

LAPORAN TUGAS BESAR

IF2110/Algoritma dan Struktur Data

<Judul Aplikasi>

Lambang?


Dipersiapkan oleh:

Kelompok 8

- | | |
|--------------------------|----------|
| 1. Ivan Jonathan | 13516059 |
| 2. Seperayo | 13516068 |
| 3. Aldo Azali | 13516125 |
| 4. Shevalda Gracielifira | 13516134 |
| 5. Hafizh Budiman | 13516137 |

Sekolah Teknik Elektro dan Informatika - Institut Teknologi Bandung

Jl. Ganesha 10, Bandung 40132

	Sekolah Teknik Elektro dan Informatika ITB	Nomor Dokumen		Halaman
		IF2110-TB-<08>-<02>		<jml hlm>
		Revisi	1	<Tgl release>

Daftar Isi

1 Ringkasan.....	4
2 Penjelasan Tambahan Spesifikasi Tugas	4
2.1 <Spesifikasi Fitur Tambahan 1>.....	4
2.2 <Spesifikasi Fitur Tambahan 2>.....	4
2.3 <Spesifikasi Fitur Tambahan 3>.....	4
2.4 <Spesifikasi Fitur Tambahan 4>.....	5
3 Struktur Data (ADT).....	5
3.1 <Struktur Data 1>	5
3.2 <Struktur Data 2>	5
3.3 <Struktur Data 3>	5
3.4 <Struktur Data 4>	5
4 Program Utama	5
5 Algoritma-Algoritma Menarik.....	5
5.1 <Algoritma 1>.....	6
5.2 <Algoritma 2>.....	6
6 Data Test	6
6.1 <Data Test 1>.....	6
6.2 <Data Test 2>.....	6
6.3 <Data Test 3>.....	6
6.4 <Data Test 4>.....	6
6.5 <Data Test 5>.....	7
6.6 <Data Test 6>.....	7
6.7 <Data Test 7>.....	7
6.8 <Data Test 8>.....	7
6.9 <Data Test 9>.....	7
6.10 <Data Test 10>.....	7
6.11 <Data Test 11>.....	7
6.12 <Data Test 12>.....	8
6.13 <Data Test 13>.....	8
6.14 <Data Test 14>.....	8
6.15 <Data Test 15>.....	8
6.16 <Data Test 16>.....	8
6.17 <Data Test 17>.....	8
6.18 <Data Test 18>.....	8
6.19 <Data Test 19>.....	9
6.20 <Data Test 20>.....	9
6.21 <Data Test 21>.....	9
6.22 <Data Test 22>.....	9
7 Test Script	10
8 Pembagian Kerja dalam Kelompok	20
9 Lampiran	20
9.1 Deskripsi Tugas Besar 2	20

9.2	Notulen Rapat	20
9.3	Log Activity Anggota Kelompok	20
9.4	<Lain-Lain>	21

1 Ringkasan

Laporan kali ini membahas seputar aplikasi dari bahasa pemrograman C dalam bentuk "Battle of Olympia" yang merupakan suatu *turn based game*.

1.1.Deskripsi Umum

1.2.Isi Laporan

1.3.Kesimpulan

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.

2 Penjelasan Tambahan Spesifikasi Tugas

Berikut beberapa fitur yang belum memiliki penjelasan rinci di dalam spesifikasi awal :

2.1.Fitur Bonus

2.2.Fitur Tambahan

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.

2.1 <Spesifikasi Fitur Tambahan 1>

Isi dengan penjelasan spesifikasi fitur tambahan 1.

2.2 <Spesifikasi Fitur Tambahan 2>

Isi dengan penjelasan spesifikasi fitur tambahan 2.

2.3 <Spesifikasi Fitur Tambahan 3>

Isi dengan penjelasan spesifikasi fitur tambahan 1.

2.4 <Spesifikasi Fitur Tambahan 4>

Isi dengan penjelasan spesifikasi fitur tambahan 2.

3 Struktur Data (ADT)

Struktur data yang kami gunakan dalam menyelesaikan tugas kali ini jumlahnya ada beberapa, yaitu :

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.

3.1 <Struktur Data 1>

Isi dengan penjelasan mengenai sketsa struktur data, persoalan apa yang diselesaikan, dan alasan pemilihan, serta jelaskan diimplementasikan sebagai ADT apa (berikut nama file-nya).

3.2 <Struktur Data 2>

Isi dengan penjelasan mengenai sketsa struktur data, persoalan apa yang diselesaikan, dan alasan pemilihan, serta jelaskan diimplementasikan sebagai ADT apa (berikut nama file-nya).

3.3 <Struktur Data 3>

Isi dengan penjelasan mengenai sketsa struktur data, persoalan apa yang diselesaikan, dan alasan pemilihan, serta jelaskan diimplementasikan sebagai ADT apa (berikut nama file-nya).

3.4 <Struktur Data 4>

Isi dengan penjelasan mengenai sketsa struktur data, persoalan apa yang diselesaikan, dan alasan pemilihan, serta jelaskan diimplementasikan sebagai ADT apa (berikut nama file-nya).

4 Program Utama

Berikut adalah penjelasan program utama dalam tugas kami :

Isi dengan penjelasan mengenai algoritma program utama.

5 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 1>

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

5.2 <Algoritma 2>

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

6 Data Test

Berikut beberapa data yang kami gunakan untuk melakukan pengecekan atau *testing* :

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.

6.1 <Data Test 1>

- a. Fitur = MOVE
- b. Data Test = Koordinat awal<0,1>; MOVE → <0,0>
- c. Hasil seharusnya =
- d. Penjelasan = -

6.2 <Data Test 2>

- a. Fitur = MOVE
- b. Data Test = Koordinat awal<0,0>; MOVE → <0,1>
- c. Hasil seharusnya =
- d. Penjelasan =

6.3 <Data Test 3>

- a. Fitur = MOVE
- b. Data Test = Koordinat awal<1,0>; MOVE → <0,0>
- c. Hasil seharusnya =
- d. Penjelasan =

6.4 <Data Test 4>

- a. Fitur = MOVE
- b. Data Test = Koordinat awal<0,0>; MOVE → <1,0>
- c. Hasil seharusnya =
- d. Penjelasan =

6.5 <Data Test 5>

- a. Fitur = MOVE (Bonus)
- b. Data Test = Koordinat awal<0,0>; MOVE → <1,1>
- c. Hasil seharusnya =
- d. Penjelasan =

6.6 <Data Test 6>

- a. Fitur = Movement Point - MOVE (Bonus)
- b. Data Test = Koordinat awal<0,0>; MOVE → <1,1>; CHANGE_UNIT
- c. Hasil seharusnya =
- d. Penjelasan =

6.7 <Data Test 7>

- a. Fitur = MOVE
- b. Data Test = command MOVE
- c. Hasil seharusnya =
- d. Penjelasan =

6.8 <Data Test 8>

- a. Fitur = MOVE
- b. Data Test = command MOVE
- c. Hasil seharusnya =
- d. Penjelasan =

6.9 <Data Test 9>

- a. Fitur = MOVE
- b. Data Test = Ukuran 8x8; Koordinat awal <1,4>; MOVE → <1,5>
- c. Hasil seharusnya =
- d. Penjelasan =

6.10 <Data Test 10>

- a. Fitur = MOVE
- b. Data Test = Koordinat awal <0,0> (Unit lain di <1,0>); MOVE → <2,0>
- c. Hasil seharusnya =
- d. Penjelasan =

6.11 <Data Test 11>

- a. Fitur = MOVE
- b. Data Test = Koordinat awal <0,0> (Unit lain di <1,0> dan <2,0>); MOVE → <2,0> | MOVE → <1,0>
- c. Hasil seharusnya =
- d. Penjelasan = Unit tidak bisa digerakkan karena di petak tujuan telah terdapat unit lain.

6.12<Data Test 12>

- a. Fitur = MOVE
- b. Data Test = Koordinat awal <0,0> (Unit musuh di <1,0>); MOVE → <2,0>
- c. Hasil seharusnya =
- d. Penjelasan = Unit tidak dapat menembus karena unit yang ditembus bukan unit kerajaan sendiri

6.13<Data Test 13>

- a. Fitur = Movement Point - MOVE
- b. Data Test = Movement Point=0; Koordinat Awal <0,0>; MOVE → <1,0>
- c. Hasil seharusnya =
- d. Penjelasan = Unit tidak dapat digerakkan karena Movement Point yang dimiliki sudah habis.

6.14<Data Test 14>

- a. Fitur = Movement Point – MOVE - ATTACK
- b. Data Test = Koordinat Awal<0,0>; ATTACK; MOVE → <1,0>
- c. Hasil seharusnya =
- d. Penjelasan = Tidak dapat melakukan gerakan akibat command ATTACK telah dilakukan.

6.15<Data Test 15>

- a. Fitur = Movement Point
- b. Data Test = Koordinat awal<0,0>; MOVE → <1,0>; CHANGE_UNIT
- c. Hasil seharusnya =
- d. Penjelasan =

6.16<Data Test 16>

- a. Fitur = MOVE
- b. Data Test = Koordinat awal<0,0>; MOVE → <-1,0>
- c. Hasil seharusnya =
- d. Penjelasan = Unit tidak dapat digerakkan karena petak tujuan di luar rentang peta.

6.17<Data Test 17>

- a. Fitur = UNDO
- b. Data Test = Movement Point=Max; UNDO
- c. Hasil seharusnya =
- d. Penjelasan = Unit tidak memiliki rekap gerakan sebelumnya, sehingga command UNDO gagal

6.18<Data Test 18>

- a. Fitur = Movement Point-UNDO
- b. Data Test = Koordinat awal<0,0>; Movement Point=2; MOVE → <1,0>; Movement Point=1; UNDO; CHANGE_UNIT

- c. Hasil seharusnya =
- d. Penjelasan =

6.19 <Data Test 19>

- a. Fitur = UNDO
- b. Data Test = Koordinat awal <0,0>; MOVE → <1,0>; UNDO; INFO → <0,0>
- c. Hasil seharusnya =
- d. Penjelasan =

6.20 <Data Test 20>

- a. Fitur = UNDO
- b. Data Test = Koordinat awal <0,0>; MOVE → (2,0); UNDO; UNDO; UNDO
- c. Hasil seharusnya =
- d. Penjelasan = Command UNDO yang ketiga akan gagal dan posisi unit kembali ke <0,0>

6.21 <Data Test 21>

- a. Fitur = UNDO
- b. Data Test = ATTACK, UNDO
- c. Hasil seharusnya =
- d. Penjelasan = Command UNDO akan gagal karena Movement Point yang dimiliki setelah command ATTACK adalah 0.

6.22 <Data Test 22>

- a. Fitur = CHANGE_UNIT
- b. Data Test = Movement Point=2; CHANGE_UNIT → 1
- c. Hasil seharusnya =
- d. Penjelasan =

6.23 <Data Test 23>

- a. Fitur = CHANGE_UNIT
- b. Data Test = Movement Point=0; CHANGE_UNIT → 1
- c. Hasil seharusnya =
- d. Penjelasan = Unit tidak dapat dipilih karena tidak dapat digerakkan kembali.

6.24 <Data Test 24>

- a. Fitur = CHANGE_UNIT
- b. Data Test = CHANGE_UNIT
- c. Hasil seharusnya =
- d. Penjelasan =

6.25 <Data Test 25>

- a. Fitur = CHANGE_UNIT
- b. Data Test = CHANGE_UNIT

- c. Hasil seharusnya =
- d. Penjelasan =

6.26 <Data Test 26>

- a. Fitur = RECRUIT
- b. Data Test = CurrentUnit=Archer; RECRUIT
- c. Hasil seharusnya =
- d. Penjelasan = command RECRUIT gagal untuk dijalankan karena unit bukan KING.

6.27 <Data Test 27>

- a. Fitur = RECRUIT
- b. Data Test = King at <0,0>; RECRUIT
- c. Hasil seharusnya =
- d. Penjelasan = command RECRUIT gagal untuk dijalankan karena KING harus di Tower.

6.28 <Data Test 28>

- a. Fitur = RECRUIT
- b. Data Test = Archer at <1,5> (Castle); RECRUIT(1,5)
- c. Hasil seharusnya =
- d. Penjelasan = command RECRUIT gagal untuk diselesaikan karena Castle dihuni unit lain.

6.29 <Data Test 29>

- a. Fitur = RECRUIT
- b. Data Test = CHANGE_UNIT → 1
- c. Hasil seharusnya =
- d. Penjelasan =

7 Test Script

Test Script yang kami lakukan dapat digambarkan sebagai berikut :

No.	Fitur yang Dites	Tujuan Testing	Langkah-Langkah Testing	Input Data Test	Hasil yang Diharapkan	Hasil yang Keluar
1.	MOVE	Menguji command MOVE ke atas	...	<tinggal refer ke data test di atas>	Unit dapat bergerak ke atas.	
2.	MOVE	Menguji command MOVE ke bawah.			Unit dapat bergerak ke bawah.	
3.	MOVE	Menguji command MOVE			Unit dapat bergerak ke	

No.	Fitur yang Dites	Tujuan Testing	Langkah-Langkah Testing	Input Data Test	Hasil yang Diharapkan	Hasil yang Keluar
		ke kiri.			kiri.	
4.	MOVE	Menguji command MOVE ke kanan.			Unit dapat bergerak ke kanan.	
5.	MOVE (Bonus)	Menguji apakah unit bergerak ke petak diagonal pada command MOVE.			Unit berhasil bergerak ke petak diagonal.	
6.	Movement Point - MOVE (Bonus)	Menguji apakah gerakan diagonal command MOVE akan menggunakan dua kali lipat movement point dari gerakan <i>adjacent</i> .			Gerakan diagonal menghabiskan 2 kali Movement Point dari gerakan <i>adjacent</i> .	
7.	MOVE	Menguji apakah peta muncul ketika diberikan command MOVE.			Peta berhasil muncul.	
8.	MOVE	Menguji apakah kemungkinan gerak unit (#) ditampilkan.			Kemungkinan gerak unit ditampilkan.	
9.	MOVE	Menguji apakah unit dapat masuk ke <i>castle</i> atau <i>tower</i> musuh.			Unit dapat bergerak ke <i>castle</i> atau <i>tower</i> musuh.	
10.	MOVE	Menguji command MOVE ketika unit menembus unit kerajaan sendiri dan movement point lebih banyak dari jumlah unit yang ditembus.			Unit berhasil menembus unit kerajaan sendiri.	
11.	MOVE	Menguji command MOVE ketika unit menembus unit kerajaan sendiri dan movement point lebih sedikit atau sama dengan jumlah unit yang ditembus.			Unit tidak berhasil menembus unit kerajaan sendiri.	
12.	MOVE	Menguji			Unit tidak	

No.	Fitur yang Dites	Tujuan Testing	Langkah-Langkah Testing	Input Data Test	Hasil yang Diharapkan	Hasil yang Keluar
		command MOVE ketika menembus unit kerajaan musuh.			dapat menembus.	
13.	Movement Point - MOVE	Menguji apakah unit dapat bergerak setelah movement point bernilai 0.			Unit tidak dapat bergerak.	
14.	Movement Point – MOVE - ATTACK	Menguji apakah unit dapat bergerak setelah command ATTACK.			Unit tidak dapat bergerak.	
15.	Movement Point-MOVE	Menguji apakah Movement Point berkurang setelah unit bergerak.			Movement Point berhasil dikurangi oleh jumlah gerakan.	
16.	MOVE	Menguji apakah command MOVE dapat menggerakkan unit melebihi ukuran peta			Unit gagal digerakkan	
17.	UNDO	Menguji command UNDO ketika movement point pada nilai maksimum.	Command UNDO tidak dapat dijalankan.	
18.	Movement Point - UNDO	Menguji apakah Movement Point bertambah sebanding dengan jumlah command UNDO yang diberikan.			Movement Point bertambah.	
19.	UNDO	Menguji command UNDO untuk mengubah posisi unit ke posisi sebelumnya.			Command UNDO berhasil.	
20.	UNDO	Menguji UNDO yang dapat dijalankan lebih dari jumlah command MOVE yang telah diberikan.			Unit dapat kembali ke petak awal, dan command UNDO selanjutnya gagal.	
21.	UNDO	Menguji command UNDO			Command UNDO gagal.	

No.	Fitur yang Dites	Tujuan Testing	Langkah-Langkah Testing	Input Data Test	Hasil yang Diharapkan	Hasil yang Keluar
		setelah diberikan command selain MOVE, MAP, dan INFO.				
22.	CHANGE_UNIT	Menguji apakah unit yang masih memiliki movement point dapat dipilih pada command CHANGE_UNIT.			Unit berhasil dipilih dari daftar unit.	
23.	CHANGE_UNIT	Menguji apakah unit yang tidak memiliki movement point dapat dipilih pada command CHANGE_UNIT.			Unit tidak dapat dipilih dari daftar unit.	
24.	CHANGE_UNIT	Menguji apakah daftar unit yang ditampilkan pada command CHANGE_UNIT terdiri dari seluruh unit yang dimiliki <i>player</i> .			Daftar unit yang ditampilkan lengkap.	
25.	CHANGE_UNIT	Menguji apakah rincian daftar unit yang ditampilkan pada command CHANGE_UNIT lengkap dan benar.			Rincian unit pada daftar unit lengkap dan benar.	
26.	RECRUIT	Menguji apakah command RECRUIT dapat dijalankan di jenis unit selain King.			Command RECRUIT tidak dapat dijalankan.	
27.	RECRUIT	Menguji apakah command RECRUIT dapat dijalankan ketika King tidak di petak <i>tower</i> .			Command RECRUIT tidak dapat dijalankan.	
28.	RECRUIT	Menguji apakah command RECRUIT dapat dijalankan ketika King memilih castle yang masih ditempati oleh			Command RECRUIT tidak dapat dijalankan.	

No.	Fitur yang Dites	Tujuan Testing	Langkah-Langkah Testing	Input Data Test	Hasil yang Diharapkan	Hasil yang Keluar
		sebuah unit.				
29.	RECRUIT	Menguji apakah unit yang baru direcruit ditampilkan di koordinat castle yang telah dipilih			Unit berhasil ditampilkan di koordinat castle yang telah dipilih.	
30.	RECRUIT	Menguji apakah command RECRUIT dapat menampilkan daftar harga dan rincian unit yang dapat dibeli.			Command RECRUIT berhasil menampilkan daftar harga dan rincian unit.	
31.	RECRUIT	Menguji apakah King dapat merecruit unit yang harganya lebih mahal dari cash yang dipunya pada command RECRUIT.			Command RECRUIT tidak dapat dijalankan.	
32.	Gold - RECRUIT	Menguji apakah gold berkurang sesuai harga unit yang direcruit.			Gold berkurang sesuai harga unit yang direcruit.	
33.	RECRUIT-Unit	Menguji apakah unit memiliki movement point setelah direcruit pada turn yang sama.			Nilai movement point unit yang baru direcruit bernilai 0.	
34.	RECRUIT-Unit	Menguji apakah Health dari unit yang baru di recruit bernilai maksimum.			Nilai Health unit yang baru direcruit bernilai maksimum.	
35.	RECRUIT	Menguji apakah nilai movement point King berkurang setelah command RECRUIT.			Nilai movement point King tidak berubah setelah command RECRUIT.	
36.	ATTACK	Menguji apakah list unit musuh yang dapat diserang tampil saat command ATTACK			List unit musuh yang dapat diserang berhasil ditampilkan.	

No.	Fitur yang Dites	Tujuan Testing	Langkah-Langkah Testing	Input Data Test	Hasil yang Diharapkan	Hasil yang Keluar
		dijalankan.				
37.	ATTACK	Menguji apakah unit-unit yang ditampilkan di list sesuai dengan rumus $ i_1 - i_2 + j_1 - j_2 = 1$			List unit yang ditampilkan benar	
38.	ATTACK	Menguji command ATTACK untuk kedua kalinya pada unit dan turn yang sama.			Command ATTACK tidak dapat dijalankan.	
39.	ATTACK (Bonus)	Menguji apakah command ATTACK memiliki probabilitas untuk <i>miss</i>			Command ATTACK berhasil memiliki probabilitas untuk <i>miss</i> .	
40.	MAP	Menguji apakah command MAP dapat menampilkan peta dengan koordinat yang sesuai input			Tampilan peta dan koordinatnya sesuai dengan input <i>user</i>	
41.	MAP	Menguji apakah command MAP dapat menampilkan jenis petak dengan benar			Jenis petak di map ditampilkan secara benar	
42.	MAP	Menguji apakah command MAP dapat menampilkan kepemilikan petak (warna) dengan benar			Warna setiap petak ditampilkan sesuai kepemilikan player di petak tersebut.	
43.	MAP	Menguji apakah warna untuk masing-masing unit player berbeda			Warna masing-masing unit player berbeda	
44.	MAP	Menguji apakah command MAP dapat menampilkan unit di petak yang benar			Unit berhasil ditampilkan dengan benar	
45.	MAP	Menguji apakah command MAP			Tampilan petak sesuai	

No.	Fitur yang Dites	Tujuan Testing	Langkah-Langkah Testing	Input Data Test	Hasil yang Diharapkan	Hasil yang Keluar
		dapat menampilkan unit dan jenis petak dengan format yang benar			dengan format yang ditentukan	
46.	MAP-Inisiasi	Menguji apakah susunan Tower, Castle, dan King ditampilkan sesuai dengan format.			Tower, Castle, dan King ditampilkan sesuai dengan format.	
47.	MAP-Village	Menguji apakah Village dapat ditempatkan secara random dan tidak bertabrakan dengan lokasi lain.			Village berhasil ditempatkan secara random dan tidak bertabrakan dengan lokasi lain.	
48.	MAP-Village	Menguji apakah Village menunjukkan kepemilikan pada saat Inisiasi Peta			Village tidak memiliki warna.	
49.	INFO	Menguji apakah command INFO dapat menampilkan jenis petak yang dipilih dengan benar.			Command INFO dapat menampilkan jenis petak dengan benar.	
50.	INFO	Menguji apakah command INFO dapat menampilkan rincian unit dari petak yang dipilih.			Command INFO dapat menampilkan informasi unit dengan benar.	
51.	END_TURN	Menguji apakah command END_TURN dapat mengakhiri turn player yang sebelumnya.			Command END_TURN dapat mengakhiri turn player sebelumnya.	
52.	END_TURN	Menguji apakah player dapat mengakses unitnya setelah player mengakhiri <i>turn</i> -nya.			Player tidak bisa mengakses unitnya setelah <i>end turn</i> .	
53.	END_TURN	Menguji apakah player selanjutnya			Player selanjutnya	

No.	Fitur yang Dites	Tujuan Testing	Langkah-Langkah Testing	Input Data Test	Hasil yang Diharapkan	Hasil yang Keluar
		dapat mengakses unitnya setelah player sebelumnya menjalankan command END TURN.			dapat mengakses unitnya.	
54.	EXIT	Menguji apakah <i>user</i> dapat keluar dari program saat command EXIT dijalankan.			<i>User</i> dapat keluar setelah command EXIT dijalankan.	
55.	SAVE (Bonus)	Menguji apakah command SAVE berhasil menyimpan data di file eksternal.	Command SAVE berhasil menyimpan data permainan ke file eksternal.	
56.	LOAD (Bonus)	Menguji apakah command LOAD dapat mengambil data yang benar dari file eksternal yang dipilih.	Command LOAD berhasil mengambil data yang benar dari file eksternal yang dipilih.	
57.	Kondisi akhir	Menguji apakah permainan dapat berakhir ketika hanya ada satu King yang tersisa.			Permainan berhasil berakhir setelah hanya satu King yang tersisa.	
58.	Kondisi akhir	Menguji apakah permainan dapat menampilkan pemenang dengan benar.			Pemain yang ditampilkan adalah pemain yang memenangkan permainan.	
59.	Main Menu	Menguji apakah Main Menu dapat ditampilkan dengan benar saat program pertama kali dijalankan			Main Menu tampil dengan benar	
60.	Main Menu	Menguji apakah <i>user</i> dapat memilih New Game di Main Menu			New Game berhasil dipilih di Main Menu	
61.	Main Menu	Menguji apakah <i>user</i> dapat			Load Game berhasil	

No.	Fitur yang Dites	Tujuan Testing	Langkah-Langkah Testing	Input Data Test	Hasil yang Diharapkan	Hasil yang Keluar
		memilih Load Game di Main Menu			dipilih di Main Menu	
62.	Main Menu	Menguji apakah input peta di bawah 8x8 dapat dijalankan			Inisiasi permainan gagal.	
63.	Main Menu	Menguji apakah input peta di atas batas minimum (8x8) dapat dijalankan			Inisiasi permainan berhasil	
64.	Unit-Health	Menguji apakah unit dihapus ketika Health \leq 0			Unit berhasil dihapus dari map	
65.	Unit-Health	Menguji apakah Health berkurang sesuai dengan attack damage yang diterima			Health berkurang sejumlah dengan attack damage yang diterima.	
66.	Unit-Health	Menguji apakah unit suatu player yang berada di Village bertambah Health-nya secara otomatis setiap turn player itu sendiri.			Unit berhasil bertambah Healthnya.	
67.	Unit-Health	Menguji apakah unit suatu player yang berada di dekat White Mage bertambah Health-nya secara otomatis setiap turn player itu sendiri.			Unit berhasil bertambah Healthnya.	
68.	Movement Point	Menguji apakah Movement Point akan selalu menjadi Max Movement Point untuk setiap awal <i>turn</i> .			Movement Point selalu menjadi Max Movement Point untuk setiap awal <i>turn</i> .	
69.	Unit-Retaliatio	Menguji apakah unit yang tipe serangannya sama dengan unit penyerang dapat melakukan			Unit yang diserang dapat melakukan retaliatio	

No.	Fitur yang Dites	Tujuan Testing	Langkah-Langkah Testing	Input Data Test	Hasil yang Diharapkan	Hasil yang Keluar
		retaliation jika diserang.				
70.	Unit-Retaliation	Menguji apakah unit (kecuali King dan Archer) yang tipe serangannya berbeda dengan unit penyerang dapat melakukan retaliation jika diserang.			Unit yang diserang tidak dapat melakukan retaliation.	
71.	Unit-Retaliation	Menguji apakah unit dapat melakukan retaliation setelah Health-nya ≤ 0 .			Unit tidak dapat melakukan retaliation	
72.	Unit-Retaliation	Menguji apakah King dan Archer dapat melakukan retaliation terhadap unit apapun jika diserang.			Unit King dan Archer selalu dapat melakukan retaliation terhadap unit apapun jika diserang.	
73.	Unit-King	Menguji apakah pada kondisi awal King adalah unit pertama yang ditampilkan			King adalah unit pertama yang ditampilkan.	
74.	Gold-Income-Upkeep	Gold milik player adalah hasil dari jumlah donasi Village dikurang Upkeep setiap unit kecuali King.			Gold = Donasi Village – Upkeep	
75.	Unit-Village	Menguji apakah Village yang baru ditempati berubah kepemilikan (warna) sesuai unit yang menempati			Village berhasil berubah warna sesuai unit yang masuk ke Village itu.	
76.	Movement Point-Village	Menguji apakah movement point dari unit menjadi 0 setelah masuk ke Village.			Movement Point sebuah unit menjadi 0 setelah masuk ke Village.	
77.	Village	Menguji apakah kepemilikan Village tidak akan berubah sampai unit			Village akan berubah kepemilikan hanya saat ditempati oleh	

No.	Fitur yang Dites	Tujuan Testing	Langkah-Langkah Testing	Input Data Test	Hasil yang Diharapkan	Hasil yang Keluar
		player lain masuk ke Village itu.			unit player lain.	

8 Pembagian Kerja dalam Kelompok

No.	Nim-Nama	Coding	Debugging	Testing	Keterangan
1.	13516069-Ivan Jonathan	<13516>	<13516>	<13516>	
2.	13516068-Seperayo	<13516>	<13516>	<13516>	
3.	13516125-Aldo Azali	<13516>	<13516>	<13516>	
4.	13516134-Shevalda Gracielira	<13516>	<13516>	<13516>	
5.	13516137-Hafizh Budiman	<13516>	<13516>	<13516>	

Isi dengan penjelasan pembagian kerja dalam kelompok. Buat dalam bentuk tabel.

9 Lampiran

9.1 Deskripsi Tugas Besar 2

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.

9.2 Notulen Rapat

Isi dengan semua notulen rapat.

9.3 Log Activity Anggota Kelompok

No.	Hari, Tanggal	Kegiatan
1.	Minggu, 12 November 2017	a. Ivan Jonathan Membuat ADT listlinierunit.c, listlinierbuilding.c, gameproperty.c. b. Seperayo Membuat detail Struktur Data (ADT) dalam tugas dan point.c. c. Aldo Azali Membuat map.h dan main.c (versi awal) d. Shevalda Gracielira Membuat queue.c (versi awal) dan debugging turnqueue.c e. Hafizh Budiman

		Melakukan implementasi listlinierunit.c, listlinierbuilding.c, gameproperty.c.
2.	Senin, 13 November 2017	a. Ivan Jonathan Melakukan debugging terhadap ADT listlinierunit.c, listlinierbuilding.c, gameproperty.c. (versi 2) b. Seperayo Menyelesaikan point.c dan implementasinya. c. Aldo Azali (Tidak Hadir). d. Shevalda Gracielira Menyelesaikan turnqueue.c dan implementasinya. e. Hafizh Budiman Memperbaiki tipe bentukan di listlinierunit.c, listlinierbuilding.c, gameproperty.c. (versi 2)
3.		
4.		

Isi dengan log activity tiap anggota kelompok.

9.4 <Lain-Lain>

Berikut hasil asistensi yang kami lakukan selama proses pembuatan tugas ini

1. Asistensi pada Selasa, 07 November 2017
2. Asistensi pada
3. Asistensi pada

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