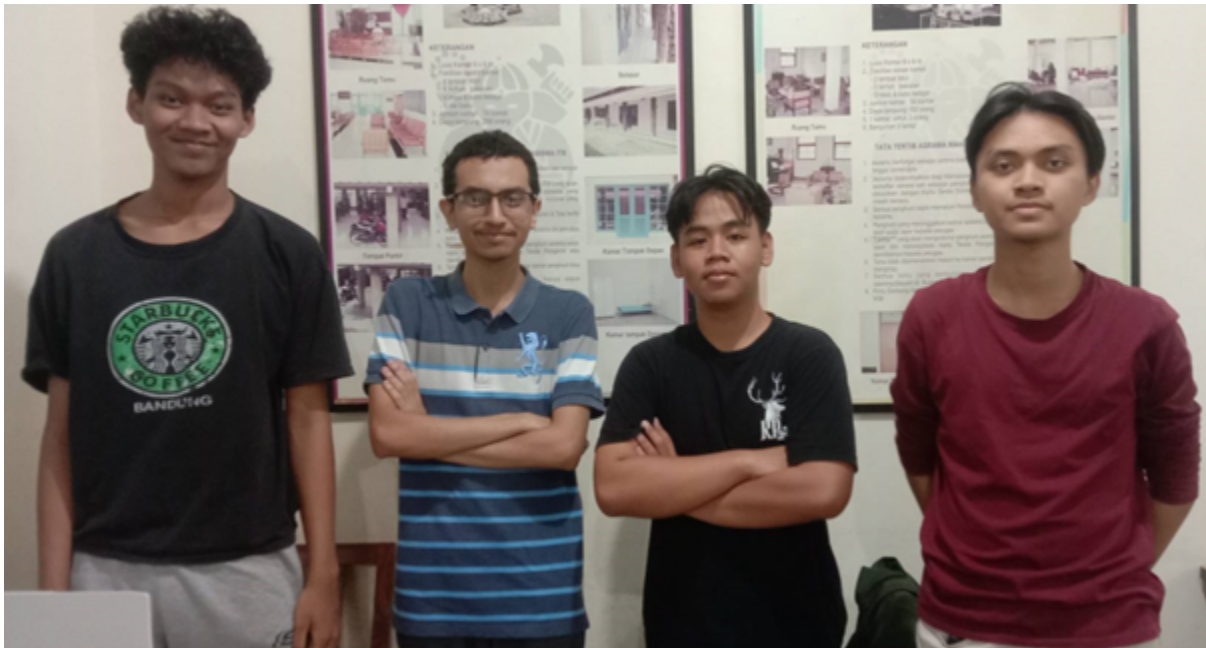


Laporan Tugas Besar
IF2121 Logika Komputasional
Global Conquest: Battle for Supremacy
Semester I Tahun 2023/2024



Disusun Oleh:

Azmi Mahmud Bazeid - 13522109
Nyoman Ganadipa Narayana – 13522066
Randy Verdian – 13522067
Emery Fathan Zwageri – 13522079

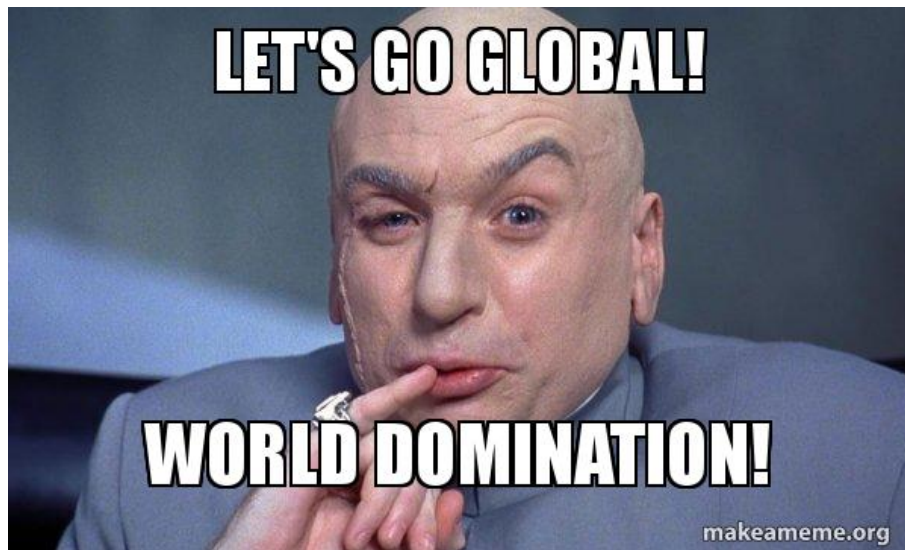
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung
2023

Bab 1

Deskripsi Masalah

1.1. Topik

Membuat sebuah game strategi sebagai seorang programmer yang ingin mendominasi dunia dengan menggunakan bahasa pemrograman deklaratif Prolog (GNU Prolog).



Implementasi tugas besar harus mengandung materi:

1. Rekurens
2. *List*
3. *Cut*
4. *Fail*
5. *Loop*

Tujuan dari tugas besar ini adalah mengkombinasikan berbagai keterampilan dan teknik yang telah dipelajari dalam perkuliahan Logika Komputasional IF2121, pra-praktikum, dan eksplorasi mandiri mengenai Logika Komputasional dan Prolog.

1.2. Alur Permainan

1. Program diinisiasi dengan meminta *input* jumlah pemain. Permainan dimainkan oleh 2 hingga 4 pemain. Pemain lalu bergiliran melakukan *input* nama.
2. *World map* dibagi menjadi 24 wilayah yang terdiri dari 6 benua.
3. Pada awal permainan, pemain mendapatkan total jumlah tentara tergantung banyaknya pemain.
4. Pemain dapat menduduki wilayah dengan mendistribusikan tentara secara bergantian atau memilih sistem untuk mendistribusikan tentara otomatis secara acak. Distribusi tentara manual dilakukan dengan memilih wilayah yang ingin ditempati beserta jumlah tentara yang ingin ditempatkan.
5. Pada setiap giliran, pemain diberikan 1 tentara tambahan setiap 2 wilayah yang dimiliki (contoh: pemain dengan 9 wilayah akan mendapatkan 4 tentara tambahan). Jika pemain berhasil menguasai sebuah benua, maka total tentara tambahan yang diberikan akan bertambah.
6. Pada gilirannya, pemain juga bisa memindahkan tentara pada wilayah yang dimiliki, mendapatkan *risk card* secara acak, serta menyerang wilayah tetangga. Berikut merupakan daftar perintah yang bisa dipanggil pemain:
 - Draft
 - Move
 - Risk
 - Attack
 - EndTurn
7. (Draft) Meletakkan tentara tambahan yang didapatkan oleh pemain pada awal giliran. Dilakukan dengan memilih wilayah tujuan dan jumlah tentara yang ingin dipindah. Peletakan tentara dapat dilakukan hingga semua tentara tambahan sudah diletakkan pada wilayah.
8. (Move) Pemindahan tentara dilakukan dengan memilih wilayah asal, jumlah tentara yang ingin dipindah, serta wilayah tujuan. Pemindahan tentara wajib menyisakan 1 tentara untuk menjaga wilayah yang dimiliki.

Pemindahan tentara dapat dilakukan secara tak terbatas pada setiap giliran.

9. (Risk) *Risk card* didapatkan dengan melempar dadu dan bersifat opsional. Pada setiap giliran, pemain hanya memiliki 1 kesempatan untuk mendapatkan *risk card*.
10. (Attack) Pemain dapat menyerang dengan memilih wilayah penyerang, menentukan jumlah tentara penyerang, dan memilih wilayah tetangga yang akan diserang. Hasil pertempuran antara 2 wilayah ditentukan dengan lemparan dadu. Penyerangan hanya dapat dilakukan sekali setiap giliran.
11. (EndTurn) Giliran pemain diakhiri setelah perintah EndTurn dipanggil.
12. Pemain yang berhasil menguasai setiap wilayah pada papan dan mengeliminasi semua lawan menjadi pemenang.

Bab 2

Implementasi Program

2.1. Map

2.1.1. Penjelasan

Command	Kegunaan	Penggunaan
displayMap	Menampilkan map	Saat player ingin melihat kondisi peta setiap kali pemain mulai melakukan suatu aksi
total_troops(Region, TroopCount)	Mengambil jumlah pasukan di setiap wilayah	Command perantara

2.1.2. Contoh

```
| ?- displayMap.
```

2.2. Initiating

Command	Kegunaan	Penggunaan
readPlayers	Membaca dan memvalidasi jumlah pemain	Command perantara
readPlayerName	Membaca dan menyimpan nama pemain tertentu	Command perantara
sortPlayers	Mengurutkan pemain berdasarkan nilai dadu	Command perantara
printPlayerNames	Mencetak urutan pemain yang diurutkan dan pemain pertama	Command perantara
startGame	Memulai permainan dengan membaca nama pemain, mencetak detail, dan menentukan urutan giliran	Saat pemain ingin memulai game
region_owner(Region, Player)	Mengambil pemain dan wilayah yang dikuasainya	Command perantara
takeLocation(Region)	Mengambil suatu wilayah	Saat pemain ingin mengambil wilayah yang dikuasai

placeTroops(Region, TroopCount)	Melakukan pembagian tentara secara manual	Saat pemain ingin membagikan tentaranya ke wilayahnya secara manual
placeAutomatic	Melakukan pembagian tentara secara otomatis	Saat pemain ingin membagikan tentaranya ke wilayahnya secara otomatis

2.2.2. Contoh

startGame

```
| ?- startGame.
```

takeLocation(Region)

```
| ?- takeLocation(a7).
```

placeTroops(Region, TroopCount)

```
| ?- placeTroops(af1, 2).
```

placeAutomatic

```
| ?- placeAutomatic.
```

2.3. Turn

2.3.1. End Turn

2.3.1.1. Penjelasan

Command	Kegunaan	Penggunaan
endTurn	Mengakhiri giliran pemain saat ini dan beralih ke pemain berikutnya	Saat pemain ingin mengakhiri gilirannya
current_player(NewPlayer)	Mengambil pemain yang sekarang memiliki giliran	Command perantara
region_owned_length(NewPlayer, TotalRegion)	Menghitung total wilayah yang dimiliki oleh pemain yang sekarang memiliki giliran	Command perantara
total_additional_troops(NewPlayer, OldAdditionalTroops)	Mengambil total tentara tambahan yang sudah dimiliki pemain sebelumnya	Command perantara

update_additional_troops(NewPlayer, NewAdditionalTroops)	Memperbarui fakta total additional troops dengan total tentara tambahan yang baru	Command perantara
--	---	-------------------

2.3.1.2. Contoh

```
| ?- endTurn.
```

2.3.2. Draft

2.3.2.1. Penjelasan

Command	Kegunaan	Penggunaan
draft(Region, Troops)	Memperbarui jumlah pasukan suatu wilayah	Saat pemain ingin meletakkan tentara di suatu wilayah
total_troops(Region, InitialTroops)	Mengambil jumlah pasukan awal di wilayah tersebut	Command perantara

2.3.2.2. Contoh

draft

```
| ?- draft(au1,2).
```

2.3.3. Move

Command	Kegunaan	Penggunaan
move(X1, X2, Y)	Melakukan pemindahan tentara dari satu wilayah (X1) ke wilayah lain (X2) dengan jumlah tentara sebanyak Y	Saat pemain ingin melakukan pemindahan tentara
move_count(Now)	Mengambil jumlah langkah saat ini	Command perantara
valid_move(X1, X2, Y, Player)	Mengecek apakah pemindahan tersebut valid	Command perantara
transfer_tentara(X1, X2, Y)	Melakukan pemindahan tentara dari wilayah X1 ke X2 sebanyak Y	Command perantara
print_current_status(X1, X2)	Menampilkan status saat ini	Command perantara

2.3.3.1 contoh

move

```
| ?- move(au2,au1,1).
```

2.3.4. Attack

Command	Kegunaan	Penggunaan
attack	Menyerang wilayah lawan	Saat pemain ingin melakukan penyerangan terhadap wilayah lawan
valid_start_region(Region, Player)	Mengecek apakah wilayah awal serangan yang dipilih oleh pemain valid atau tidak	Command perantara
valid_attacking_troops(Troops, MaxTroops)	Memvalidasi jumlah tentara yang akan digunakan untuk menyerang	Command perantara
valid_target_region(TargetRegion, Player, StartRegion)	Memvalidasi wilayah target serangan harus dimiliki oleh pemain lawan	Command perantara

2.3.4.1 contoh

```
| ?- attack.  
Sekarang giliran Player p2 menyerang.
```

2.3.5. Risk

Command	Kegunaan	Penggunaan
risk	Mengambil secara acak sebuah kartu Risk kepada pemain	Saat pemain ingin mengambil kartu Risk
random_risk_card(Card)	Memilih secara acak sebuah kartu Risk dari daftar kartu	Command perantara
apply_risk_effect	Mengimplementasikan efek yang terkait dengan masing-masing kartu Risk	Command perantara
print_risk_effect	Menampilkan pesan yang menjelaskan efek dari kartu Risk yang dimiliki oleh pemain	Command perantara

2.3.4.2 contoh

2.4. Wilayah

2.4.1. Penjelasan

Command	Kegunaan	Penggunaan
checkLocationDetail(Kode)	Menampilkan detail wilayah	Saat pemain ingin memeriksa detail dari setiap wilayah
code(Kode, X)	Mengambil kode wilayah	Command perantara
region_name(Kode,Y)	Mengambil nama wilayah	Command perantara
region_owner(Kode,Z)	Mengambil pemilik wilayah	Command perantara
total_troops(Kode, W)	Mengambil total tentara di wilayah	Command perantara
writeTetangga(Kode)	Menampilkan daftar wilayah tetangga	Command perantara

2.4.2. Contoh

2.5. Player

2.5.1. Penjelasan

Command	Kegunaan	Penggunaan
checkPlayerDetail(Player)	Menampilkan detail pemain	Saat pemain ingin mengetahui kondisinya
checkPlayerTerritories(Player)	Menampilkan informasi wilayah yang dimiliki oleh seorang pemain di setiap benua	Saat pemain ingin mengetahui informasi wilayahnya
regions_owned_in_continent(Name, Continent, Count)	Menghitung jumlah wilayah yang dimiliki oleh pemain di suatu benua	Command perantara
writeRegionsOwned(Player, Continent)	Menampilkan wilayah-wilayah yang dimiliki oleh pemain di suatu benua tertentu	Command perantara
checkIncomingTroops(Player)	Menampilkan informasi tentang jumlah tentara tambahan	Saat pemain ingin memeriksa jumlah tentara yang akan masuk pada giliran selanjutnya
bonus_from_continent(Continent, Bonus)	Mendapatkan jumlah bonus tentara tambahan di suatu benua	Command perantara

2.5.2. Contoh

Hasil Eksekusi Program

```
| ?- startGame.  
Masukkan jumlah pemain: 2.  
Masukkan nama pemain 1: budi.  
Masukkan nama pemain 2: anto.  
  
budi melempar dadu dan mendapatkan 5.  
anto melempar dadu dan mendapatkan 7.  
  
Urutan pemain berdasarkan nilai dadu: anto - budi.  
anto dapat mulai terlebih dahulu.  
  
Setiap pemain mendapatkan 24 Troops.  
Giliran anto untuk memilih wilayahnya.  
  
(15 ms) yes  
| ?- takeLocation(na1).  
Wilayah berhasil dipilih.  
Giliran Player budi  
  
(15 ms) yes  
| ?- takeLocation(na2).  
Wilayah berhasil dipilih.  
Giliran Player anto  
  
yes  
| ?- takeLocation(na3).  
Wilayah berhasil dipilih.  
Giliran Player budi  
  
yes  
| ?- takeLocation(na4).  
Wilayah berhasil dipilih.  
Giliran Player anto  
  
  
| ?- placeAutomatic.  
  
anto berhasil meletakkan 1 di wilayah na1, sehingga total 2 berada di wilayah ini.  
anto berhasil meletakkan 1 di wilayah na3, sehingga total 2 berada di wilayah ini.  
anto berhasil meletakkan 1 di wilayah na5, sehingga total 2 berada di wilayah ini.  
anto berhasil meletakkan 1 di wilayah sa2, sehingga total 2 berada di wilayah ini.  
anto berhasil meletakkan 1 di wilayah e2, sehingga total 2 berada di wilayah ini.  
anto berhasil meletakkan 1 di wilayah e4, sehingga total 2 berada di wilayah ini.  
anto berhasil meletakkan 1 di wilayah af1, sehingga total 2 berada di wilayah ini.  
anto berhasil meletakkan 1 di wilayah af3, sehingga total 2 berada di wilayah ini.  
anto berhasil meletakkan 1 di wilayah a2, sehingga total 2 berada di wilayah ini.  
anto berhasil meletakkan 1 di wilayah a4, sehingga total 2 berada di wilayah ini.  
anto berhasil meletakkan 1 di wilayah a6, sehingga total 2 berada di wilayah ini.  
anto berhasil meletakkan 1 di wilayah au1, sehingga total 2 berada di wilayah ini.
```

```
| ?- endTurn.  
Player anto mengakhiri giliran.
```

Sekarang giliran Player budi!
Player budi mendapat 6 tentara tambahan.

```
yes  
| ?- endTurn.  
Player budi mengakhiri giliran.
```

Sekarang giliran Player anto!
Player anto mendapat 6 tentara tambahan.

```
yes  
| ?- checkIncomingTroops(p2).
```

Nama	:	budi
Total wilayah	:	12
Jumlah tentara tambahan dari wilayah	:	6
Total tentara tambahan	:	6

```
yes  
| ?- checkPlayerDetail(p2).
```

PLAYER P2

Nama	:	budi
Benua	:	Africa, Asia, Australia, Europe, North America, South America,
Total Wilayah	:	12
Total Tentara Aktif	:	24
Total Tentara Tambahan	:	6

```
| ?- checkPlayerTerritories(p2).  
Nama           : budi
```

Benua Amerika Selatan (1/2)

```
SA1  
Nama           : Amazonas  
Jumlah Tentara : 2
```

Benua Amerika Utara (2/5)

```
NA2  
Nama           : Ontario  
Jumlah Tentara : 2
```

```
NA4  
Nama           : California  
Jumlah Tentara : 2
```

Benua Asia (4/7)

```
A1  
Nama           : Bali  
Jumlah Tentara : 2
```

```
A3  
Nama           : Jeju  
Jumlah Tentara : 2
```

```
A5  
Nama           : Kyoto  
Jumlah Tentara : 2
```

```
A7  
Nama           : Siem Reap  
Jumlah Tentara : 2
```

yes

Pembagian Tugas

NIM	Nama	Tugas	Persentase
13522109	Azmi Mahmud Bazeid	Initiating, Map, Turn	100%
13522066	Nyoman Ganadipa Narayana	Wilayah, Player	100%
13522067	Randy Verdian	Initiating, Laporan	100%
13522079	Emery Fathan Zwageri	Risk, Attack, Laporan	100%