

LAPORAN TUGAS BESAR
IF 2121 LOGIKA KOMPUTASIONAL
Global Conquest: Battle for Supremacy

Kelompok G04 - KeosEnjoyer



Disusun oleh:

Ignatius Jhon Hezkiel Chan	13522029
Daniel Mulia Putra Manurung	13522043
Elbert Chailes	13522045
Benjamin Sihombing	13522054

SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA
INSTITUT TEKNOLOGI BANDUNG

2023

DAFTAR ISI

DAFTAR ISI.....	1
BAB I	
PENDAHULUAN.....	2
1.1 Topik Permasalahan.....	2
1.2 Alur Permainan.....	2
BAB II	
PENJELASAN PROGRAM.....	4
BAB III	
IMPLEMENTASI DAN PENGUJIAN.....	16
BAB IV	
PENUTUP.....	17
4.1 Pembagian dan Persentase Pekerjaan.....	17
4.2 Pranala.....	17
DAFTAR PUSTAKA.....	18

BAB I

PENDAHULUAN

1.1 Topik Permasalahan

Tujuan dari tugas besar ini adalah untuk membuat sebuah permainan strategi sebagai seorang *programmer* yang ingin mendominasi dunia dengan menggunakan bahasa pemrograman deklaratif Prolog (GNU Prolog). Pada implementasinya, harus mengandung materi-materi yang sudah dipelajari selama masa perkuliahan, yaitu materi rekurens, *list*, *cut*, *fail*, dan *loop*. Dengan tugas besar ini, *programmer* dituntut untuk mengkombinasikan berbagai keterampilan dan teknik yang telah dipelajari dalam perkuliahan Logika Komputasional IF2121, pra praktikum, dan eksplorasi mandiri terkait Logika Komputasional dan Prolog.

1.2 Alur Permainan

Berikut merupakan alur permainan yang dibuat oleh *programmer* untuk merealisasi permainan *Global Conquest: Battle for Supremacy*.

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:
 - a. Draft
 - b. Move
 - c. Risk
 - d. Attack
 - e. 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 II

PENJELASAN PROGRAM

Untuk memainkan permainan *Global Conquest: Battle for Supremacy* hasil buatan dari *programmer* yang bersangkutan dengan menggunakan bahasa pemrograman Prolog, pengguna dapat melakukan beberapa perintah / *command* yang dapat dijalankan pada program dalam proses permainan tersebut. Perintah-perintah tersebut akan dituliskan sebagai berikut.

1. `startGame`.

a. Kegunaan perintah

Perintah ini dibuat dengan tujuan untuk memberikan kesempatan bagi pengguna untuk menjalankan perintah untuk memulai permainan. Dengan perintah ini, maka permainan akan secara otomatis dimulai dan akan meminta pengguna untuk memasukkan nama-nama pemain yang ikut bermain. Penentuan urutan pemain akan ditentukan dengan melakukan pengocokan dadu secara otomatis oleh sistem.

b. Skenario Penggunaan

```
Masukkan jumlah pemain: 4.
Masukkan nama pemain 1: kiel.
Masukkan nama pemain 2: daniel.
Masukkan nama pemain 3: ben.
Masukkan nama pemain 4: cai.

cai melempar dadu dan mendapatkan 1.
ben melempar dadu dan mendapatkan 7.
daniel melempar dadu dan mendapatkan 3.
kiel melempar dadu dan mendapatkan 9.

Urutan Pemain: kiel - ben - daniel - cai
kiel dapat memulai terlebih dahulu.
Setiap pemain mendapatkan 12 tentara.

true ?
(15 ms) yes
```

Gambar 2.1. Luaran hasil pemanggilan perintah `startGame`.

2. `displayMap`.

a. Kegunaan perintah

Perintah ini memberikan kesempatan bagi pengguna yang bermain untuk melihat denah atau peta permainan sehingga memberikan pemain dapat menyusun strategi penyerangan dan strategi

Skenario Penggunaan

Gambar 2.2. Luaran pemanggilan perintah `displayMap`.

a. Kegunaan perintah

b. Skenario Penggunaan

5

Gambar 2.3. Luaran pemanggilan perintah takeLocation(KodeWilayah)
jika wilayah berhasil diambil oleh pemain

```
| ?- takeLocation(na2).  
Wilayah sudah dikuasai. Tidak bisa mengambil.  
Giliran daniel untuk memilih wilayahnya.  
yes
```

Gambar 2.4. Luaran pemanggilan perintah takeLocation(KodeWilayah)
jika gagal karena wilayah telah dimiliki oleh pemain lain

4. placeTroops(KodeWilayah, BanyakTroops).

a. Kegunaan perintah

Perintah ini bertujuan untuk memberikan kesempatan bagi pemain untuk meletakkan tentara-tentara tambahan yang dimilikinya ke wilayah-wilayah yang telah dikuasai olehnya. Dengan perintah ini, pemain dibebaskan untuk meletakkan sebanyak mungkin tentara pada suatu wilayah, dengan syarat bahwa tentara tambahan yang dimiliki oleh pemain tersebut cukup. Perintah ini menerima dua argumen, yaitu kode wilayah yang dituju untuk meletakkan tentara-tentaranya, dan banyak tentara yang hendak diletakkan pada wilayah tersebut juga.

b. Skenario Penggunaan

```
| ?- placeTroops(na1,5).  
kiel meletakkan 5 tentara di wilayah na1.  
Terdapat 1 tentara yang tersisa.  
Giliran ben untuk meletakkan tentaranya.  
yes
```

Gambar 2.5. Luaran program jika dilakukan pemanggilan perintah
placeTroops(KodeWilayah,BanyakToops) dan berhasil

5. placeAutomatic.

a. Kegunaan perintah

Perintah ini merupakan fitur tambahan untuk membantu pengguna untuk melakukan penempatan tentara pada wilayah-wilayah yang dimiliki oleh pemain tersebut. Konsep penempatan tentara otomatis

ini dilakukan dengan cara distribusi acak. Perintah ini hanya dapat dijalankan pada fase setelah melakukan pengambilan wilayah oleh masing-masing pemain. Oleh karena itu, jika pemain merasa bingung, perintah ini akan membantu pemain untuk melakukan distribusi secara acak ke semua wilayah yang dimiliki oleh pemain tersebut dan terdapat kemungkinan bahwa terdapat wilayah yang tidak mendapatkan tentara.

b. Skenario Penggunaan

```
| ?- placeAutomatic.  
kiel meletakkan 1 tentara di wilayah na5  
Seluruh tentara kiel sudah diletakkan.  
Giliran kiel untuk meletakkan tentaranya.  
Seluruh pemain telah meletakkan sisa tentara.  
Memulai permainan.  
yes
```

Gambar 2.6. Luaran program jika dilakukan pemanggilan perintah placeAutomatic.

6. endTurn.

a. Kegunaan perintah

Perintah ini akan memberikan kesempatan kepada pemain untuk melakukan pengakhiran giliran yang dimilikinya sehingga ia tidak perlu melakukan perintah tambahan. Perintah ini akan memberikan efek samping juga kepada pemain selanjutnya, yang mana pemain selanjutnya mendapatkan bonus tentara yang seharusnya sesuai dengan syarat-syarat bonus tentara yang telah ditentukan angka-angkanya. Bonus-bonus yang dapat didapatkan adalah bonus benua, bonus wilayah, dan bonus per putaran.

b. Skenario Penggunaan

```
| ?- endTurn.  
Player kiel mengakhiri giliran.  
Sekarang giliran Player kiel!  
Player kiel terdampak SUPPLY CHAIN ISSUE!  
Player kiel tidak mendapatkan tentara tambahan.  
yes
```


Gambar 2.7. Tampilan luaran program jika pengguna memutuskan untuk mengakhiri giliran dengan menggunakan perintah endTurn.

7. **draft(KodeWilayah, BanyakTroops).**

a. **Kegunaan perintah**

Perintah ini bertujuan untuk memberikan kesempatan bagi pemain yang menggunakan perintah ini untuk menempatkan sejumlah tentaranya sebanyak BanyakTroops ke KodeWilayah tertentu. Perintah ini dapat dilakukan secara tidak terbatas, dan pemain tidak melakukan perintah draft ini hanya ketika ia sudah tidak memiliki tentara tambahan. Perlu diingat bahwa penggunaan perintah hanya dapat dilakukan pada wilayah yang dimiliki oleh pemain yang melakukan perintah draft ini.

b. **Skenario Penggunaan**

```
| ?- draft(sa2,5).  
Player chai meletakkan 5 tentara tambahan di wilayah sa2  
  
Tentara total di sa2: 6  
Jumlah Pasukan Tambahan Player chai: 7
```

Gambar 2.8. Tampilan luaran program jika melakukan perintah draft ke wilayah sa2 sebanyak 5 tentara

8. **move(KodeWilayah1, KodeWilayah2, BanyakTroops).**

a. **Kegunaan perintah**

Perintah ini memberikan kesempatan bagi pemain untuk melakukan pemindahan tentaranya dari satu wilayah ke wilayah lainnya (KodeWilayah1 ke KodeWilayah2) sebanyak BanyakTroops. Perlu diingat bahwa, pemain hanya dapat melakukan pemindahan tentara dari wilayah yang satu ke wilayah lainnya yang dimilikinya. Maksimal penggunaan perintah move oleh setiap pemain pada gilirannya hanyalah sebanyak tiga kali.

b. **Skenario Penggunaan**

```
| ?- move(e1,e3,3).
kiel memindahkan 3 tentara dari e1 ke e3.

Jumlah tentara di e1: 8
Jumlah tentara di e3: 4

true ?
yes
```

Gambar 2.9. Tampilan luaran program jika menggunakan perintah move untuk memindahkan pemain dari wilayah e1 ke e3 sebanyak 3 tentara

9. attack.

a. Kegunaan perintah

Perintah ini memberikan kesempatan bagi pemain yang satu untuk melakukan penyerangan terhadap wilayah pemain lainnya. Dalam proses penyerangan ini, akan dilakukan pengocokan dadu oleh masing-masing pemain dengan jumlah dadu sebanyak tentara yang dimiliki oleh masing-masing pemain pada wilayah tersebut yang terjadi penyerangan. Perlu diingat bahwa setiap pemain hanya dapat melakukan perintah attack sebanyak satu kali pada setiap giliran yang dimilikinya. Dalam attack ini, juga terdapat pengaruh *risk card* yang dapat mempengaruhi persentase kemenangan seorang pemain ketika melakukan aksi penyerangan. Jika pemain yang menyerang menang, maka seluruh tentara lawan pada wilayah tersebut akan hangus, dan berlaku sebaliknya. Pemenang atas peperangan antara dua pemain tersebut pada wilayah tersebut akan menentukan siapa pemilik dari wilayah tersebut.

b. Skenario Penggunaan

```
| ?- attack.
Sekarang giliran Player kiel menyerang.
Pilihlah daerah yang ingin Anda mulai untuk melakukan penyerangan: e2.
Wilayah tersebut bukan milikmu. Pilih yang lain!
Pilihlah daerah yang ingin Anda mulai untuk melakukan penyerangan: e1.
Player kiel ingin memulai penyerangan dari daerah e1
Dalam daerah e1, Anda memiliki sebanyak 8 tentara.
Masukkan banyak tentara yang akan bertempur: 8.
Banyak input tentara tidak valid. Silahkan coba lagi.
Masukkan banyak tentara yang akan bertempur: 7.
Player kiel mengirim sebanyak 7
1. e2
2. na5
Pilih: 1.
Perang telah dimulai.
Player kiel
Dadu 7: 3
Dadu 6: 1
Dadu 5: 1
Dadu 4: 6
Dadu 3: 3
Dadu 2: 1
Dadu 1: 1
Total: 16
Player chai
Dadu 1: 1
Total: 1
Player kiel menang! Wilayah e2 sekarang dikuasai oleh Player kiel
Silahkan tentukan banyak tentara yang menetap di wilayah tersebut: 3.
Tentara di wilayah e1: 5
Tentara di wilayah e2: 3

true ?

(31 ms) yes
```

Gambar 2.10. Tampilan luaran program jika pemain melakukan penyerangan dengan menggunakan perintah attack.

10. risk.

a. Kegunaan perintah

Perintah ini memberikan kesempatan bagi pemain untuk melakukan *gamble* untuk mendapatkan *risk* yang memberikan keuntungan baginya ataupun sebaliknya. Perintah ini hanya dapat dilakukan sebanyak satu kali setiap gilirannya. Pada gilirannya, pemain dapat memanggil perintah *risk* untuk mendapatkan *risk card* secara acak. Perintah risk ini bersifat opsional.

b. Skenario Penggunaan

```
| ?- risk.
Player kiel mendapatkan risk card SUPPLY CHAIN ISSUE.
Pada giliran berikutnya, pemain tidak mendapatkan tentara tambahan.

yes
| ?- |
```

Gambar 2.11. Tampilan luaran program jika pemain mendapatkan sebuah *risk card* dengan memanggil perintah risk.

11. checkLocationDetail(KodeWilayah).

a. Kegunaan perintah

Perintah ini akan memberikan kesempatan bagi pemain untuk melihat detail informasi dari KodeWilayah yang hendak dilakukan penelusuran. Perintah ini tidak memiliki batas penggunaan. Luaran dari penggunaan perintah ini adalah menampilkan kode daripada wilayah, nama wilayah, pemilik wilayah, total tentara yang terdapat pada wilayah tersebut, dan wilayah-wilayah yang bertetangga dengan wilayah tersebut.

b. Skenario Penggunaan

```
| ?- checkLocationDetail(na1).  
Kode                               :na1  
Nama                               :canada  
Pemilik                           :kiel  
Total Tentara                     :1  
Tetangga                          :3
```

Gambar 2.12. Tampilan luaran program jika melakukan pemanggilan perintah untuk melihat informasi detail lokasi kode wilayah na1

12. checkPlayerDetail(LabelPemain).

a. Kegunaan perintah

Perintah ini bertujuan untuk menampilkan keluaran berupa informasi yang dimiliki oleh seorang pemain tertentu. Penggunaan perintah ini dapat dilakukan oleh siapapun dan tidak memiliki batasan penggunaan. Informasi-informasi yang ditampilkan berupa nama pemain, benua yang dikuasai oleh pemain tersebut, total wilayah, total tentara aktif, dan total tentara tambahan yang dimiliki oleh pemain tersebut.

b. Skenario Penggunaan

```
| ?- checkPlayerDetail(p1).  
PLAYER P1  
Nama                               :kiel  
Benua                              :-  
Total Wilayah                     :6  
Total Tentara Aktif               :12  
Total Tentara Tambahan            :0
```

Gambar 2.13. Tampilan luaran program jika melakukan pengecekan informasi player 1 dengan menggunakan perintah `checkPlayerDetail(LabelPemain)`.

13. `checkPlayerTerritories(LabelPemain)`.

a. Kegunaan perintah

Perintah ini digunakan untuk menampilkan informasi terkait teritori-teritori wilayah yang dimiliki oleh seorang pemain yang dituju. Perintah ini tidak memiliki batasan penggunaan dan dapat digunakan oleh siapapun. Perintah ini akan menampilkan informasi berupa nama pemain yang ditelusuri, wilayah-wilayah yang dimiliki oleh pemain tersebut sehingga pemain yang melakukan perintah ini dapat mengetahui wilayah apa saja dan berapa bagian dari benua yang dimiliki oleh pemain tersebut.

b. Skenario Penggunaan

```
| ?- checkPlayerTerritories(p2).  
Nama :chai  
Benua afrika (2/3)  
af3  
Nama : ghana  
Jumlah tentara : 1  
  
af1  
Nama : nigeria  
Jumlah tentara : 4  
  
Benua amerikaselatan (1/2)  
sa2  
Nama : argentina  
Jumlah tentara : 10  
  
Benua amerikautara (3/5)  
na1  
Nama : canada  
Jumlah tentara : 1  
  
na3  
Nama : mexico  
Jumlah tentara : 1  
  
na5  
Nama : jamaica  
Jumlah tentara : 1  
  
Benua asia (3/7)  
a2  
Nama : indonesia  
Jumlah tentara : 1
```

Gambar 2.14. Tampilan luaran program jika pemain melakukan perintah `checkPlayerTerritories` pada pemain tertentu

14. `checkIncomingTroops(LabelPemain)`.

a. Kegunaan perintah

Perintah ini bertujuan untuk melihat jumlah tentara yang akan masuk pada seorang pemain tertentu ketika sudah mencapai gilirannya. Perintah ini tidak memiliki batas penggunaan dan dapat digunakan siapapun. Dengan perintah ini, pemain akan mendapatkan tampilan keluaran berupa informasi terkait pemain yang dituju dengan informasi total wilayah yang dimiliki pemain tersebut, jumlah tentara bonus yang didapatkan akibat banyak wilayah yang dimilikinya, bonus dari benua-benua yang dimilikinya, dan total tentara tambahan yang akan didapatkan oleh pemain tersebut ketika sudah mencapai giliran bermainnya.

b. Skenario Penggunaan

```
| ?- checkIncomingTroops(p1).  
Nama : kiel  
Total wilayah : 13  
Jumlah tentara tambahan dari wilayah : 6  
Total tentara tambahan : 6  
  
yes  
| ?- |
```

Gambar 2.15. Tampilan luaran program jika pengguna melakukan perintah checkIncomingTroops pada pemain spesifik

15. cheatTambahTentara(JumlahTentara).

a. Kegunaan perintah

Perintah ini tergolong pada perintah **Cheats**. Perintah ini bertujuan untuk menabahkan tentara-tentara tambahan sebanyak yang player yang sedang bermain inginkan. Perintah ini akan menampilkan message yang menunjukkan bahwa tentara sebanyak yang diinginkan player telah ditambahkan sebagai tentara tambahan, dan message yang menandakan bahwa player telah menggunakan cheats. Penggunaan perintah cheat ini tidak akan memberikan konsekuensi.

b. Skenario Penggunaan

```
~  
| ?- cheatTambahTentara(200).  
Berhasil menambahkan 200 tentara kepada player tes  
JANGAN CURANG WOIIII!!!  
  
yes  
| ?-
```

Gambar 2.16. Tampilan luaran program jika pengguna melakukan perintah `cheatTambahTentara`

16. `cheatAkuisisiWilayah(WilayahAkuisisi)`.

a. Kegunaan perintah

Perintah ini tergolong pada perintah **Cheats**. Perintah ini bertujuan untuk melakukan akuisisi wilayah terhadap wilayah yang dipilih, tanpa memperhatikan apakah wilayah tersebut telah diakuisisi oleh pemain lain ataupun tidak. Perintah ini akan menampilkan message yang menandakan bahwa wilayah akuisisi pilihan player telah diakuisisi oleh player tersebut. Penggunaan perintah `cheat` ini tidak akan memberikan konsekuensi.

b. Skenario Penggunaan

```
yes
| ?- mapInformation(X, na1, Y).
X = kiel
Y = 2 ?

yes
| ?- currentPlayer(X).
X = tes

(15 ms) yes
| ?- cheatAkuisisiWilayah(na1).
Wilayah na1 diakuisisi oleh tes

true ?

yes
| ?- |
```

Gambar 2.17 Tampilan luaran program jika pengguna melakukan perintah `cheatAkuisisiWilayah`, dapat dilihat daerah `na1` dimiliki oleh pemain lain yang bukanlah pemain yang sedang dalam giliran.

17. `cheatAmbilKartu`.

a. Kegunaan perintah

Perintah ini tergolong pada perintah **Cheats**. Perintah ini bertujuan untuk memberikan pemain kemampuan untuk mengambil kartu sesuai keinginan pemain tersebut. Ketika perintah ini dipanggil, program akan menampilkan pilihan-pilihan kartu yang dapat dipilih oleh pemain dan akan meminta pemain untuk memasukkan angka sesuai dengan pilihan kartu yang diinginkan. Ketika pemilihan kartu valid,

program akan menampilkan message yang menunjukkan bahwa pemain telah berhasil mendapatkan kartu yang diinginkan. Penggunaan perintah cheat ini tidak akan memberikan konsekuensi.

b. Skenario Penggunaan

```
yes
| ?- cheatAmbilKartu.
Pilih kartu risk yang ingin diambil
1. Ceasefire Order
2. Super Soldier Serum
3. Auxiliary Troops
4. Rebellion
5. Disease Outbreak
6. Supply Chain Issue
Masukan angka: 1.
Anda mendapat kartu risk Ceasefire Order
yes
| ?-
```

Gambar 2.18 Tampilan luaran program jika pengguna melakukan perintah cheatAmbilKartu

BAB III

IMPLEMENTASI DAN PENGUJIAN

BAB IV

PENUTUP

4.1 Pembagian dan Persentase Pekerjaan

NIM	NAMA	TUGAS
13522029	Ignatius Jhon Hezkiel Chan	placeTroops, placeAutomatic, endTurn, draft, move, attack, risk,
13522043	Danie Mulia Putra Manurung	displayMap, attack
13522045	Elbert Chailes	startGame, checkLocationDetail, checkPlayerDetail, checkPlayerTerritories, checkIncomingTroops
13522054	Benjamin Sihombing	takeLocation, placeTroops, placeAutomatic, bonus cheat

4.2 Pranala

Untuk menyelesaikan tugas besar ini, kami menggunakan *github repository* untuk membantu proses pengembangan dan koordinasi antar *developer* sehingga pranala pengembangan tersebut terlampir sebagai berikut.

<https://github.com/GAIB21/tugas-besar-if2121-logika-komputasional-2023-keos-enjoyer>

DAFTAR PUSTAKA

Diaz, Daniel. “GNU-Prolog Manual.” *GNU Prolog*,

<http://www.gprolog.org/manual/gprolog.html>. Accessed 28 November 2023.