

TUGAS PRAKTIKUM STRUKTUR DATA 1

PROGRAM GAME



Oleh :

Putu Asri Sri Sutanti

(1608561030)

PROGRAM STUDI TEKNIK INFORMATIKA

FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM

UNIVERSITAS UDAYANA

BALI

2017

BAB I

LANDASAN TEORI

1.1 ARRAY

Dalam beberapa literatur, array sering disebut (diterjemahkan) sebagai larik. Array adalah kumpulan dari nilai-nilai data bertipe sama dalam urutan tertentu yang menggunakan sebuah nama yang sama. Nilai-nilai data di suatu array disebut dengan elemen-elemen array. Letak urutan dari elemen-elemen array ditunjukkan oleh suatu subscript atau indeks. Array bisa berupa array berdimensi satu, dua, tiga atau lebih. Array berdimensi satu (one-dimensional array) mewakili bentuk suatu vektor. Array berdimensi dua (twodimensional array) mewakili bentuk dari suatu matriks atau table. Array berdimensi tiga (three-dimensional array) mewakili bentuk suatu ruang.

1.1 Array Berdimensi Satu

Suatu array berdimensi satu dideklarasikan dalam bentuk umum berupa :

`tipe_data nama_var[ukuran];`

dengan :

- tipe_data : untuk menyatakan tipe dari elemen array, misalnya *int*, *char*, *float*.
- nama_var : nama variabel array
- ukuran : untuk menyatakan jumlah maksimal elemen array.

1.2 Array Berdimensi Dua

Array berdimensi satu dapat disimpan pada sebuah array berdimensi dua.

Pendeklarasian array berdimensi dua adalah sebagai berikut :

```
int data_lulus[4][3];
```

Nilai 4 untuk menyatakan banyaknya baris dan 3 menyatakan banyaknya kolom. Array seperti **data_lulus** dapat diakses dalam bentuk `data_lulus[indeks pertama, indeks kedua]` :

(1) `data_lulus[0][1] = 540;`

merupakan instruksi untuk memberikan nilai 540 ke array **data_lulus** untuk indeks pertama = 0 dan indeks kedua bernilai 1.

(2) `printf("%d",data_lulus[2][0]);`

merupakan perintah untuk menampilkan elemen yang memiliki indeks pertama = 2 dan indeks kedua = 0.

1.3 Array Berdimensi Banyak.

C memungkinkan untuk membuat array yang dimensinya lebih dari dua. Bentuk umum pendeklarasian array berdimensi banyak :

`tipe nama_var[ukuran 1][ukuran2}...[ukuranN];`

sebagai contoh :

`int data_huruf[2][8][8];`

Merupakan pendeklarasian array **data_huruf** sebagai array berdimensi tiga.

1.2STRUCT

Struktur adalah koleksi dari variabel yang dinyatakan dengan sebuah nama, dengan sifat setiap variabel dapat memiliki tipe yang berlainan. Struktur dapat digunakan untuk mengelompokkan beberapa informasi yang berkaitan menjadi sebuah satu kesatuan. Bentuk umum deklarasi struktur adalah sebagai berikut:

```
struct nama_tipe_struktur
{
    tipe field1;
    .
    .
    tipe fieldN;
} variabel_struktur1, ..., variabel_strukturM;
```

Elemen dari struktur dapat diakses dengan menggunakan bentuk:

`variabel_struktur.nama_field`

1.3ARRAY DAN STRUCT

Elemen-elemen dari suatu array juga dapat berbentuk sebuah struktur. Misalnya array yang dipakai untuk menyimpan sejumlah data siswa (*struct student*). Array struktur berdimensi satu ini membentuk suatu tabel, dengan barisnya menunjukkan elemen dari array-nya dan kolomnya menunjukkan elemen dari struktur.

BAB II

PEMBAHASAN

2.1 SOURCE CODE

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

int pil1,pil2, i,sems;
    int pemain1,pemain2;
struct main
{
    int power;
    int attack;
    int defence;

};
struct main p[50];

void pembuka()
{
    i = 0;
    while (i<5)
    {
        p[i].power=200;
        i++;
    }
    p[0].attack=50;p[1].attack=40;p[2].attack=60;p[3].attack=50;p[4].attack=30;
    p[0].defence=28;p[1].defence=45;p[2].defence=48;p[3].defence=29;p[4].defence=56;
    printf("\t\t\tSELAMAT DATANG DI GAME PRAKTIKUM 1\n\n");
    printf("Tabel Karakter Beserta Dengan Atributnya :\n");
    printf(" =====\n");
    printf("|          | A    | B    | C    | D    | E    |\n");
    printf(" =====\n");
    printf("| Power   | %d   | %d   | %d   | %d   | %d   |\n",p[0].power,p[1].power,p[2].power,p[3].power,p[4].power);
    printf("| Attack  | %d   | %d   | %d   | %d   | %d   |\n",p[0].attack,p[1].attack,p[2].attack,p[3].attack,p[4].attack);
```

```

    printf("| Defence| %d    | %d    | %d    | %d    | %d\n",p[0].defence,p[1].defence,p[2].defence,p[3].defence,p[4].defence);
    printf(" =====\n");
}
int main()
{
    pembuka();
    printf("Pilih Pemain Yang Ingin Digunakan : ");
    printf("\nPemain Pertama : "); scanf("%d",&pemain1);
    printf("\nPemain Kedua : "); scanf("%d",&pemain2);

    pemain1 = pemain1-1 ;//karena indeks dari pemain1 dimulai dari 0
    pemain2 = pemain2 -1 ;//karena indeks dari pemain2 dimulai dari 0

    while (p[pemain2].power > 0 && p[pemain1].power > 0)
    {
        system("cls");
        printf("\n=====\\n");
        printf("|          | Pemain1   | Pemain2   |\\n");
        printf(" =====\\n");
        printf("| Power   |      %d      |      %d\n",p[pemain1].power,p[pemain2].power);
        printf("| Attack |      %d      |      %d\n",p[pemain1].attack,p[pemain2].attack);
        printf("| Defence|      %d      |      %d\n",p[pemain1].defence,p[pemain2].defence);
        printf(" =====\\n");
        printf("Apakah Anda Ingin : \\n\\t1. SERANG \\n\\t2. BERTAHAN\\n");
        printf("\nPemain1 : "); scanf("%d",&pil1);
        p[pemain1].power=p[pemain1].power-10;
        if(pil1==1)
        {
            printf("\nPemain Pertama memilih untuk menyerang !!");
        }
        else if (pil1==2)
        {
            printf("\nPemain Pertama memilih untuk bertahan !!");
        }
        else
        {
            printf("\nInputan Salah !");
            system("cls");
            getch();
            main();
        }
        printf("\nPemain2 :"); scanf("%d",&pil2);
        p[pemain2].power=p[pemain2].power-10;
        if(pil2==1)
        {
            printf("\nPemain Pertama memilih untuk menyerang !!");
        }
        else if (pil2==2)
        {
            printf("\nPemain Pertama memilih untuk bertahan !!");
        }
        else
        {

```

```

        printf("\nInputan Salah !");
        system("cls");
        getch();
        main();
    }
    if(pil1==1 && pil2==1)
    {
        if(p[pemain1].attack > p[pemain2].defence)
        {
            sems = p[pemain1].attack - p[pemain2].defence;
            p[pemain2].power = p[pemain2].power-sems;
        }
        else if(p[pemain2].attack > p[pemain1].defence)
        {
            sems = p[pemain2].attack - p[pemain1].defence;
            p[pemain1].power = p[pemain1].power-sems;
        }
    }
    else if(pil1==1 && pil2==2)
    {
        if(p[pemain1].attack > p[pemain2].defence)
        {
            sems = p[pemain1].attack - p[pemain2].defence;
            p[pemain2].power = p[pemain2].power-sems;
        }
    }
    else if(pil1==2 && pil2==1)
    {
        if(p[pemain2].attack > p[pemain1].defence)
        {
            sems = p[pemain2].attack - p[pemain1].defence;
            p[pemain1].power = p[pemain1].power-sems;
        }
    }
}

if(p[pemain1].power > p[pemain2].power)
{
    printf("\n\n\n\t\t\tPEMAIN 1 MENANG !!");
}
else if(p[pemain1].power<p[pemain2].power)
{
    printf("\n\n\n\t\t\tPEMAIN 2 MENANG !!");
}
else if(p[pemain1].power=p[pemain2].power)
{
    printf("\n\n\n\t\t\tPERMAINAN SERI");
}

return 0;;
}

```

Penjelasan :

Dalam program ini, saya menggunakan struct dan array, program ini merupakan program game dimana di dalam program ini user dapat melakukan serangan dan bertahan

saat program berjalan selain itu program ini berisi fitur power dimana power ini merupakan nyawa dan dapat menentukan kalah menangnya user.

Untuk menjalankan program ini, saya menggunakan algoritma seperti berikut :

```
if(pil1==1 && pil2==1)
{
    if(p[pemain1].attack > p[pemain2].defence)
    {
        sems = p[pemain1].attack - p[pemain2].defence;
        p[pemain2].power = p[pemain2].power-sems;
    }
    else if(p[pemain2].attack > p[pemain1].defence)
    {
        sems = p[pemain2].attack - p[pemain1].defence;
        p[pemain1].power = p[pemain1].power-sems;
    }
}
else if(pil1==1 && pil2==2)
{
    if(p[pemain1].attack > p[pemain2].defence)
    {
        sems = p[pemain1].attack - p[pemain2].defence;
        p[pemain2].power = p[pemain2].power-sems;
    }
}
else if(pil1==2 && pil2==1)
{
    if(p[pemain2].attack > p[pemain1].defence)
    {
        sems = p[pemain2].attack - p[pemain1].defence;
        p[pemain1].power = p[pemain1].power-sems;
    }
}
```

Source code diatas merupakan beberapa kondisi yang akan terjadi saat program dijalankan. Source di bawah ini, merupakan source code saat kondisi dimana pemain 1 memilih untuk menyerang dan pemain 2 juga memilih untuk menyerang. Di kondisi ini juga terdapat kondisi di dalamnya yaitu kondisi saat nilai serang dari pemain 1 lebih besar dari nilai defence dari pemain 2 dan juga saat nilai serang pemain 2 lebih besar dari nilai defence pemain 1.

```
if(pil1==1 && pil2==1)
```

```

{
    if(p[pemain1].attack > p[pemain2].defence)
    {
        sems = p[pemain1].attack - p[pemain2].defence;
        p[pemain2].power = p[pemain2].power-sems;
    }
    else if(p[pemain2].attack > p[pemain1].defence)
    {
        sems = p[pemain2].attack - p[pemain1].defence;
        p[pemain1].power = p[pemain1].power-sems;
    }
}

```

Selanjutnya, ada kondisi kedua yaitu kondisi saat pemain 1 memilih untuk menyerang dan pemain 2 memilih untuk bertahan, disini juga terdapat kondisi di dalamnya yaitu saat nilai serang dari pemain 1 lebih besar dari nilai defene pemain 2 sehingga dapat mempengaruhi power dari pemain 2.

```

else if(pil1==1 && pil2==2)
{
    if(p[pemain1].attack > p[pemain2].defence)
    {
        sems = p[pemain1].attack - p[pemain2].defence;
        p[pemain2].power = p[pemain2].power-sems;
    }
}

```

Untuk kondisi yang ketiga yaitu kondisi kebalikan dari kondisi ke dua dimana pemain 1 memilih untk bertahan dan pemain 2 memilih untuk menyerang. Di kondisi ini juga terdapat kondisi di dalamnya yaitu kondisi saat nilai serang pada pemain 2 lebih besar dari nilai defence pemain 1.

```

else if(pil1==2 && pil2==1)
{
    if(p[pemain2].attack > p[pemain1].defence)
    {
        sems = p[pemain2].attack - p[pemain1].defence;
        p[pemain1].power = p[pemain1].power-sems;
    }
}

```

2.2RUN PROGRAM

SELAMAT DATANG DI GAME PRAKTIKUM 1

Tabel Karakter Beserta Dengan Atributnya :

	A	B	C	D	E
Power	200	200	200	200	200
Attack	50	40	60	50	30
Defence	28	45	48	29	56

Pilih Pemain Yang Ingin Digunakan [A/B/C/D] :

Pemain Pertama : 1

Pemain Kedua : 1

	Pemain1	Pemain2
Power	116	116
Attack	50	50
Defence	28	28

Apakah Anda Ingin :

1. SERANG
2. BERTAHAN

Pemain1 :

PERMAINAN SERI

Process exited after 62.69 seconds with return value 0
Press any key to continue . . .