

Aplikasi Teori Graf, Kombinatorial, dan Pohon dalam Permainan Daring Ragnarok M: Eternal Love

Winston Wijaya - 13517018
 Program Studi Teknik Informatika
 Sekolah Teknik Elektro dan Informatika
 Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
 13517018@std.stei.itb.ac.id

Abstract—Dalam kehidupan sehari-hari, ada banyak sekali aplikasi graf, kombinatorial, dan tree yang kadang tidak disadari, namun telah diterapkan, salah satunya pada permainan daring. Salah satu permainan daring yang tengah populer di akhir tahun 2018 ini adalah Ragnarok M: Eternal Love yang termasuk kategori MMORPG. Pada makalah ini akan membahas mengenai aplikasi graf, tree, dan kombinatorial pada game daring Ragnarok M: Eternal Love

Keywords—Graf, Kombinatorial, Online Game, Tree



Gambar 1.2 Poster Game Daring Ragnarok M: Eternal Love^[5]

I. PENDAHULUAN

Kian berkembangnya ilmu pengetahuan dan teknologi memacu pertumbuhan dan perkembangan berbagai sektor dalam kehidupan, salah satunya pada industri permainan. Pada era digital ini sudah banyak jenis permainan yang muncul salah satunya permainan daring yang sekarang sudah sangat populer dengan berbagai *genre* yang bisa dimainkan.

Secara singkat, permainan daring (*online game*) dapat didefinisikan sebagai permainan yang dimainkan oleh banyak *player* melalui berbagai bentuk jaringan komputer, yakni internet^[2]. *Online game* sendiri mencakup *game* sederhana seperti *game* berbasis teks (*text-based games*) sampai *game* yang melibatkan grafis yang kompleks dan dunia virtual yang ditempati banyak pemain secara bersamaan^[2].

Permainan daring mulai berkembang sejak tahun 1984 saat BITNET mengembangkan MAD (*Multi Access Dungeon*). Salah satu area dari SGI Dogfight yakni pada area *game* merupakan *game* pertama yang memanfaatkan *Internet Protocol* yang diinisiasi sekitar tahun 1985.

Makin berkembangnya teknologi sekarang mendorong para *developer online game* untuk merilis produk-produk yang mampu bersaing di pasar, salah satunya GRAVITY Co., Ltd yang berbasis di Korea. Salah satu *game* terbaru yang dirilis *developer* tersebut, Ragnarok M: Eternal Love, per 5 November 2018 masih menduduki peringkat pertama sebagai *The Highest Grossing Ranking* pada *Google Play* dan *Apple App Store* di Thailand, Filipina, dan Indonesia.

| The Highest Grossing Ranking in Southeast Asia (as of 5 November 2018) | | |
|--|-------------|-----------------|
| | Google play | Apple App Store |
| Thailand | No. 1 | No. 1 |
| The Philippines | No. 1 | No. 1 |
| Indonesia | No. 1 | No. 1 |

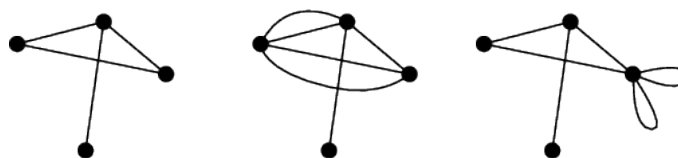
Gambar 1.1 Peringkat Ragnarok M: Eternal Love per 5 November 2018^[4]

Keunikan dari *game* ini adalah walaupun tergolong MMORPG *game* pada *mobile phone*, ada berbagai fitur dalam *game* yang mendekati fitur-fitur *game* di PC (*Personal Computer*) yang menjadi perbedaan *game* ini dengan *game* MMORPG lainnya pada *device* yang sama.

II. DASAR TEORI

2.1 Graf

Graf dapat didefinisikan sebagai pasangan himpunan (V, E) yang digunakan untuk merepresentasikan objek-objek diskrit dan hubungan antara objek-objek tersebut. Graf ditulis dengan notasi $G = (V, E)$ dimana V (*vertices* atau *node*) adalah himpunan tidak kosong dari simpul-simpul dan E (*edge* atau *arcs*) adalah himpunan sisi-sisi yang menghubungkan sepasang simpul.



simple graph

nonsimple graph with multiple edges

nonsimple graph with loops

(a) G1

(b) G2

(c) G3

Gambar 2.1.1 Tiga buah graf (a) graf sederhana, (b) graf tidak sederhana dengan sisi-ganda, dan (c) graf semu^[6]

Pengelompokan graf dapat ditentukan berdasarkan ada tidaknya sisi ganda atau sisi kalang, jumlah simpul, atau orientasi arah pada sisi. Dalam makalah ini, graf yang

dibahas adalah graf berdasarkan ada tidaknya gelang atau sisi ganda, graf berdasarkan orientasi arah pada sisi, dan graf berbobot.

Berdasarkan ada tidaknya gelang atau sisi ganda, graf terbagi menjadi 2, yakni:

1. Graf sederhana (*simple graph*)

Graf sederhana merupakan graf tak-berarah yang tidak memiliki sisi-ganda dan gelang. Salah satu contoh graf berarah adalah gambar 2.1.1 (a).

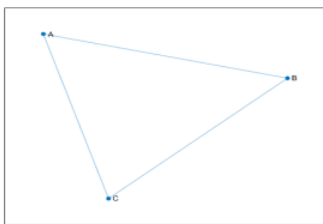
2. Graf tak-sederhana (*unsimple graph*)

Graf tak-sederhana merupakan graf tak-berarah yang memiliki sisi-ganda atau gelang. Terdapat 2 jenis graf tak-sederhana, yakni **graf ganda** (*multigraph*) dan **graf semu** (*pseudograph*). Graf ganda merupakan graf dengan sisi ganda yang menghubungkan sepasang simpul lebih dari dua buah seperti gambar 2.1.1 (b), sedangkan graf semu merupakan graf yang mengandung gelang (*loop*) seperti sebuah jalur yang menghubungkan ke diri sendiri misalkan pada gambar 2.1.1 (c).

Berdasarkan orientasi arah pada sisi, graf terbagi menjadi 2, yaitu:

1. Graf tak-berarah (*undirected graph*)

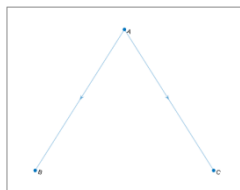
Graf tak-berarah merupakan graf yang memiliki simpul-simpul dengan sisi yang tidak mempunyai orientasi arah. Simpul yang tidak berarah menandakan hubungan dua arah (*two-way relationship*). Pada graf ini urutan simpul yang terhubung oleh sisi tidak diperhatikan, misal $(u, v) = (v, u)$. Contoh graf tak-berarah:



Gambar 2.1.2 graf tak-berarah^[7]

2. Graf berarah (*directed graph* atau *digraph*)

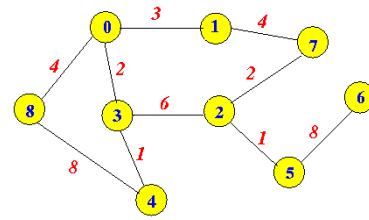
Graf berarah merupakan graf yang memiliki simpul-simpul dengan sisi mempunyai orientasi arah. Simpul yang berarah menandakan hubungan satu arah (*one-way relationship*). Pada graf ini, urutan simpul yang terhubung oleh sisi diperhatikan, misal $(u, v) \neq (v, u)$. Contoh graf tidak berarah:



Gambar 2.1.3 graf berarah^[7]

Graf berbobot merupakan graf yang setiap sisinya

memiliki bobot dengan bobot pada setiap sisi bergantung pada permasalahan yang dimodelkan. Contoh dari graf berbobot:



Gambar 2.1.4 graf berbobot^[8]

2.2 Kombinatorial

Kombinatorial merupakan salah satu cabang ilmu matematika yang mempelajari pengaturan objek-objek tanpa harus mencoba mencacah atau menghitung semua kemungkinan susunannya^[1]. Dalam makalah ini, secara spesifik, bagian dari kombinatorial yang bersangkutan adalah permutasi dan peluang.

1 Permutasi

Permutasi merupakan jumlah pengaturan dari objek-objek berbeda dimana urutan kemunculan dari objek-objek tersebut berpengaruh. Notasi dari permutasi adalah:

$$P(n, r) = \frac{n!}{(n - r)!}$$

Contoh persoalan permutasi: Terdapat 4 kandidat A, B, C, dan D. Akan dipilih ketua kelas dan wakil ketua kelas dari keempat kandidat tersebut. Berapa banyak susunan yang mungkin?

Maka banyak susunan yang mungkin adalah:

$$P(4,2) = \frac{4!}{(4 - 2)!} = \frac{4 \cdot 3 \cdot 2!}{2!} = 12$$

2 Peluang

Peluang merupakan harapan terjadinya suatu kejadian berdasarkan banya kejadian yang diharapkan terhadap banyak kejadian keseluruhan. Misal peluang sebuah kejadian adalah x, maka $0 < x < 1$.

Contoh persoalan yang berkaitan dengan peluang: Misal seseorang melempar 2 buah dadu, berapakah peluang dadu pertama adalah bilangan genap dan dadu kedua bilangan genap yang berbeda?

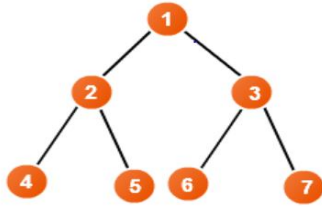
Penyelesaian dari persoalan tersebut adalah sebagai berikut:

Misal P adalah peluang kejadian dadu pertama bilangan genap dan dadu kedua adalah bilangan genap yang berbeda, maka nilai dari P adalah:

$$P = \frac{3}{6} \cdot \frac{2}{6} = \frac{1}{6}$$

2.3 Tree

Pohon (*tree*) merupakan salah satu bentuk graf tak-berarah terhubung tanpa ada sirkuit yang digunakan untuk menggambarkan hubungan yang hierarkis antar elemen-elemennya.



Gambar 2.3.1 tree^[9]

Dalam makalah ini, *tree* yang secara spesifik hal yang berkaitan adalah pencarian pohon merentang minimum (*minimum spanning tree*) dari suatu graf terhubung-berbobot. Ada 2 algoritma yang dapat digunakan, yakni algoritma Prim dan algoritma Kruskal.

1. Algoritma Prim^[1]

Langkah-langkah yang dilakukan dengan algoritma prim adalah sebagai berikut:

1. Ambil sisi dari graf **G** yang berbobot minimum, masukkan ke dalam **T**.
2. Pilih sisi (*u,v*) yang mempunyai bobot minimum dan bersisian dengan simpul **T**, tetapi tidak membentuk sirkuit di **T**.
3. Ulangi langkah kedua sebanyak *n-2* kali.

2. Algoritma Krausal^[1]

1. Urutkan sisi graf berdasarkan bobot dari terkecil hingga terbesar dan ada sebuah tree **T** yang kosong.
2. Pilih sisi (*u, v*) dengan bobot minimum yang tidak membentuk sirkuit di **T**, kemudian tambahkan sisi tersebut ke dalam **T**.
3. Ulangi langkah kedua sebanyak *n-2* kali.

III. PENERAPAN GRAF, KOMBINATORIAL, DAN POHON PADA PERMAINAN DARING RAGNAROK M: ETERNAL LOVE

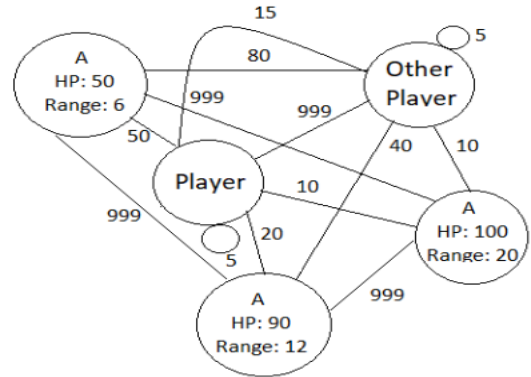
3.1 Fitur *Auto-Skill*

Fitur ini merupakan salah satu fitur yang sering digunakan pemain untuk *farming* maupun *leveling* saat *player* tersebut sedang melakukan pekerjaan lain dan dapat meninggalkan media yang digunakan untuk bermain tetap menyala. Saat menggunakan fitur ini *player* terlebih dahulu menyiapkan skill yang hendak dipakai ke dalam *auto skill slot*. Ada beberapa skill yang bisa dimasukkan, diantaranya skill untuk menyerang dan skill untuk memberikan *buff* (status tambahan pada karakter) pada diri sendiri maupun *player* lain.



Gambar 3.1.1 *auto skill slot*^[11]

Setelah itu *player* dapat memulai *auto-skill* setelah memilih *lock* pada salah satu atau semua monster di sekitarnya. Misal *player* tersebut memilih *lock* pada monster A untuk diserang, maka karakter dari *player* tersebut akan menyerang monster A dengan 2 kriteria, yakni jarak terdekat dan HP (*Health Point*) terbanyak dari *player*. Posisi *player*, *player* lain, dan monster dapat dimodelkan dengan graf tak-sederhana berbobot sebagai berikut:



Gambar 3.1.2 permodelan graf dalam *auto skill*

Bobot 999 menandakan antar *player* maupun antar monster tidak menyerang satu sama lain. Bila *player* meletakkan *skill buff* pada *auto skill slot*, maka *skill* tersebut akan diprioritaskan dengan bobot terkecil jika *skill* tersebut belum diaktifkan dan bobot tersebut akan menjadi bobot terbesar yang kurang dari bobot pada sisi antar *player* jika *skill* tersebut masih aktif. Hal tersebut juga berlaku bila ada *skill buff* yang ingin diberikan ke *player* lain, bobot pada sisi antar *player* akan berbeda saat *skill* belum dan sudah aktif. *Player* akan menyerang monster dengan bobot terkecil sesuai dengan posisi antar *player*, *player* lain, dan monster pada graf terus menerus dengan algoritma pencarian rute terpendek hingga *player* kehabisan HP atau *auto skill* dimatikan. Dalam kondisi ideal (tidak ada *player* lain yang berada di dekat *player* tersebut dan monster A berada di dekat *player* tersebut) hanya akan terbentuk 1 *minimum spanning tree* saja, namun dalam kondisi yang tidak ideal dapat terbentuk *minimum spanning tree* yang berbeda tergantung dengan kondisi saat itu.

3.2 Item *Fly Wing*

Salah satu *item* yang sering digunakan oleh para *player* yang ingin melakukan *hunt MVP* (*boss monster* pada seutu *map*) dan *miniboss* karena dinilai lebih efektif dibandingkan berjalan-jalan di *map* sampai bertemu dengan target yang dicari.



Fly wings can be cut off and made into magical objects that can move anywhere on a map.

Gambar 3.2 *fly wing*^[12]

Item ini digunakan untuk memindahkan posisi (*teleport*) suatu karakter pada koordinat tertentu di *map*. Berikut adalah salah satu contoh *map* di *game*:



Gambar 3.2.1 *geffen map*^[13]

Map tersebut dalam bentuk 2 dimensi terdiri dari sebuah matriks berukuran $n \times n$ yang indeksnya terdiri dari absis dan ordinat. Saat *item* tersebut digunakan, akan ada fungsi yang *generate* angka *random* untuk absis dan ordinat yang memunculkan karakter *player* pada koordinat tersebut di matriks. Misal dalam 1 map diatas terdiri dari matriks berukuran 60×60 . Maka saat menggunakan *fly wing* akan ada kombinasi posisi *player* misal Pos sebanyak:

$$Pos = P(60,1) \times P(60,1) - 1 - X = 60 \times 60 - 1 - X = 3599 - X$$

Banyak posisi dikurangi 1 karena saat menggunakan item tersebut tidak ada perpindahan posisi yang sama dengan posisi *player* sebelum menggunakan *item* dan dikurangi X lagi karena ada X buah koordinat di *map* yang tidak dapat ditempati oleh *player*. Misal pada *map* berikut:



Gambar 3.2.2 *geffen map*^[13]

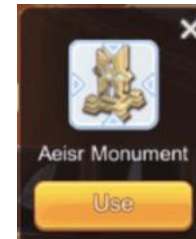
Dalam segiempat tersebut, daerah berwarna hitam merupakan daerah pada *map* yang tidak dapat diakses berikut walaupun menggunakan *item* apapun.

Misal *player* ingin melakukan *hunt* terhadap MVP ataupun *miniboss* pada sebuah *map* berupa matriks berukuran 100×100 , *item* tersebut akan berhenti digunakan saat karakter *player* muncul di suatu koordinat yang dekat dengan monster tersebut dalam jarak sekitar $3\sqrt{2}$ dari posisi *player*. Misal peluang *player* tersebut bertemu dengan targetnya adalah *ChM* dengan koordinat yang tidak bisa diakses X, maka *ChM* dengan estimasi sebesar:

$$ChM = \frac{P(100,7) \times P(100,7)}{P(100,1) \times P(100,1) - X} \times 2$$

3.3 Sistem Rune

Untuk para *player* yang sudah memenuhi beberapa syarat, yakni sudah tergabung dalam sebuah *guild* dan sudah menyelesaikan *quest* untuk mendapatkan *item* bernama *Aeisir Monumen*, rune dapat diaktifkan untuk memperkuat karakter *player* dengan berbagai jenis tambahan status.



Gambar 3.3.1 *Aeisir Monument*^[14]

Saat *player* menekan *Use* maka akan muncul tampilan sebagai berikut:



Gambar 3.3.2 *rune system*^[14]

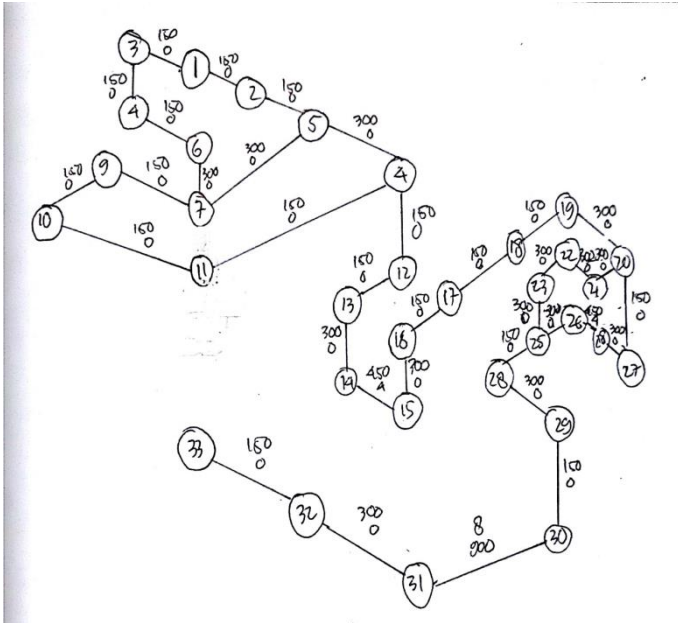
Sistem *rune* tersebut jika dilihat dengan baik diimplementasikan dalam bentuk graf. Untuk *player* yang terkadang malas untuk mentukan rute sendiri, sistem dalam game sudah mengimplementasikan algoritma untuk menentukan jalur terpendek (*shortest path*) menuju target yang diinginkan, seperti berikut ini:



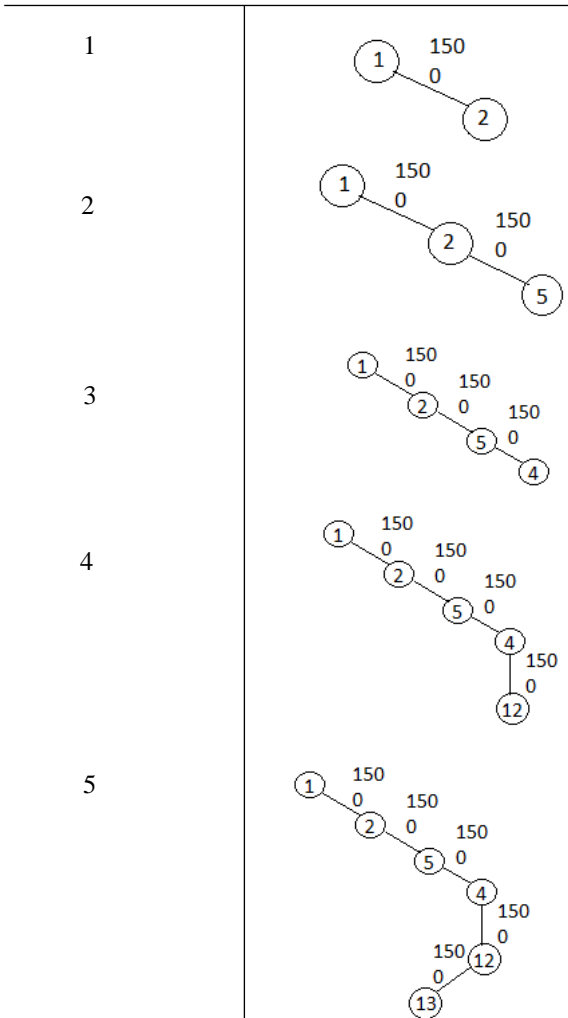
Gambar 3.3.3 pemilihan jalur terpendek menuju target

Pada gambar tersebut terlihat ada 2 buat jalur menuju target yang diinginkan, tetapi sistem secara otomatis memilih langkah dengan bobot terkecil, dalam hal ini bobotnya berupa *guild contribution* dan *gold medal* terkecil, serta *additional skill tambahan terbanyak* menuju titik yang diharapkan. Berikut adalah penerapan dari algoritma prim

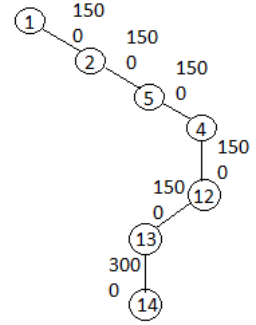
untuk menentukan rute di atas:



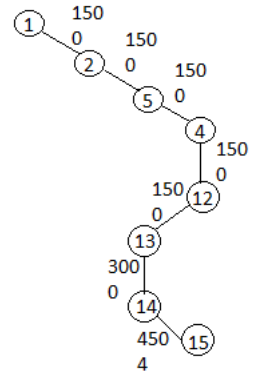
Gambar 3.3.4 graf dari *rune*
 Penerapan dari algoritma prim untuk graf di atas:
 Langkah Spanning Tree



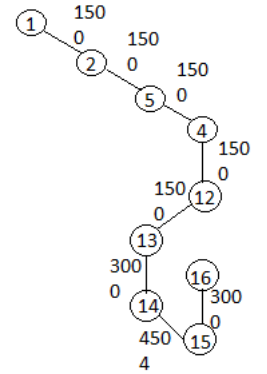
6



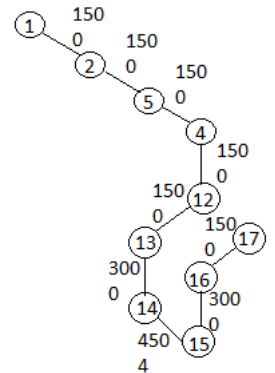
7



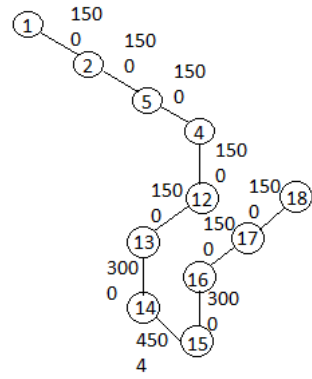
8



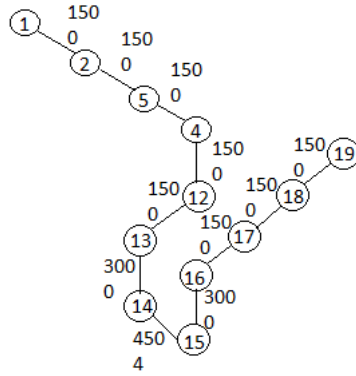
9



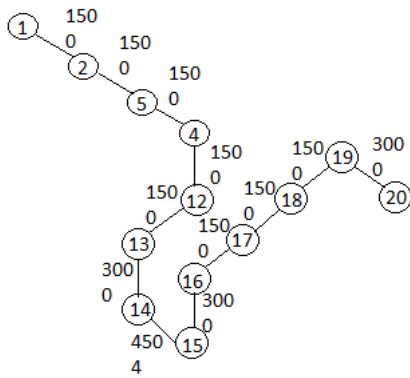
10



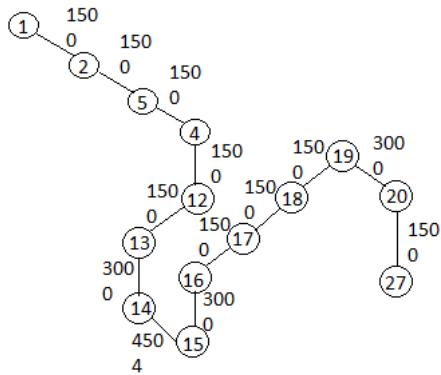
11



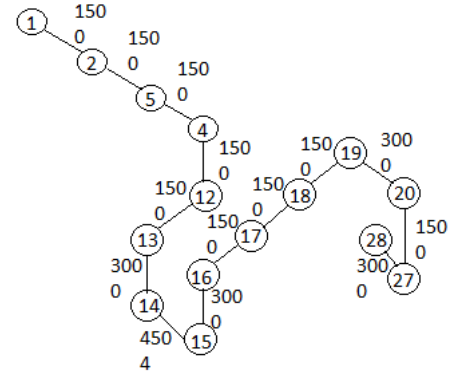
12



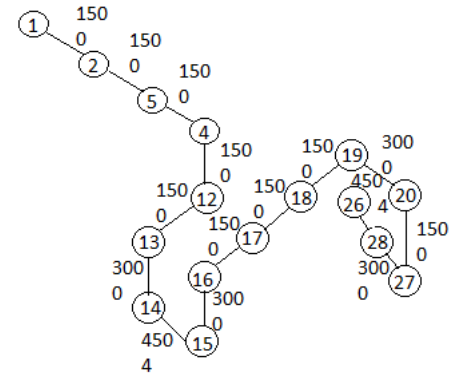
13



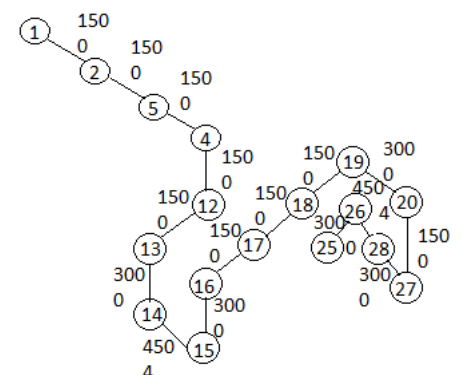
14



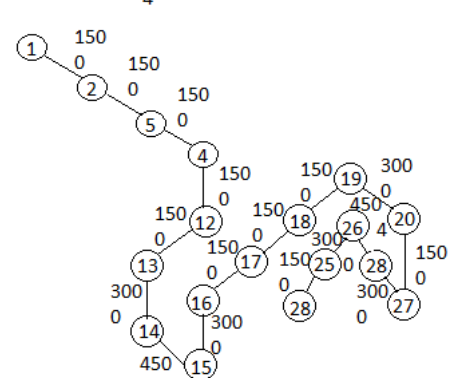
15



16



17



dan monster) dan penggambaran sistem *rune* pada *Aeiss Monument*, teori kombinatorial untuk menentukan kemungkinan lokasi *teleport player* dan peluang seorang player untuk menemukan lokasi MVP dan *miniboss* pada suatu *map* dengan menggunakan *item fly wing*, dan teori *tree* untuk menentukan urutan resep, menentukan jalur terpendek untuk menyerang saat *auto skill* diaktifkan dan jalur terpendek dari titik asal menuju target dengan jalur terpendek.

REFERENSI

- [1] Munir, Rinaldi, Matematika Diskrit, Ed.3, Bandung: Informatika Bandung, 2007
- [2] <https://www.igi-global.com/dictionary/online-games/20978> (Diakses terakhir pada tanggal 9 Desember 2018)
- [3] https://medium.com/@datapath_io/the-history-of-online-gaming-2e70d51ab437 (Diakses terakhir pada tanggal 9 Desember 2018)
- [4] <https://globenewswire.com/news-release/2018/11/06/1645929/0/en/Ragnarok-M-Eternal-Love-marks-No-1-on-Google-Play-in-Three-Countries-followed-by-Apple-Store.html> (Diakses terakhir pada tanggal 9 Desember 2018)
- [5] <https://www.gamerbraves.com/ragnarok-m-eternal-love-will-launch-at-the-end-of-october/> (Diakses terakhir pada tanggal 9 Desember 2018)
- [6] <http://mathworld.wolfram.com/SimpleGraph.html> (Diakses terakhir pada tanggal 9 Desember 2018)
- [7] <https://www.mathworks.com/help/matlab/math/directed-and-undirected-graphs.html> (Diakses terakhir pada tanggal 9 Desember 2018)
- [8] <http://www.mathcs.emory.edu/~cheung/Courses/171/Syllabus/11-Graph/weighted.html> (Diakses terakhir pada tanggal 9 Desember 2018)
- [9] <https://math.tutorvista.com/discrete-math/graph-theory.html> (Diakses terakhir pada 9 Desember 2018)
- [10] <https://www.scribd.com/doc/78500590/Peluang-Adalah-Harapan-Terjadinya-Suatu-Kejadian-Yang-Dikuantitatifkan> (Diakses terakhir pada tanggal 9 Desember 2018)
- [11] <https://duniagames.co.id/news/10218-tak-perlu-pusing-inilah-rekomendasi-adventure-skill-ragnarok-eternal-love-yang-cocok-kamu-miliki> (Diakses terakhir pada tanggal 9 Desember 2018)
- [12] <http://www.romwiki.net/items/50/fly-wing> (Diakses terakhir pada tanggal 9 Desember 2018)
- [13] <http://www.romwiki.net/maps/41/geffen> (Diakses terakhir pada tanggal 9 Desember 2018)
- [14] <https://ragnamobileguide.com/guide-to-rune-system-ragnarok-mobile-eternal-love/> (Diakses terakhir pada tanggal 9 Desember 2018)
- [15] <https://gamingph.com/2018/11/list-of-all-recipes-in-ragnarok-m-eternal-love/> (Diakses terakhir pada tanggal 9 Desember 2018)

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 9 Desember 2018



Winston Wijaya
13517018