

LAPORAN TUGAS I GAME HERO
PRAKTIKUM STRUKTUR DATA



I MADE WARDANA

NIM. 1608561029

PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS UDAYANA
BUKIT JIMBARAN

2017

DAFTAR ISI

Halaman Sampul.....	1
Daftar Isi	2
BAB I LANDASAN TEORI.....	3
1.1 Struktur Data.....	3
1.2 Bahasa Pemrograman C dan Strukturnya	3
1.3 Code::Blocks.....	6
1.4 Permainan Strategi Berbasis Giliran	6
BAB II PERMASALAHAN.....	7
2.1 Program <i>Game Hero</i>	7
BAB III PEMBAHASAN.....	8
3.1 Source Code Program	8
3.2 Penjelasan Singkat Program	21
3.3 Tampilan Running Program	22

BAB I

LANDASAN TEORI

1.1 Struktur Data

Dalam istilah ilmu komputer, sebuah struktur data adalah cara penyimpanan, penyusunan dan pengaturan data di dalam media penyimpanan komputer sehingga data tersebut dapat digunakan secara efisien.

Dalam teknik pemrograman, struktur data berarti tata letak data yang berisi kolom-kolom data, baik itu kolom yang tampak oleh pengguna (user) ataupun kolom yang hanya digunakan untuk keperluan pemrograman yang tidak tampak oleh pengguna. Setiap baris dari kumpulan kolom-kolom tersebut dinamakan catatan (record). Lebar kolom untuk data dapat berubah dan bervariasi. Ada kolom yang lebarnya berubah secara dinamis sesuai masukan dari pengguna, dan juga ada kolom yang lebarnya tetap. Dengan sifatnya ini, sebuah struktur data dapat diterapkan untuk pengolahan database (misalnya untuk keperluan data keuangan) atau untuk pengolah kata (word processor) yang kolomnya berubah secara dinamis. Contoh struktur data dapat dilihat pada berkas-berkas lembar-sebar (spreadsheet), pangkal-data (database), pengolahan kata, citra yang dipampat (dikompres), juga pemampatan berkas dengan teknik tertentu yang memanfaatkan struktur data.

1.2 Bahasa Pemrograman C dan Strukturnya

Bahasa C merupakan perkembangan dari bahasa BCPL yang dikembangkan oleh Martin Richards pada tahun 1967, Selanjutnya bahasa ini memberikan ide kepada Ken Thompson yang mengembangkan bahasa yang disebut bahasa B pada tahun 1970. Perkembangan selanjutnya dari bahasa B adalah bahasa C oleh Dennis Ritchie sekitar tahun 1970-an di Bell Telephone Laboratories Inc (sekarang adalah AT&T Bell Laboratories). Bahasa C pertama kali digunakan di komputer Digital Equipment Corporation PDP-11 yang menggunakan system operasi UNIX.

Hingga saat ini penggunaan bahasa C telah merata di seluruh dunia. Hampir semua perguruan tinggi di dunia menjadikan bahasa C sebagai salah satu mata kuliah wajib. Selain itu, banyak bahasa pemrograman populer seperti PHP dan Java

menggunakan sintaks dasar yang mirip bahasa C. Oleh karena itu, kita juga sangat perlu mempelajarinya. Menguasai bahasa pemrograman C adalah langkah dasar untuk menguasai bahasa pemrograman lainnya.

Dalam bahasa pemrograman C, keyword dan function ditampilkan dalam bentuk tulisan monofont, seperti `printf()` dan `break`. Sedangkan nama file ditampilkan dalam tipe monospace, seperti `program.exe`.

Berikut ini struktur dalam bahasa pemrograman C yang digunakan pada program game hero.

1.3.1 Library

Library `stdio.h` dan `stdlib.h` adalah beberapa dari library yang sering digunakan pada program bahasa C. Library `stdio.h` adalah library pada bahasa C yang digunakan untuk operasi input output (`stdio` = Standard Input and Output). Tanpa menggunakan library ini maka perintah-perintah input/output tidak dapat dieksekusi atau dijalankan. Library `stdlib.h` mendefinisikan fungsi konversi numerik, pseudo-acak nomor fungsi generasi, alokasi memori, fungsi kontrol proses. Library `stdlib.h` merupakan library yang hampir sama dengan library `math.h`, namun isi dari library `stdlib.h` berbeda dengan library `math.h`.

1.3.2 Fungsi Utama

Fungsi merupakan bagian yang terpisah dari program dan dapat diaktifkan atau dipanggil di manapun di dalam program. Fungsi dalam bahasa C ada yang sudah disediakan sebagai fungsi pustaka seperti `printf()`, `scanf()`, `getch()` dan untuk menggunakannya tidak perlu dideklarasikan. Fungsi yang perlu dideklarasikan terlebih dahulu adalah fungsi yang dibuat oleh programmer.

1.3.3 Konstanta

Konstanta merupakan suatu nilai yang tidak dapat diubah selama proses program berlangsung. Konstanta nilainya selalu tetap. Konstanta harus didefinisikan terlebih dahulu di awal program. Konstanta dapat bernilai integer, pecahan, karakter dan string. Contoh konstanta yaitu 50, 3.14, 'A', dan 'Bahasa C'.

1.3.4 Tipe Data

Data merupakan suatu nilai yang bisa dinyatakan dalam bentuk konstanta atau variabel. Konstanta menyatakan nilai yang tetap, sedangkan variabel menyatakan nilai yang dapat diubah-ubah selama eksekusi berlangsung. Data berdasarkan jenisnya dapat dibagi menjadi lima kelompok, yang dinamakan sebagai tipe data dasar. Kelima tipe data dasar adalah bilangan bulat (integer), bilangan real presisi-tunggal, bilangan real presisi-ganda, karakter, dan variabel tak-bertipe (void).

No.	Tipe Data	Ukuran	Range (Jangkauan)	Format	Keterangan
1.	Char	1 byte	-128 s/d 127	%c	Karakter
2.	Int	2 byte	-32768 s/d 32767	%i, %d	Bilangan bulat
3.	Float	4 byte	-3.4E-38 s/d 3.4E+38	%f	Bilangan pecahan
4.	Double	8 byte	1.7E-308 s/d 1.7E+308	%lf	Pecahan presisi ganda
5.	Void	0 byte	-	-	Tidak bertipe

Tabel 1: Jenis tipe data pada bahasa C

1.3.5 Variabel

Variabel adalah suatu pengenal (identifier) yang digunakan untuk mewakili suatu nilai tertentu di dalam proses program. Berbeda dengan konstanta yang nilainya selalu tetap, nilai dari suatu variabel bisa diubah-ubah sesuai kebutuhan. Nama dari suatu variabel dapat ditentukan sendiri oleh pemrogram dengan beberapa aturan.

Nama variabel dapat terdiri dari gabungan huruf dan angka dengan karakter pertama harus berupa huruf. Bahasa C bersifat case-sensitive artinya huruf besar dan kecil dianggap berbeda. Nama variabel tidak boleh mengandung spasi dan simbol-simbol khusus, kecuali garis bawah (underscore). Panjang nama variabel bebas, tetapi hanya 32 karakter pertama yang terpakai.

1.3.7 Operator

Bahasa C menyediakan lima operator aritmatika, yaitu “*” (untuk perkalian), “/” (untuk pembagian), “%” (untuk sisa pembagian (modulus)), “+” (untuk penjumlahan), dan “-” (untuk pengurangan). Selain itu juga terdapat operator hubungan yang digunakan untuk membandingkan hubungan antara dua buah operand, sebuah nilai, atau variabel.

1.3.6 Komentar

Komentar program hanya diperlukan untuk memudahkan pembacaan dan pemahaman suatu program (untuk keperluan dokumentasi program). Dengan kata lain, komentar program hanya merupakan keterangan atau penjelasan program. Untuk memberikan komentar atau penjelasan dalam bahasa C digunakan pembatas `/*` dan `*/` atau menggunakan tanda `//` untuk komentar yang hanya terdiri dari satu baris. Komentar program tidak akan ikut diproses dalam program (akan diabaikan).

1.3 Code::Blocks

Cara yang mudah dan profesional untuk meng-coding dewasa ini adalah dengan Integrated Development Environment – atau sering disebut IDE, yang menggabungkan semua tools yang dibutuhkan untuk pemrograman menjadi satu kesatuan program yang kompak. IDE dapat digunakan untuk menulis kode, membangun program, men-debug program, dan sebagainya. IDE juga memecahkan berbagai masalah pengaturan compiler, mengkonfigurasi editor teks, dan mendapatkan semua elemen berbeda untuk bekerja secara bersama-sama.

Dewasa ini, banyak tersedia berbagai macam IDE di internet – dan semuanya memiliki kelebihan masing-masing. Code::Blocks hadir dengan fitur multiplatform-nya; software tersebut bisa berjalan baik di Windows, Mac OS X, maupun Linux. Selain itu, Code::Blocks juga memiliki banyak fitur lainnya yang tak kalah dengan IDE lainnya.

1.4 Permainan Strategi Berbasis Giliran

Permainan strategi berbasis giliran (bahasa Inggris: *turn-based strategy*, atau TBS) adalah permainan strategi (biasanya jenis permainan perang) di mana pemain bermain bergiliran. Permainan jenis ini berbeda dengan *real time strategy* di mana pemain bermain secara simultan. Dalam hal ini, permainan hero termasuk ke dalam *genre* permainan strategi berbasis giliran.

BAB II

PERMASALAHAN

2.1 Program Game Hero

Program ini menyimulasikan sebuah permainan bertemakan RPG sederhana berbasis teks, dengan ketentuan setiap pemain mempunyai nilai *power*, *defense*, dan *attack* masing-masing.

BAB III

PEMBAHASAN

3.1 Source Code Program

Berikut ini *source code* program *game hero*.

```
#include <stdio.h>
#include <stdlib.h>
#include <conio.h>
#include <windows.h>
#include <time.h>

struct coordinate{
    int x;
    int y;
    int direction;
};

void gotoxy(int x, int y)
{
    COORD coord;
    coord.X = x;
    coord.Y = y;
    SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE),
coord);
}

void AA()
{
    printf("Angga\n");
}

void BB()
{
    printf("Bobby\n");
}

void CC()
{
    printf("Chandra\n");
}

void DD()
{
    printf("Dharma\n");
}

void EEE()
{
    printf("Ega\n");
}

void FF()
{

```



```

    printf("Frisca\n");
}

void GG()
{
    printf("Goenk Rama\n");
}

void ZZ()
{
    printf("Hendra\n");
}

void II()
{
    printf("Rio\n");
}

void SS()
{
    printf("Tangkas\n");
}

void A()
{
    gotoxy(0,4);
    printf("Attack : 50");
    gotoxy(0,5);
    printf("Defend : 30");
}

void B()
{
    gotoxy(0,4);
    printf("Attack : 60");
    gotoxy(0,5);
    printf("Defend : 40");
}

void C()
{
    gotoxy(0,4);
    printf("Attack : 60");
    gotoxy(0,5);
    printf("Defend : 10");
}

void D()
{
    gotoxy(0,4);
    printf("Attack : 50");
    gotoxy(0,5);
    printf("Defend : 20");
}

void E()
{
    gotoxy(0,4);
    printf("Attack : 10");
    gotoxy(0,5);

```

```

        printf("Defend : 90");
    }

void F()
{
    gotoxy(0,4);
    printf("Attack : 40");
    gotoxy(0,5);
    printf("Defend : 60");
}

void G()
{
    gotoxy(0,4);
    printf("Attack : 20");
    gotoxy(0,5);
    printf("Defend : 40");
}

void Z()
{
    gotoxy(0,4);
    printf("Attack : 70");
    gotoxy(0,5);
    printf("Defend : 10");
}

void I()
{
    gotoxy(0,4);
    printf("Attack : 30");
    gotoxy(0,5);
    printf("Defend : 50");
}

void S()
{
    gotoxy(0,4);
    printf("Attack : 20");
    gotoxy(0,5);
    printf("Defend : 40");
}

void EA()
{
    gotoxy(0,9);
    printf("Attack : 50");
    gotoxy(0,10);
    printf("Defend : 30");
}

void EB()
{
    gotoxy(0,9);
    printf("Attack : 60");
    gotoxy(0,10);
    printf("Defend : 40");
}

void EC()
{

```

```

        gotoxy(0,9);
        printf("Attack : 60");
        gotoxy(0,10);
        printf("Defend : 10");
    }

void ED()
{
    gotoxy(0,9);
    printf("Attack : 50");
    gotoxy(0,10);
    printf("Defend : 20");
}

void EE()
{
    gotoxy(0,9);
    printf("Attack : 10");
    gotoxy(0,10);
    printf("Defend : 90");
}

void EF()
{
    gotoxy(0,9);
    printf("Attack : 40");
    gotoxy(0,10);
    printf("Defend : 60");
}

void EG()
{
    gotoxy(0,9);
    printf("Attack : 20");
    gotoxy(0,10);
    printf("Defend : 40");
}

void EZ()
{
    gotoxy(0,9);
    printf("Attack : 70");
    gotoxy(0,10);
    printf("Defend : 10");
}

void EI()
{
    gotoxy(0,9);
    printf("Attack : 30");
    gotoxy(0,10);
    printf("Defend : 50");
}

void ES()
{
    gotoxy(0,9);
    printf("Attack : 20");
    gotoxy(0,10);
    printf("Defend : 40");
}

```

```

int main()
{
    system("cls");
    int p,r,q,HEA=200,ATT=0,DEF=0,hea=200,att=0,def=0,pl,pil,rn;
    printf("\t| Monster | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |\n");
    printf("\t|   Health   |200|200|200|200|200|200|200|200|200|200|\n");
    printf("\t| Attack   |50 |60 |60 |50 |10 |40 |20 |70 |30 |20  |\n");
    printf("\t| Defend    |30 |40 |10 |20 |90 |60 |40 |10 |50 |40 |\n\n");
    printf("1. Angga\n");
    printf("2. Bobby\n");
    printf("3. Chandra\n");
    printf("4. Dharma\n");
    printf("5. Ega\n");
    printf("6. Frisca\n");
    printf("7. Goenk Rama\n");
    printf("8. Hendra\n");
    printf("9. Rio\n");
    printf("10. Tangkas\n\n");
    printf("Pilih Monster: ");
    scanf("%d",&p);
    srand(time(NULL));
    system("cls");
    switch(p)
    {
        case 1: AA(); break;
        case 2: BB(); break;
        case 3: CC(); break;
        case 4: DD(); break;
        case 5: EEE(); break;
        case 6: FF(); break;
        case 7: GG(); break;
        case 8: ZZ(); break;
        case 9: II(); break;
        case 10: SS(); break;
        default: return main(); break;
        break;
    }
    for(q=0;q<=250000000;q++);
    printf("    VS\n");
    for(q=0;q<=250000000;q++);
    ER:
    r=rand()%11;
    switch(r)
    {
        case 1: AA(); break;
        case 2: BB(); break;
        case 3: CC(); break;
        case 4: DD(); break;
        case 5: EEE(); break;
        case 6: FF(); break;
        case 7: GG(); break;
        case 8: ZZ(); break;
        case 9: II(); break;
        case 10: SS(); break;
        default: goto ER; break;
    }
    for(q=0;q<=250000000;q++);
    system("cls");
}

```

```

gotoxy(25,5);
printf("MULAI!!!");
for(q=0;q<=250000000;q++);
while (HEA>0&&hea>0)
{
    system("cls");
    printf("\t\t<PROGRAM GAME HERO>");
    gotoxy(0,2);
    switch(p)
    {
        case 1: AA(); break;
        case 2: BB(); break;
        case 3: CC(); break;
        case 4: DD(); break;
        case 5: EEE(); break;
        case 6: FF(); break;
        case 7: GG(); break;
        case 8: ZZ(); break;
        case 9: II(); break;
        case 10: SS(); break;
        break;
    }
    gotoxy(0,3);
    printf("Health : %d",HEA);
    switch(p)
    {
        case 1: A(); ATT=50; DEF=30; break;
        case 2: B(); ATT=60; DEF=40;break;
        case 3: C(); ATT=60; DEF=10;break;
        case 4: D(); ATT=50; DEF=20;break;
        case 5: E(); ATT=10; DEF=90;break;
        case 6: F(); ATT=40; DEF=60;break;
        case 7: G(); ATT=20; DEF=40;break;
        case 8: Z(); ATT=70; DEF=10;break;
        case 9: I(); ATT=30; DEF=50;break;
        case 10: S(); ATT=20; DEF=40;break;
        break;
    }
    gotoxy(0,7);
    switch(r)
    {
        case 1: AA(); break;
        case 2: BB(); break;
        case 3: CC(); break;
        case 4: DD(); break;
        case 5: EEE(); break;
        case 6: FF(); break;
        case 7: GG(); break;
        case 8: ZZ(); break;
        case 9: II(); break;
        case 10: SS(); break;
        break;
    }
    gotoxy(0,8);
    printf("Health : %d",hea);
    switch(r)
    {
        case 1: EA(); att=50; def=30; break;
        case 2: EB(); att=60; def=40;break;
        case 3: EC(); att=60; def=10;break;
        case 4: ED(); att=50; def=20;break;

```

```

        case 5: EE(); att=10; def=90;break;
        case 6: EF(); att=40; def=60;break;
        case 7: EG(); att=20; def=40;break;
        case 8: EZ(); att=70; def=10;break;
        case 9: EI(); att=30; def=50;break;
        case 10: ES(); att=20; def=40;break;
        break;
    }
    gotoxy(0,12);
    printf("ATTACK/DEFENDS (1/0) : ");
    scanf("%d",&pl);
    gotoxy(0,12);
    printf("                                ");
    gotoxy(0,12);
    {
    }
    rn=rand()%2;
    if(pl==0&&rn==0)
    {
        switch(p)
        {
            case 1:
            {
                printf("Angga Bertahan\n");
            };break;
            case 2:
            {
                printf("Bobby Bertahan\n");
            };break;
            case 3:
            {
                printf("Chandra Bertahan\n");
            };break;
            case 4:
            {
                printf("Dharma Bertahan\n");
            };break;
            case 5:
            {
                printf("Ega Bertahan\n");
            };break;
            case 6:
            {
                printf("Frisca Bertahan\n");
            };break;
            case 7:
            {
                printf("Goenk Rama Bertahan\n");
            };break;
            case 8:
            {
                printf("Hendra Bertahan\n");
            };break;
            case 9:
            {
                printf("Rio Bertahan\n");
            };break;
            case 10:
            {
                printf("Tangkas Bertahan\n");
            };break;
        }
    }

```

```

        };break;
    }
    for(q=0;q<=250000000;q++){
    }
    switch(r)
    {
        case 1:
        {
            printf("Angga Bertahan\n");
        };break;
        case 2:
        {
            printf("Bobby Bertahan\n");
        };break;
        case 3:
        {
            printf("Chandra Bertahan\n");
        };break;
        case 4:
        {
            printf("Dharma Bertahan\n");
        };break;
        case 5:
        {
            printf("Ega Bertahan\n");
        };break;
        case 6:
        {
            printf("Frisca Bertahan\n");
        };break;
        case 7:
        {
            printf("Goenk Rama Bertahan\n");
        };break;
        case 8:
        {
            printf("Hendra Bertahan\n");
        };break;
        case 9:
        {
            printf("Rio Bertahan\n");
        };break;
        case 10:
        {
            printf("Tangkas Bertahan\n");
        };break;
    }
}
else if(pl==1&&rn==0)
{
    switch(p)
    {
        case 1:
        {
            printf("Angga Melakukan Serangan\n");
        };break;
        case 2:
        {
            printf("Bobby Melakukan Serangan\n");
        };break;
    }
}

```

```

        };break;
    case 3:
    {
        printf("Chandra Melakukan Serangan\n");
    };break;
    case 4:
    {
        printf("Dharma Melakukan Serangan\n");
    };break;
    case 5:
    {
        printf("Ega Melakukan Serangan\n");
    };break;
    case 6:
    {
        printf("Frisca Melakukan Serangan\n");
    };break;
    case 7:
    {
        printf("Goenk Rama Melakukan Serangan\n");
    };break;
    case 8:
    {
        printf("Hendra Melakukan Serangan\n");
    };break;
    case 9:
    {
        printf("Rio Melakukan Serangan\n");
    };break;
    case 10:
    {
        printf("Tangkas Melakukan Serangan\n");
    };break;
}
for(q=0;q<=250000000;q++){
}
switch(r)
{
    case 1:
    {
        printf("Angga Bertahan\n");
    };break;
    case 2:
    {
        printf("Bobby Bertahan\n");
    };break;
    case 3:
    {
        printf("Chandra Bertahan\n");
    };break;
    case 4:
    {
        printf("Dharma Bertahan\n");
    };break;
    case 5:
    {
        printf("Ega Bertahan\n");
    };break;
    case 6:

```



```

        {
            printf("Frisca Bertahan\n");
        };break;
    case 7:
    {
        printf("Goenk Rama Bertahan\n");
    };break;
    case 8:
    {
        printf("Hendra Bertahan\n");
    };break;
    case 9:
    {
        printf("Rio Bertahan\n");
    };break;
    case 10:
    {
        printf("Tangkas Bertahan\n");
    };break;
    }
    def=def-ATT;
    for(q=0;q<=250000000;q++);
    {

    }
    if(def<=0)
    hea=hea+def;
    if(hea<=0)
    hea=0;
}

else if(pl==0&&rn==1)
{
    switch(p)
    {
        case 1:
        {
            printf("Angga Bertahan\n");
        };break;
        case 2:
        {
            printf("Bobby Bertahan\n");
        };break;
        case 3:
        {
            printf("Chandra Bertahan\n");
        };break;
        case 4:
        {
            printf("Dharma Bertahan\n");
        };break;
        case 5:
        {
            printf("Ega Bertahan\n");
        };break;
        case 6:
        {
            printf("Frisca Bertahan\n");
        };break;
        case 7:
        {

```

```

        printf("Goenk Rama Bertahan\n");
    };break;
case 8:
    {
        printf("Hendra Bertahan\n");
    };break;
case 9:
    {
        printf("Rio Bertahan\n");
    };break;
case 10:
    {
        printf("Tangkas Bertahan\n");
    };break;
}
DEF=DEF-att;
for(q=0;q<=250000000;q++);
{
}
switch(r)
{
    case 1:
    {
        printf("Angga Melakukan Serangan\n");
    };break;
    case 2:
    {
        printf("Bobby Melakukan Serangan\n");
    };break;
    case 3:
    {
        printf("Chandra Melakukan Serangan\n");
    };break;
    case 4:
    {
        printf("Dharma Melakukan Serangan\n");
    };break;
    case 5:
    {
        printf("Ega Melakukan Serangan\n");
    };break;
    case 6:
    {
        printf("Frisca Melakukan Serangan\n");
    };break;
    case 7:
    {
        printf("Goenk Rama Melakukan Serangan\n");
    };break;
    case 8:
    {
        printf("Hendra Melakukan Serangan\n");
    };break;
    case 9:
    {
        printf("Rio Melakukan Serangan\n");
    };break;
    case 10:
    {
        printf("Tangkas Melakukan Serangan\n");
    };break;
}

```

```

        };break;
    }
    if(DEF<=0)
    HEA=HEA+DEF;
    if(HEA<=0)
    HEA=0;
}
else
{
    switch(p)
    {
        case 1:
        {
            printf("Angga Melakukan Serangan\n");
        };break;
        case 2:
        {
            printf("Bobby Melakukan Serangan\n");
        };break;
        case 3:
        {
            printf("Chandra Melakukan Serangan\n");
        };break;
        case 4:
        {
            printf("Dharma Melakukan Serangan\n");
        };break;
        case 5:
        {
            printf("Ega Melakukan Serangan\n");
        };break;
        case 6:
        {
            printf("Frisca Melakukan Serangan\n");
        };break;
        case 7:
        {
            printf("Goenk Rama Melakukan Serangan\n");
        };break;
        case 8:
        {
            printf("Hendra Melakukan Serangan\n");
        };break;
        case 9:
        {
            printf("Rio Melakukan Serangan\n");
        };break;
        case 10:
        {
            printf("Tangkas Melakukan Serangan\n");
        };break;
    }
    hea=hea-ATT;
    for(q=0;q<=250000000;q++){
    }
    switch(r)
    {
        case 1:
        {

```

```

        printf("Angga Melakukan Serangan\n");
    };break;
case 2:
    {
        printf("Bobby Melakukan Serangan\n");
    };break;
case 3:
    {
        printf("Chandra Melakukan Serangan\n");
    };break;
case 4:
    {
        printf("Dharma Melakukan Serangan\n");
    };break;
case 5:
    {
        printf("Ega Melakukan Serangan\n");
    };break;
case 6:
    {
        printf("Frisca Melakukan Serangan\n");
    };break;
case 7:
    {
        printf("Goenk Rama Melakukan Serangan\n");
    };break;
case 8:
    {
        printf("Hendra Melakukan Serangan\n");
    };break;
case 9:
    {
        printf("Rio Melakukan Serangan\n");
    };break;
case 10:
    {
        printf("Tangkas Melakukan Serangan\n");
    };break;
    }
    HEA=HEA-att;
    if(HEA<=0)
    HEA=0;
    if(hea<=0)
    hea=0;
    }
    for(q=0;q<=250000000;q++);
    gotoxy(9,3);
    printf("%d    ",HEA);
    gotoxy(9,3);
    printf("%d    ",HEA);
    gotoxy(9,8);
    printf("%d    ",hea);
    gotoxy(9,8);
    printf("%d    ",hea);
    gotoxy(0,14);
    for(q=0;q<=250000000;q++);
}
system("cls");
if(hea<HEA)
printf("Selamat! Anda MENANG!\n");
else if(HEA<hea)

```

```
printf("Sayang sekali, Anda KALAH!\n");
else
printf("Wah, kalian sama-sama KUAT! Permainan SERI!\n");
getch();
system("cls");
printf("Bermain lagi? (1/0)");
scanf("%d",&pil);
if(pil==1)
{
    return main();
}
return 0;
}
```

3.2 Penjelasan Singkat Program

Program ini menyimulasikan permainan RPG. Terdapat dua pemain yang saling mengalahkan dengan cara menyerang dan bertahan.

Jika kedua pemain sama-sama menyerang, maka health kedua pemain akan berkurang sesuai dengan poin attack lawannya. Namun, jika salah satu pemain menyerang dan pemain lain bertahan, atau kedua pemain sama-sama bertahan, maka tidak ada perubahan pada health kedua pemain. Selama health salah satu pemain belum habis (bernilai 0), permainan tetap berlanjut. Pemain dinyatakan menang apabila health pemain lawannya habis.

Penentuan lawan pemain dan keputusan "menyerang atau bertahan" memanfaatkan fungsi rand. Fungsi rand meng-random pilihan dengan hasil modulus bilangan yang telah ditentukan. Bentuk switch-case juga banyak digunakan dalam program ini. Switch-case digunakan dalam pembagian pemilihan karakter dan keadaan serang-bertahan.

3.3 Tampilan Running Program

Monster	1	2	3	4	5	6	7	8	9	10
Health	200	200	200	200	200	200	200	200	200	200
Attack	50	60	60	50	10	40	20	70	30	20
Defend	30	40	10	20	90	60	40	10	50	40

- Angga
- Bobby
- Chandra
- Dharma
- Ega
- Frisca
- Goenk Rama
- Hendra
- Rio
- Tangkas

Pilih Monster:

Tampilan awal program game hero

Angga
VS
Goenk Rama

Tampilan penentuan lawan pemain

```
<PROGRAM GAME HERO>

Angga
Health : 200
Attack : 50
Defend : 30

Goenk Rama
Health : 200
Attack : 20
Defend : 40

ATTACK/DEFENDS (1/0) :
```

Tampilan program menunggu inputan user untuk memutuskan apakah akan menyerang atau bertahan

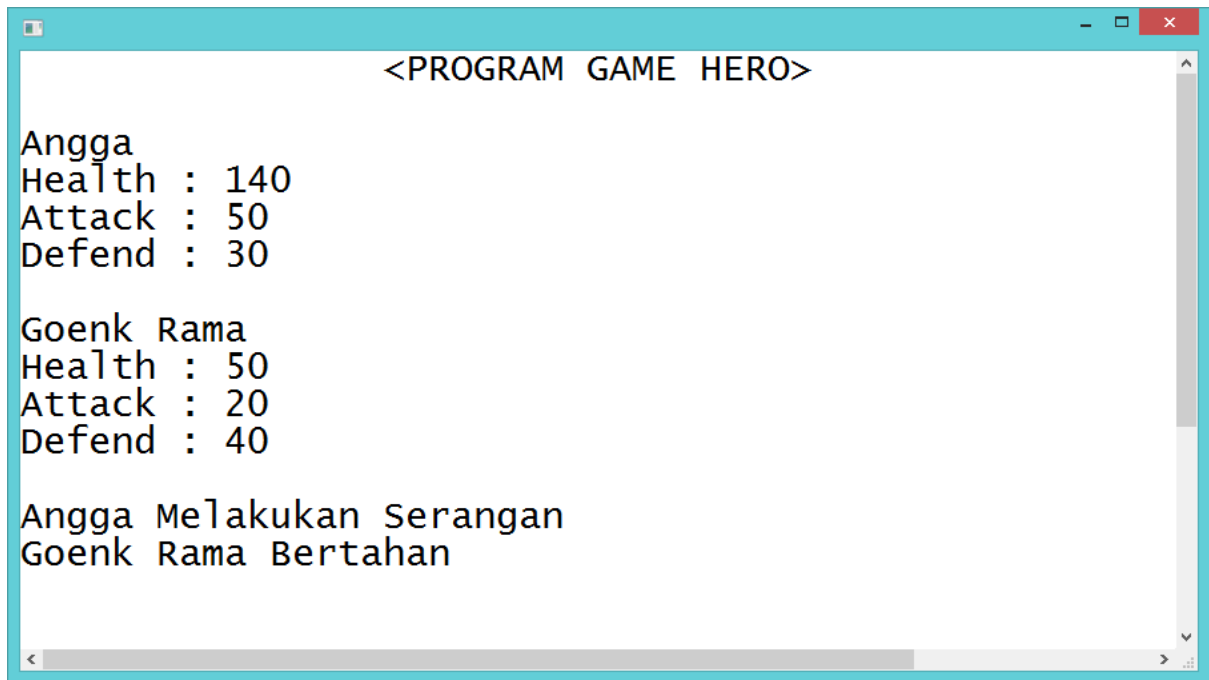
```
<PROGRAM GAME HERO>

Angga
Health : 160
Attack : 50
Defend : 30

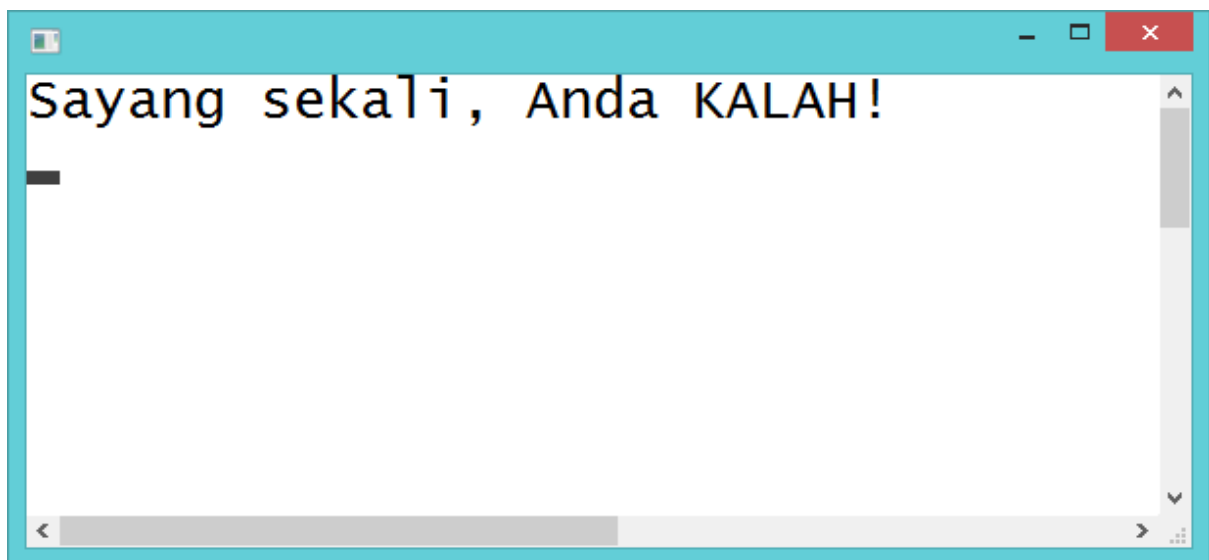
Goenk Rama
Health : 100
Attack : 20
Defend : 40

Angga Melakukan Serangan
Goenk Rama Melakukan Serangan
```

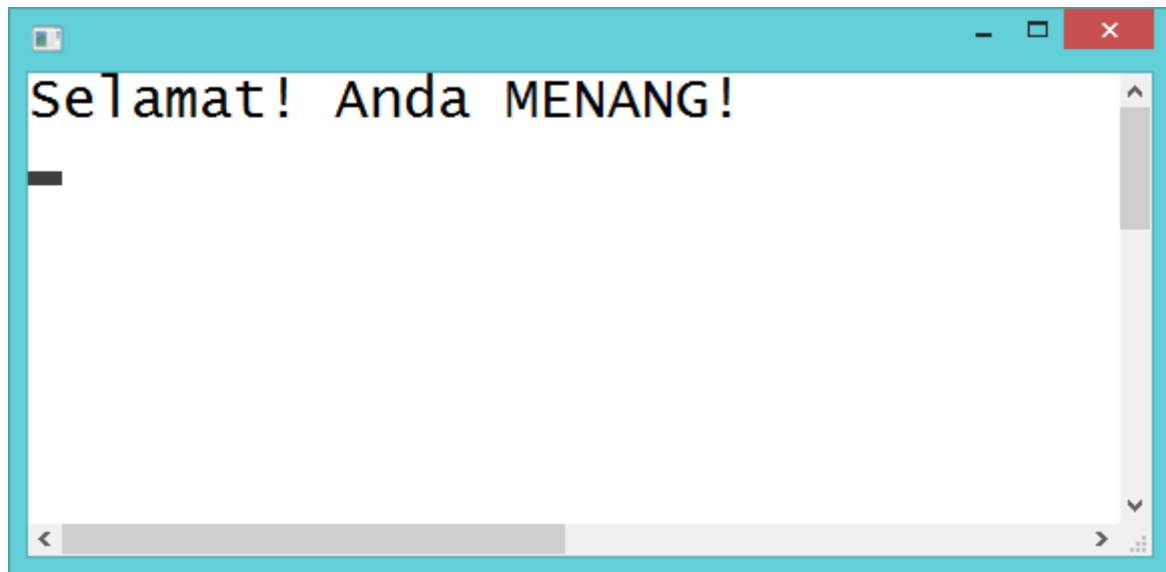
Tampilan saat pemain dan bot sama-sama menyerang, yang menyebabkan pengurangan health di kedua pihak



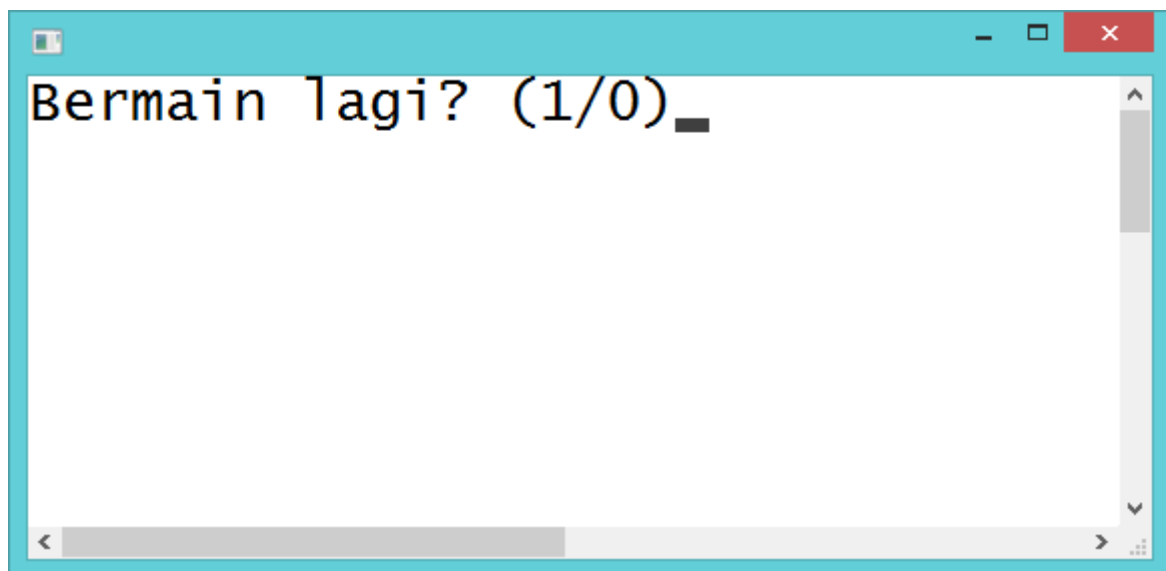
Tampilan saat pemain menyerang dan bot bertahan



Tampilan saat pemain dinyatakan kalah



Tampilan saat pemain dinyatakan menang



Tampilan saat ditanyakan ingin bermain lagi atau tidak