LAPORAN TUGAS BESAR

IF2110/Algoritma dan Struktur Data

<Judul Aplikasi>

Dipersiapkan oleh:

Kelompok 01 – K02

13516035 – M. Sulthan Adhipradhanna

13516062 – Yusuf Rahmat Pratama

13516083 – Abram Perdanaputra

13516086 – Dandy Arif Rahman

13516095 – Faza Fahleraz

13516116 – Ahmad Izzan

Sekolah Teknik Elektro dan Informatika - Institut Teknologi Bandung

Jl. Ganesha 10, Bandung 40132

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| picture | **Sekolah Teknik Elektro dan Informatika ITB** | Nomor Dokumen | | Halaman |
| *IF2110-TB-1-2* | | *<jml hlm>* |
| *Revisi* | *1* | *5/11/2017* |

Daftar Isi

Update daftar isi di bawah ini.

1. Ringkasan 3
2. Penjelasan Tambahan Spesifikasi Tugas 3
   1. Unit White Mage 4
   2. Probabilitas Serangan 4
3. Struktur Data (ADT) 4
   1. ADT Jam 4
   2. ADT Point 5
   3. ADT Matriks 5
   4. ADT Mesin Karakter + Mesin Kata 5
   5. ADT Queue 5
   6. ADT Stack 5

3.7 ADT List Linier (Double Pointer) 5

1. Program Utama 7
2. Algoritma-Algoritma Menarik 7
3. Data Test 7
   1. <Data Test 1> 7
   2. <Data Test 2> 7
4. Test Script 7
5. Pembagian Kerja dalam Kelompok 8
6. Lampiran 8
   1. Deskripsi Tugas Besar 2 8
   2. Notulen Rapat 9
   3. Log Activity Anggota Kelompok 9
   4. <Lain-Lain> 9

# Ringkasan

Pada tugas besar ini, dibuat sebuah program berupa game ber-*genre turn-based* *strategy* menggunakan bahasa C. Game ini dieksekusi menggunakan *command line* dan tanpa menggunakan GUI. Game ini memiliki mekanika sesuai algoritma yang sudah diberikan dalam spesifikasi, dan menggunakan berbagai tipe struktur data yang berbeda.

Dalam laporan ini akan dibahas lebih lengkap spesifikasi dari game yang dibuat, berbagai tipe struktur data yang digunakan, penjelasan algoritma game yang telah dibuat, contoh implementasi program, pembagian tugas, dan lampiran-lampiran yang diperlukan.

Secara keseluruhan, pengerjaan tugas besar ini dilakukan berkelompok dan bersifat modular, dengan pengaplikasian beberapa struktur data yang dibutuhkan sesuai spek. Selain itu, tugas besar ini bermanfaat sebagai salah satu implementasi terhadap berbagai algoritma dan struktur data yang telah dipelajari di mata kuliah Algoritma & Struktur Data.

Deskripsi umum persoalan, dalam tugas besar ini, kami membuat game yang mengadopsi dari *Battle For Wesnoth.* Game ini terdiri dari 2 pemain, dimana masing-masing pemain memiliki *king, castle, dan tower.* Objektif dari game ini adalah membunuh king dari lawan untuk menang. Untuk mencapai objektif dari game ini, pemain bisa merekrut pasukan / unit, menduduki village, menyerang unit & pasukan musuh.

Secara singkat laporan ini terdiri dari ringkasan, penjelasan tambahan spesifikasi tugas, struktur data yang digunakan, program utama, algoritma-algoritma menarik, data test, test script, pembagian kerja kelompok, dan lampiran.

Kesimpulan dari tugas besar ini, game berhasil dibuat dengan spesifikasi-spesifikasi yang wajib, ditambah spesifikasi-spesifikasi bonus diantaranya <spek bonus yg kita kerjain> , dan kami membuat user interface sederhana yang membuat user lebih mudah dalam memainkan game ini.

Isi dengan executive summary dari laporan, antara lain mengenai:

* Deskripsi umum persoalan (hati-hati: tidak menyalin dokumen Deskripsi Tugas besar, dokumen deskripsi Tugas besar diletakkan sebagai lampiran).
* Isi umum laporan secara singkat.
* Kesimpulan tentang hasil Tugas besar secara umum.

# Penjelasan Tambahan Spesifikasi Tugas

Isi dengan penjelasan tambahan mengenai spesifikasi-spesifikasi fitur yang belum rinci dari Deskripsi Tugas besar.

Ingat: Tidak semua fitur perlu dimasukkan. Hanya yang spesifikasinya belum rinci dan perlu penjelasan tambahan. Jika Anda mengerjakan bonus, maka untuk tiap fitur bonus yang Anda kerjakan, harus Anda buat spesifikasinya.

## Unit White Mage

White Mage merupakan unit yang dapat melakukan heal kepada unit yang berlokasi di kiri, kanan, atas, bawah. White Mage dapat menaikan nyawa dari unit sebesar n pada setiap turn. White Mage tidak dapat mengobati diri sendiri.

## Probabilitas Serangan

Probabilitas serangan merupakan mekanisme penyerangan sehingga setiap serangan mempunyai kesempatan menyerang sebesar n persen. Sehingga musuh mempunyai kemungkinan terkena atau tidak terkena serangan sebesar n %.

**2.3 Command Next Unit**

Command Next Unit merupakan mekanisme game sehingga pemain data mengganti unit secara otomatis jika terdapat unit yang masih mimiliki movement points atau masih mempunyai kesempatan serangan .

**2.4 Pergerakan Unit Secara Diagonal**

Secara otomatis, ketika Command Move digunakan, pemain dapat bergerak secara diagonal. Sistem akan menampilkan seluruh kemungkinan gerakan dengan karakter pager. Pada lokasi diagonal, karakter tetap menggunakan 2 movement points.

**2.5 Save dan Load Game**

Pemain dapat menyimpan dan me-load game yang sedang berjalan ke dalam file eksternal. Command save digunakan untuk menyimpan game yang sedang berjalan ke dalam file eksternal. Command load berguna untuk meneruskan game yang sudah disimpan ke file eksternal.

# Struktur Data (ADT)

Isi dengan penjelasan mengenai sketsa struktur data-struktur data (ADT) yang Anda gunakan untuk mengatasi persoalan-persoalan yang Anda temukan pada Tugas Besar 2. Berikan juga alasan pemilihan struktur data tersebut untuk mengatasi persoalan yang berkaitan.

## ADT Jam

ADT Jam ini digunakan untuk menyimpan waktu dalam sebuah tipe yang terstruktur khusus untuk waktu. Dalam tipe jam, terdapat 3 *integer* berupa jam, menit, dan detiknya yang secara keseluruhan merepresentasikan waktu yang disimpan. ADT Jam ini disimpan dalam file ”Jam.h” dan ”Jam.c”.

## ADT Point

ADT Point ini digunakan sebagai penanda lokasi objek (unit dan bangunan) dalam peta. Point ini bertindak sebagai koordinat, dimana dalam tipe point terdiri dari 2 variabel *integer* berupa absis dan ordinat. ADT Point ini disimpan dalam file ”Point.h” dan ”Point.c”.

## ADT Matriks

ADT Matriks digunakan untuk menyimpan data untuk peta dunia game ini, sekaligus indeks dari matriks ini dijadikan representasi koordinat yang ditunjuk menggunakan ADT Point. Dalam tipe Matriks ini terdiri dari matriks itu sendiri, yang didalamnya terdapat data jenis unit dan jenis bangunan yang berada pada setiap koordinat dalam dunia, serta disimpan nilai efektif matriks baris dan kolomnya supaya tidak mengakses indeks kosong. ADT ini disimpan dalam file ”map.h” dan ”map.c”.

## ADT Mesin Karakter + Mesin Kata

ADT Mesin Karakter dan Mesin Kata digunakan untuk menyimpan data ke file eksternal, dan membaca data dari file eksternal. Mesin Kata digunakan untuk *parsing string* yang terdapat pada file eksternal dan menyimpan *string* tersebut dalam struktur data internal yang berupa Point, Matriks, Queue, Stack, dan struktur data lainnya.

## ADT Queue

ADT Queue digunakan untuk menyimpan urutan pemain dalam sebuah permainan.

Dengan aturan FIFO dalam Queue, maka struktur data ini cocok digunakan untuk *swap turn* setiap pemain di dalam *game*, dengan cara menghapus elemen pertama untuk menentukan *current player*. ADT ini disimpan dalam file ”queue.h” dan ”queue.c”.

## ADT Stack

ADT Stack digunakan untuk mengimplementasikan UNDO dalam permainan, yaitu *command* untuk mengembalikan poisisi awal *unit* sebelum *unit* tersebut dilakukan *command MOVE*. Dalam element stack ini disimpan *id unit* dan posisi awal *unit* tersebut. ADT ini disimpan dalam file ”stack.h” dan ”stack.c”.

## 3.7 ADT List Linier (Double Pointer)

ADT List Linier dengan variasi Double Pointer digunakan dalam banyak hal dalam tugas besar ini. Sebagian besar struktur data ini digunakan sebagai *database* untuk permainan. Beberapa contoh adalah ListPlayer, ListUnit, dll. ADT List Linier ini disimpan dalam file ”player.c”, ”player.h”,

**3.8 ADT Castle**

ADT Castle ini digunakan untuk mendefinisikan bangunan castle yang terdiri dari id yang menyatakan unit apa yang terdapat disana, player id yang menyatakan kepemilikan dari castle tersebut, position yang menyatakan koordinat dari castle tersebut, has\_unit yang menyatakan apakah terdapat unit di dalam castle tersebut. ADT castle ini disimpan dalam file "castle.h" dan “castle.c".

**3.9 ADT Map**

ADT Map ini digunakan mendefiniskan map yang diimplementasikan sebagai matriks 2 dimensi, jumlah baris efektif, dan jumlah kolom efektif. ADT Map ini disimpan dalam file "map.h" dan "map.c".

**3.10 ADT Player**

ADT Player ini digunakan untuk mendefinisikan player, struktur dari player adalah id yang mendefinisikan id dari player tersebut, alive yang menyatakan apakah player masih hidup, cash yang menyatakan uang yang dimiliki player, income yang menyatakan pendapatan yang diterima player setiap turn, upkeep yang menyatakan pengeluaran yang harus dikeluarkan player setiap turn, king\_id yang menyatakan id dari king player, array of units yang berisi id unit yang dimiliki player, unit\_count yang menyatakan banyakanya unit yang dimiliki oleh player, array of villages yang berisi id village yang dimiliki player, village\_count yang menyatakan jumlah village yang dimiliki player. ADT player ini disimpan dalam file "player.h" dan "player.c"

**3.11 ADT Tower**

ADT Tower ini digunakan untuk mendefinisikan tower, struktur dari tower adalah id yang menyatakan id unit dari tower tersebut, player\_id yang menyatakan id dari player pemilik tower tersebut. ADT tower ini disimpan dalam file "tower.h" dan "tower.c".

**3.12 ADT Unit**

ADT Unit ini digunakan untuk mendefinisikan unit, struktur dari unit adalah id yang menyatakan id dari unit tersebut, player\_id yang menyatakan id player pemilik unit tersebut, attack\_type yang menyatakan tipe serangan dari unit tersebut, max\_health yang menyatakan darah maksimum dari unit tersebut, health yang menyatakan darah sekarang dari unit tersebut, attack yang menyatakan besaran serangan, max\_move\_points yang menyatakan nilai maksimum dari move point unit tersebut, move\_points yang menyatakan nilai move point yang tersisa dari unit tersebut, can\_attack yang menyatakan apakah unit tersebut memiliki kesempatan untuk menyerang, location yang menyatakan koordinat dari unit tersebut, price yang menyatakan harga dari unit tersebut, heal yang menyatakan besaran heal yang bisa dilakukan oleh unit tersebut. ADT unit ini disimpan dalam file "unit.h" dan "unit.c".

**3.13 ADT Village**

ADT Village ini digunakan untuk mendefinisikan village, struktur dari village adalah id yang menyatakan id dari unit tersebut dan player\_id yang menyatakan id player pemilik unit tersebut. ADT village ini disimpan dalam file "village.h" dan "village.c"

# Program Utama

Isi dengan penjelasan mengenai algoritma program utama.

# Algoritma-Algoritma Menarik

Isi dengan penjelasan mengenai algoritma-algoritma menarik yang Anda temukan atau Anda buat dalam Tugas Besar 2, jika ada. Berikan penjelasan singkat, di mana algoritma tersebut digunakan dan mengapa Anda anggap algoritma tersebut menarik.

**Jika tidak ada, hapus bab ini.**

**5.1 Algoritma BFS**

Isi dengan algoritma yang dimaksud, di mana algoritma tersebut digunakan, dan mengapa menarik.

# Data Test

Isi dengan data test dan penjelasan yang diperlukan. Hal-hal yang harus dijelaskan untuk tiap data test:

* Fitur apa (saja) yang dites dengan data test ini
* Hasil apa yang seharusnya diberikan
* Penjelasan lain (jika ada)
* Data test-nya sendiri.

## <Data Test 1>

Isi dengan penjelasan mengenai data test.

## <Data Test 2>

Isi dengan penjelasan mengenai data test.

# Test Script

Isi dengan skenario test yang dimungkinkan untuk semua fitur yang ada. Bisa dibuat dalam bentuk tabel sebagai berikut:

| **No.** | **Fitur yang Dites** | **Tujuan Testing** | **Langkah-Langkah Testing** | **Input Data Test** | **Hasil yang Diharapkan** | **Hasil yang Keluar** |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | <Fitur 1> | … | … | <tinggal refer ke data test di atas> | … |  |
| 2 | <Fitur 1> | … | … | … | … |  |
| 3 |  |  |  |  |  |  |
| … |  |  |  |  |  |  |
| 5 | <Fitur 2> | … | … | … | … |  |
| 6 | <Fitur 2> | … | … | … | … |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

# Pembagian Kerja dalam Kelompok

|  |  |  |
| --- | --- | --- |
| NAMA | NIM | TUGAS |
| M. Sulthan A. | 13516035 | Mekanisme Game, Konstruktor Game, Laporan |
| Yusuf Rahmat P. | 13516062 | Mekanisme Game, Konstruktor Game |
| Abram Perdanaputra | 13516083 | Mekanisme Game, Testing, Konstruktor |
| Dandy Arif Rahman | 13516086 | Mekanisme Game , Laporan, |
| Faza Fahleraz | 13516095 | User Interface, Mekanisme Game , Konstruktor |
| Ahmad Izzan | 13516116 | Konstruktor Game, Mekanisme Game |

# Lampiran

## Deskripsi Tugas Besar 2

Battle for Olympia adalah permainan dengan *genre* ***turn-based strategy*** yang dimainkan dengan cara memasukkan perintah melalui *command line interface* . Tujuan utama dari permainan ini adalah menghancurkan kerajaan musuh dengan cara membunuh raja yang memimpin seluruh unit musuh. Seorang raja dapat merekrut beberapa pasukan, dan pasukan dapat dikerahkan untuk membunuh pasukan dan raja kerajaan musuh. Apabila raja mati, maka pemain selesai bermain dan seluruh pasukan yang tersisa ikut mati.

Permainan ini merupakan simplifikasi dari permainan Battle for Wesnoth. Permainan ini *open source* dan *setup* -nya berukuran sekitar 300 MB. Anda dapat mencoba memainkan permainan tersebut (minimal menyelesaikan tutorial) agar permainan tugas besar ini menjadi lebih intuitif.

Dari deskripsi game di atas, anda perlu membuat sebuah program yang dapat mensimulasikan *game* tersebut menggunakan bahasa C. Interaksi dari pengguna terhadap program tersebut perlu diimplementasikan menggunakan *Command Line Interface* (CLI) dengan *command* yang akan dijelaskan pada masing-masing fitur game.

## Notulen Rapat

**2/11/2017**

Selasar Dingdong, pada malam ini kami membahas asbtraksi persoalan tugas besar.

**6/11/2017**

Apartement Izzan, pada malam ini kami telah membahas lebih lanjut tentang pembagian tugas dan abstraksi persoalan tugas besar, membuat list ADT yang diperlukan diantaranya *player, map,* *unit*, membuat beberapa fungsi yang diperlukan diantaranya *Mechanics, Recruit, Attack, PlayerTurn, ChangeActiveUnit*, membuat google drive untuk tugas besar.

**12/11/2017**

Apartement Faza, pada malam ini kami memutuskan untuk memindakan fungsi-fungsi yang sebelumnya dalam bentuk ADT terpisah, kemudian dimasukkan ke *mechanics,* melanjutkan pengerjaan fungsi *recruit,* dan membuat UI untuk game tersebut.

**13/11/2017**

Appartement Izzan, pada malam ini kami melanjutkan menyelesaikan fungsi *NextActiveUnit, Attack, Random, ChangeActiveUnit,* melakukan percobaan ke program, *debugging.*

**14/11/2017**

## Log Activity Anggota Kelompok

13516035 – M. Sulthan Adhipradhanna

13516062 – Yusuf Rahmat Pratama

13516083 – Abram Perdanaputra

13516086 – Dandy Arif Rahman

13516095 – Faza Fahleraz

13516116 – Ahmad Izzan

## <Lain-Lain>

Hanya diisi jika memang ada hal-hal lain yang perlu dilampirkan. Jika tidak ada yang ditambahkan, hapus subbab ini.