Algoritma dan Struktur Data

Menganalisis dan Latihan soal kode program Bahasa C



# Oleh:

## Rangga Raisha Syaputra / 5223600007

**Program Studi STr Teknologi Game Departemen Teknologi Multimedia Kreatif Politeknik Elektronika Negeri Surabaya 2023/2024**

**Mengakses Elemen Struktur**

#include <stdio.h>

#include <string.h>

struct date {

int month;

int day;

int year;

};

struct student{

char name[30];

struct date birthday;

};

struct student mhs;

main()

{

strcpy(mhs.name, "MUHAMMAD IHSAN");

mhs.birthday.month = 8;

mhs.birthday.day = 10;

mhs.birthday.year = 1970;

printf("Name : %s\n", mhs.name);

printf("Birthday : %d-%d-%d\n",mhs.birthday.month,

mhs.birthday.day, mhs.birthday.year);

}

1. Dalam kode diatas terdapat dua struct yaitu: 'date' untuk merepresentasikan tanggal dengan bulan, hari, dan tahun, lalu 'struct student' untuk merepresentasikan seorang siswa dengan nama dan tanggal lahir.

2. Terdapat variabel dengan nama 'mhs' dan tipe data struct 'student'. Variabel ini untuk menyimpan informasi tentang seorang siswa.

3. perintah printf berfungsi untuk menampilkan isi dari struct.

**Menginisialisasi Struktur**

#include <stdio.h>

main()

{

struct zodiak {

char nama[11];

int tgl\_awal;

int bln\_awal;

int tgl\_akhir;

int bln\_akhir;

};

static struct zodiak bintang =

{"Sagitarius", 22, 11, 21, 12};

int tgl\_lhr, bln\_lhr, thn\_lhr;

printf("Masukkan tgl lahir Anda (XX-XX-XXXX): ");

scanf("%d-%d-%d",&tgl\_lhr, &bln\_lhr, &thn\_lhr);

if((tgl\_lhr >= bintang.tgl\_awal && bln\_lhr ==

bintang.bln\_awal) || (tgl\_lhr <= bintang.tgl\_akhir &&

bln\_lhr == bintang.bln\_akhir))

printf("Bintang Anda adalah %s\n", bintang.nama);

else

printf("Bintang Anda bukan %s\n", bintang.nama);

}

Program ini bertujuan untuk menentukan zodiak seseorang sagitarius atau bukan berdasarkan tanggal lahir yang dimasukkan.

1. Terdapat 'struct zodiak' untuk menyimpan beberapa field tanggal, bulan, dan tahun.

2. Variabel bintang diinisialisasi ke dalam 'struct zodiak'.

3. Program hanya terdapat 2 kemungkinan yakni, berzodiak sagitarius atau bukan.

4. User diminta untuk memasukkan tanggal lahir berisi tanggal bulan dan tahun.

**Array dan Struktur**

#include <stdio.h>

#define MAKS 20

struct date {

int month;

int day;

int year;

};

struct student {

char name[30];

struct date birthday;

};

struct student data\_mhs[MAKS];

main()

{

int i=0, sudah\_benar, jml;

char lagi;

do

{

printf("Name : ");

fgets(data\_mhs[i].name,sizeof data\_mhs[i].name,stdin);

printf("Birthday (mm-dd-yyyy): ");

scanf("%d-%d-%d",

&data\_mhs[i].birthday.month,

&data\_mhs[i].birthday.day,

&data\_mhs[i].birthday.year);

printf("\n");

i++;

printf("Mau memasukkan data lagi [Y/T] ? ");

do

{

lagi = getchar( );

sudah\_benar = (lagi == 'Y') || (lagi== 'y')||

(lagi == 'T') || (lagi == 't');

} while(! sudah\_benar);

fflush(stdin);

printf("\n");

} while(lagi == 'Y' || lagi == 'y');

jml = i;

printf("DATA SISWA\n");

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

{

printf("%d. Name : %s", i+1, data\_mhs[i].name);

printf(" Birthday : %d-%d-%d\n\n",

data\_mhs[i].birthday.month,

data\_mhs[i].birthday.day,

data\_mhs[i].birthday.year );

};

}

Kode program diatas berfungsi untuk menyimpan data siswa ke dalam array.

1. Terdapat dua struktur, 'date' dan 'student', didefinisikan untuk menyimpan tanggal dan data siswa.

2. Sebuah array dari tipe data struct 'student' dengan nama 'data\_mhs' dideklarasi sebagai variabel. Array ini akan menampung data siswa sebanyak 'MAKS' yang telah ditetapkan sebelumnya, yakni sebanyak 20.

3. Program menggunakan perulangan 'do-while' untuk meminta user memasukkan data siswa yang akan disimpan dalam array 'data\_mhs'.

4. Setelah itu data akan ditampilkan dari array 'data\_mhs' menggunakan program perulangan for.

**Melewatkan Elemen Struktur ke dalam Fungsi**

#include <stdio.h>

void cetak\_tanggal(int, int, int);

main()

{

struct date {

int month;

int day;

int year;

} today;

printf("Enter the current date (mm-dd-yyyy): ");

scanf("%d-%d-%d", &today.month, &today.day, &today.year);

cetak\_tanggal(today.month, today.day, today.year);

}

void cetak\_tanggal(int mm, int dd, int yy)

{

static char \*nama\_bulan[] = {

"Wrong month", "January", "February", "March",

"April", "May", "June", "July", "August",

"September", "October", "November", "December"

};

printf("Todays date is %s %d, %d\n\n",

nama\_bulan[mm],dd,yy);

}

Kode program diatas bertujuan untuk meminta user memasukkan tanggal dan kemudian di convert ke tanggal tersebut dalam format yang lebih bersahabat dengan menggunakan nama bulan.

1. User diminta untuk memasukkan tanggal saat ini melalui input yang dibaca oleh 'scanf' dan disimpan dalam variabel 'today'.

2. 'cetak\_tanggal' berfungsi untuk mengirimkan argumen bulan '(today.month)', hari '(today.day)', dan tahun '(today.year)' lalu mencetak tanggal dengan array statis 'nama\_bulan'.

**Melewatkan Struktur ke dalam Fungsi**

struct date {

int month;

int day;

int year;

};

void cetak\_tanggal(struct date);

main()

{

struct date today;

printf("Enter the current date (mm-dd-yyyy): ");

scanf("%d-%d-%d", &today.month, &today.day, &today.year);

cetak\_tanggal(today);

}

void cetak\_tanggal(struct date now)

{

static char \*nama\_bulan[] = {

"Wrong month", "January", "February", "March",

"April", "May", "June", "July", "August",

"September", "October", "November", "December"

};

printf("Todays date is %s %d, %d\n\n",

nama\_bulan[now.month], now.day, now.year);

}

Kode program ini sama dengan kode program sebelumnya, namun parameter fungsi diubah menjadi bentuk struktur, sehingga parameter fungsi tidak lagi sebanyak tiga buah, melainkan hanya satu.

1. 'struct date' digunakan untuk menyimpan data tanggal, bulan, dan tahun.

2. 'cetak\_tanggal()' menerima argumen berupa struktur 'date' dengan nama 'now'. Fungsi ini mencetak tanggal dengan menggunakan array statis 'nama\_bulan' untuk mendapatkan nama bulan yang sesuai dengan nilai 'now.month'.

**Struktur dan Pointer (Pointer ke Struktur)**

#include <stdio.h>

struct koordinat

{ int x;

int y;

};

void tukar\_xy(struct koordinat \*);

main()

{

struct koordinat posisi;

printf("Masukkan koordinat posisi (x, y) : ");

scanf("%d, %d", &posisi.x, &posisi.y);

printf("x, y semula = %d, %d\n", posisi.x, posisi.y);

tukar\_xy(&posisi);

printf("x, y sekarang = %d, %d\n", posisi.x, posisi.y);

}

void tukar\_xy(struct koordinat \*pos\_xy)

{

int z;

z = (\*pos\_xy).x;

(\*pos\_xy).x = (\*pos\_xy).y;

(\*pos\_xy).y = z;

}

1. Sebuah struktur bernama koordinat didefinisikan untuk menyimpan dua nilai bertipe integer, yaitu x dan y.

2. void tukar\_xy(struct koordinat \*); Kode ini dideklarasikan untuk menukar nilai x dan y dari suatu struct koordinat. Fungsi ini menggunakan pointer ke struktur koordinat sebagai argumen.

3. 'printf("Masukkan koordinat posisi (x, y) : ");

scanf("%d, %d", &posisi.x, &posisi.y);'

Pada kode ini user diminta untuk memasukkan nilai x dan y untuk koordinat posisi.

4. Lalu nilai x dan y ditukar menggunakan kode:

'tukar\_xy(&posisi);'.

5. Setelah itu nilai x dan y ditampilkan dengan kode 'printf("x, y sekarang = %d, %d\n", posisi.x, posisi.y);'.

**KODE 7**

#include <stdio.h>

void tukar\_xy(int \*, int \*);

main()

{

struct koordinat {

int x;

int y;

} posisi;

printf("Masukkan koordinat posisi (x, y) : ");

scanf("%d, %d", &posisi.x, &posisi.y);

printf("x, y semula = %d, %d\n", posisi.x, posisi.y);

tukar\_xy(&posisi.x, &posisi.y);

printf("x, y sekarang = %d, %d\n", posisi.x, posisi.y);

}

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

{

int z;

z = \*a;

\*a = \*b;

\*b = z;

}

Sebenarnya kode tersebut dengan kokde sebelumnya memiliki tujuan yang sama, yaitu menukar nilai x dan y dari suatu struktur atau variabel.

**Latihan Soal**

1. Definisikan sebuah struktur (misalkan namanya = record) yang memiliki 3 buah field berupa sebuah integer (misalkan namanya = loop), sebuah array karakter dengan 5 elemen (misalkan namanya = word) dan sebuah float (misalkan Namanya = sum).
2. Deklarasikan sebuah variabel struktur (misalkan namanya = sample) yang didefinisikan memiliki tipe struktur record.
3. Masukkan nilai 10 kepada field loop dari struktur sample yang bertipe struktur record tsb.
4. Tampilkan ke layar (menggunakan fungsi printf()) string yang tersimpan dalam array word dari struktur sample.
5. Definisikan sebuah struktur (misalkan namanya = date) yang memiliki 3 field bertipe int (misalkan namanya = day, month dan year). Kemudian tuliskan potongan program untuk memasukkan 5 buah tanggal yang disimpan dalam sebuah array struktur yang bertipe date.
6. #include <stdio.h>

#include <string.h>

struct record {

int loop;

char word[5];

float sum;

};

int main() {

return 0;

}

1. #include <stdio.h>

#include <string.h>

struct record {

int loop;

char word[5];

float sum;

};

int main() {

struct record sample;

sample.loop = 10;

return 0;

}

1. #include <stdio.h>

#include <string.h>

struct record {

int loop;

char word[5];

float sum;

};

int main() {

struct record sample;

sample.loop = 10;

printf("string dalam array word adalah %s", sample.word);

return 0;

}

1. #include <stdio.h>

#include <string.h>

struct record {

int loop;

char word[5];

float sum;

};

int main() {

struct record sample;

sample.loop = 10;

printf("string dalam array word adalah %s", sample.word);

return 0;

}

1. #include <stdio.h>

struct date {

int day;

int month;

int year;

};

int main() {

struct date tanggal[5];

for (int i = 0; i < 5; ++i) {

printf("Masukkan tanggal ke-%d (dd mm yyyy): ", i + 1);

scanf("%d %d %d", &tanggal[i].day, &tanggal[i].month, &tanggal[i].year);

}

printf("\nTanggal yang dimasukkan:\n");

for (int i = 0; i < 5; ++i) {

printf("%d: %02d-%02d-%d\n", i + 1, tanggal[i].day, tanggal[i].month, tanggal[i].year);

}

return 0;

}