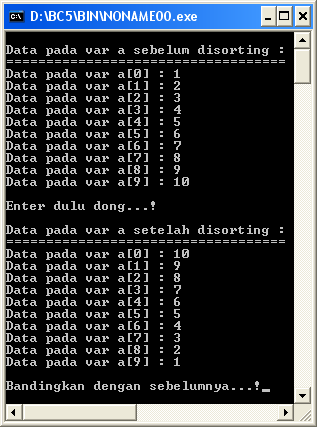
{

printf("\nData pada var a[%d] : %d",i,a[i]);

}

printf("\n\nBandingkan dengan sebelumnya...!");getch();

}***//akhir dari main()***



## Latihan :

* + Modifikasi program di atas untuk meng-ururtkan (sorting) secara Ascending
  + Buatlah program untuk meng-urutkan data tipe char !
  + Buatlah program untuk meng-urutkan data tipe string !

***- Selamat Berlatih -***

MODUL PRAKTIKUM

ALGORITMA PEMROGRAMAN II



UNIVERSITAS MADURA

FAKULTAS TEKNIK INFORMATIKA

2011-2012