

# Aplikasi Greedy dan Graf untuk Memaksimalkan Gold Mobile Legends

Mutawally Nawwar - 13521065  
Program Studi Teknik Informatika  
Sekolah Teknik Elektro dan Informatika  
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia  
13521065@std.stei.itb.ac.id

**Abstract**— Mobile Legends merupakan salah satu gim yang sangat populer di kalangan anak muda. Dalam permainannya, gim ini dibagi menjadi dua tim yang saling memperebutkan kemenangan. Setiap tim terdiri dari lima orang. Gim ini sangat menantang Kerjasama tim serta penguasaan sumber daya yang ada di peta. Dalam peta terdapat sumber daya *jungle* dan *minions*. Sumber daya ini dapat dimanfaatkan untuk mendapatkan *gold*. Jika sebuah tim dapat menguasai sumber daya serta mendapat *gold* yang lebih banyak, maka tim tersebut berkemungkinan besar menang.

**Keywords**—About four key words or phrases in alphabetical order, separated by commas.

## I. PENDAHULUAN

Mobile Legends adalah gim MOBA atau *multiplayer online battle arena*. Hal ini membuat gim ini cukup kompetitif. Gim ini dibagi menjadi 2 tim yang saling berusaha menghancurkan markas dari tim lawan. Untuk menghancurkannya tentu tidak mudah. Sebuah tim tidak bisa langsung menuju markas lawan dan menghancurkannya karena setiap *hero* belum memiliki kekuatan yang cukup serta masih ada pertahanan lain untuk menuju markas lawan.

Hero adalah karakter dalam gim yang digunakan para pemain. Setiap pemain mengontrol satu hero. Hero sendiri memiliki 5 tipe, yaitu *roamer*, *jungler*, *midlaner*, *explainer*, dan *goldlaner*. *Roamer* bertugas dalam menjaga rekan tim karena sebagian besar memiliki badan yang kuat. *Jungler* bertugas untuk menjadi penopang tim karena dia lah yang memanfaatkan sumber daya *jungle*. *Midlaner* bertugas untuk menguasai daerah tengah dan berotasi ke daerah lain untuk membantu rekan tim. *Explainer* bertugas menguasai daerah exp dan meningkatkan level secara cepat. *Goldlaner* bertugas untuk menguasai daerah gold serta sebagai perusak ketika sudah memasuki waktu akhir.

Setiap tipe hero memiliki tugasnya masing-masing, tetapi tujuannya sama yaitu memperkaya tim untuk bisa menghancurkan markas lawan. Dalam makalah ini akan dijelaskan penggunaan prinsip *greedy* dan graf untuk memaksimalkan *gold* yang didapat.

## II. TEORI DASAR

### A. Mobile Legends

Mobile Legends telah dirilis sejak 2016. Gim ini

memiliki beberapa komponen utama.

#### 1. Peta

Berikut adalah gambar peta gim Mobile Legends



Gambar 1 Peta Mobile Legend

(sumber: <https://www.adatah.com/lifestyle/pr-2445546399/yuk-kenali-map-dan-lane-di-game-mobile-legends-berikut-penjelasan-nya>)

Dalam peta Mobile Legends, terdapat beberapa hal:

#### a. Tiga jalur utama

Tiga jalur utama dalam Mobile Legends yaitu, jalur *gold*, jalur *mid*, dan jalur *exp*. Disana lah *minions* akan berjalan dan bertemu dengan *minions* lawan pada daerah setengah jalur menuju markas lawan.

#### b. Daerah hutan

Daerah hutan merupakan tempat monster-monster ada, disana lah *jungler* monster untuk meningkatkan *gold* mereka.

#### c. Turet

Turet merupakan pertahanan setiap tim. Turet ini dapat menembak lawan Ketika memasuki daerah serangnya. Setiap tim memiliki 9 turet. Untuk menghancurkan markas lawan, setiap tim harus menghancurkan turret lawan terlebih dahulu.

#### 2. Detail Hutan

Hutan memiliki beberapa macam monster, setiap monster memiliki waktu Kembali yang beragam

setelah diburu oleh hero. Setiap perburuan membutuhkan waktu 10 detik Berikut adalah detailnya.



Gambar 2 Peta Jungle Mobile Legends (sumber:j/)

1. Muncul pada menit 2, refresh setiap 2 menit, dan berhenti pada menit 8
  2. Muncul pada menit 8, refresh setiap 3 menit
  3. Muncul pada detik 20, refresh setiap 90 detik
  4. Muncul pada detik 39, refresh setiap 70 detik
  5. Muncul pada menit 2, refresh setiap 2 menit
3. Detail Minions  
Minions muncul pada detik 10 dan akan terus muncul setiap 30 detik

## B. Graf

Graf merupakan suatu representasi dari objek-objek diskrit dan hubungan antara objek-objek tersebut. Teori graf bermula saat seorang ahli matematika yang berasal dari Swiss yaitu Leonhard Euler memecahkan masalah jembatan Königsberg. Suatu graf  $G$  terdiri dari himpunan tidak kosong dari simpul  $V$  (vertex) dan himpunan sisi  $E$  (edges) yang menghubungkan sepasang simpul atau bisa didefinisikan sebagai :

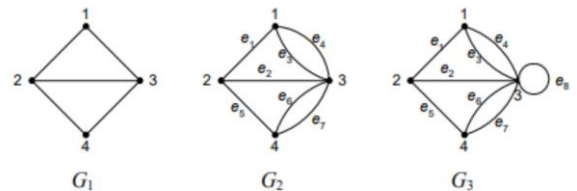
$$G = (V, E)$$

$$V = \{ V_1, V_2, V_3, \dots, V_n \}$$

$$E = \{ E_1, E_2, E_3, \dots, E_n \}$$

Jenis-jenis graf dapat dibedakan berdasarkan keberadaan gelang atau sisi ganda dan berdasarkan orientasi arah pada sisi graf. Berdasarkan keberadaan gelang atau sisi ganda, graf dapat dibedakan menjadi dua jenis yaitu :

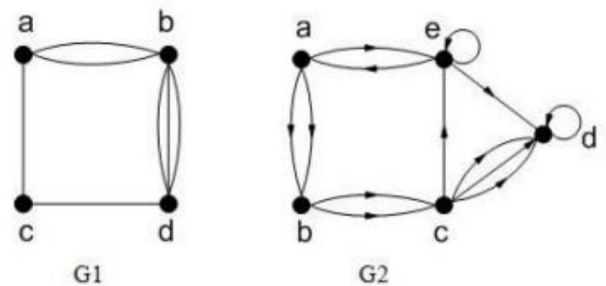
1. Graf Sederhana Graf sederhana merupakan yang tidak memiliki gelang ataupun sisi ganda
2. Graf Tak-Sederhana Graf tak-sederhana merupakan graf yang memiliki sisi ganda atau gelang.



Gambar 3 Graf Sederhana dan Tak-Sederhana (sumber: arsip penulis)

Berdasarkan orientasi arah pada sisi graf, graf dapat dibedakan menjadi dua jenis yaitu :

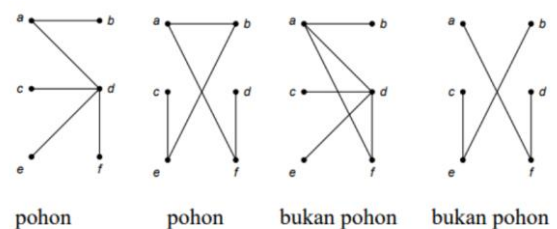
1. Graf Tak-Berarah Graf tak-berarah merupakan graf yang tidak mempunyai orientasi arah pada sisinya.
2. Graf Berarah Graf berarah merupakan graf yang mempunyai orientasi arah pada sisinya



Gambar 4 Graf Berarah dan tak-Berarah (sumber: arsip penulis)

## C. Pohon

Pohon atau bisa disebut dengan *tree* merupakan graf tak berarah terhubung yang tidak memiliki sirkuit. Sirkuit merupakan suatu lintasan yang berawal dan berakhir pada simpul yang sama.



Gambar 5 Pohon (sumber: arsip penulis)

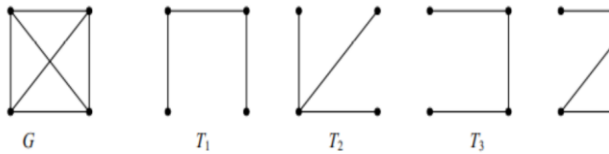
Pohon memiliki sifat-sifat (properti) yaitu misalkan  $G = (V, E)$  adalah graf tak-berarah sederhana dan jumlah simpulnya  $n$ . Sehingga pernyataan-pernyataan yang terdapat di bawah ini yaitu ekuivalen :

1.  $G$  adalah pohon
2. Setiap pasang simpul di dalam  $G$  terhubung dengan lintasan tunggal.
3.  $G$  terhubung dan memiliki  $m = n - 1$  buah sisi.
4.  $G$  tidak mengandung sirkuit dan memiliki  $m = n - 1$  buah sisi.

- 5. G tidak mengandung sirkuit dan penambahan suatu sisi pada graf akan membuat hanya satu sirkuit.
- 6. G terhubung dan semua sisinya adalah jembatan.

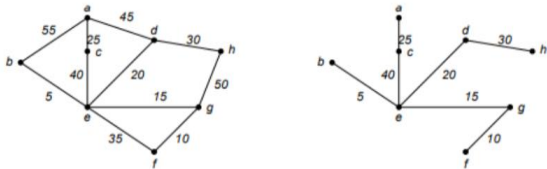
**D. Pohon Merentang**

Pohon merentang (spanning tree) merupakan upagraf merentang (upagraf yang mengandung semua simpul graf yang direntangnya) yang merupakan pohon. Pohon merentang didapatkan dengan memotong sirkuit di dalam graf.



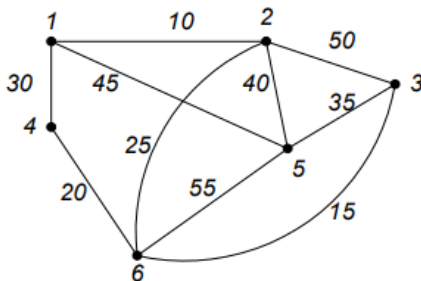
Gambar 6 Pohon Merentang (sumber: arsip penulis)

Pohon merentang minimum merupakan graf terhubung berbobot yang termasuk ke dalam pohon merentang dan memiliki bobot paling minimum.



Gambar 7 Pohon Merentang Minimum (sumber: arsip penulis)

Salah satu algoritma untuk menentukan pohon merentang minimum adalah algoritma Kruskal. berikut adalah gambaran algoritma Kruskal.



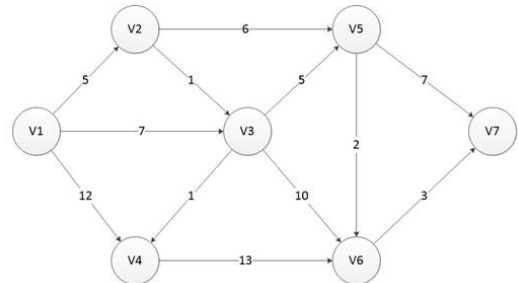
Gambar 8 Contoh Graf (sumber: arsip penulis)

Langkah	Sisi	Bobot	Hutan merentang
0			1 2 3 4 5 6
1	(1, 2)	10	1-2
2	(3, 6)	15	1-2, 3-6
3	(4, 6)	20	1-2, 3-6, 4-6
4	(2, 6)	25	1-2, 3-6, 4-6, 2-6
5	(1, 4)	30	ditolak
6	(3, 5)	35	1-2, 3-6, 4-6, 2-6, 3-5

Gambar 9 Langkah Kruskal (sumber: arsip penulis)

**E. Algoritma Dijkstra**

Algoritma djikstra digunakan untuk menentukan rute terpendek antar dua node dalam sebuah graf. Berikut adalah contohnya.



Gambar 10 Contoh Graf (sumber: <https://mti.binus.ac.id/2017/11/28/algoritma-dijkstra/>)

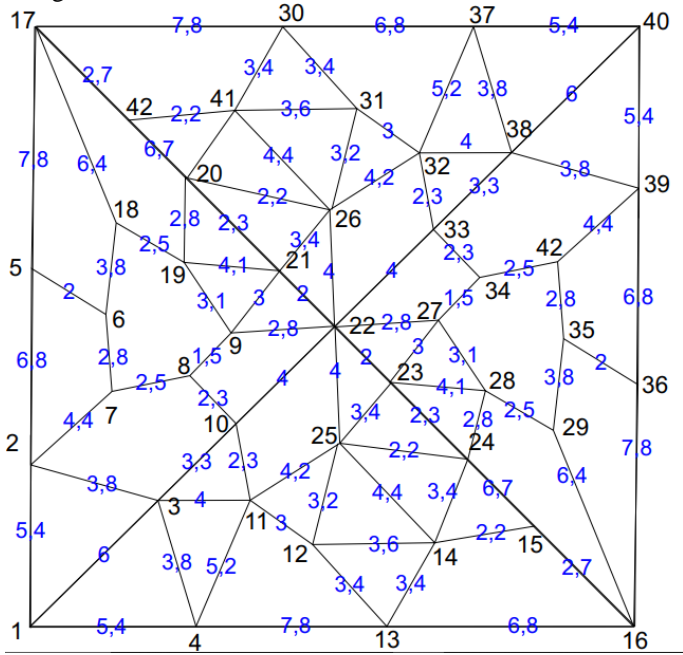
Iteration	Unvisited (Q)	Visited (S)	Current	Node : Min = (dist[node], prev[node])iteration						
				V1	V2	V3	V4	V5	V6	V7
	Initialization (V1, V2, V3, V4, V5, V6, V7)	{-}		(0, -0)	(∞, -0)	(∞, -0)	(∞, -0)	(∞, -0)	(∞, -0)	(∞, -0)
1	{V2, V3, V4, V5, V6, V7}	{V1}	V1		(5, V1)1	(7, V1)1	(12, V1)1	(∞, V1)1	(∞, V1)1	(∞, V1)1
2	{V3, V4, V5, V6, V7}	{V1, V2}	V2			(6, V2)2	(12, V1)1	(11, V2)2	(∞, V2)2	(∞, V2)2
3	{V4, V5, V6, V7}	{V1, V2, V3}	V3				(7, V3)3	(11, V3)3	(16, V3)3	(∞, V3)3
4	{V5, V6, V7}	{V1, V2, V3, V4}	V4					(11, V3)3	(16, V3)3	(∞, V3)3
5	{V6, V7}	{V1, V2, V3, V4, V5}	V5						(13, V5)5	(18, V5)5
6	{V7}	{V1, V2, V3, V4, V5, V6}	V6							(16, V6)6

Gambar 11 Tabel Hasil Algoritma Dijkstra (sumber: <https://mti.binus.ac.id/2017/11/28/algoritma-dijkstra/>)

Dari table tersebut dapat disimpulkan jarak terpendek V1 menuju V7 adalah V1-V2-V3-V5-V6-V7.

### III. PEMBAHASAN

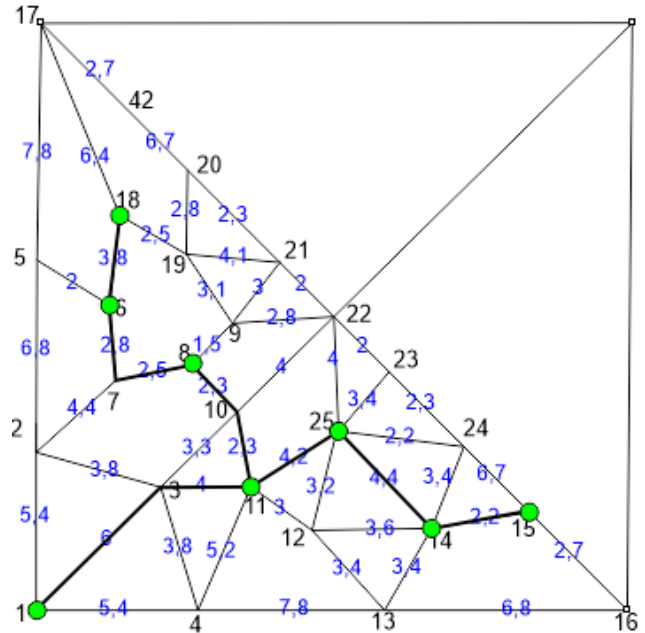
Dari Bab II dapat diketahui bahwa Mobile Legends memiliki peta sebagai tempat permainan. Peta tersebut dapat diubah menjadi graf dengan setiap *edge* memiliki bobot waktu tempuh sebagai berikut.



Gambar 12 Konversi Graf dari Peta Mobile Legends (sumber: arsip penulis)

Untuk menentukan jalan yang mendapatkan *gold* paling banyak dan waktu paling singkat, akan digunakan pohon merentang minimum menggunakan algoritma Kruskal dengan menganggap simpul selain monster sebagai sambungan sisi.

Didapat pohon merentang minimum sebagai berikut.

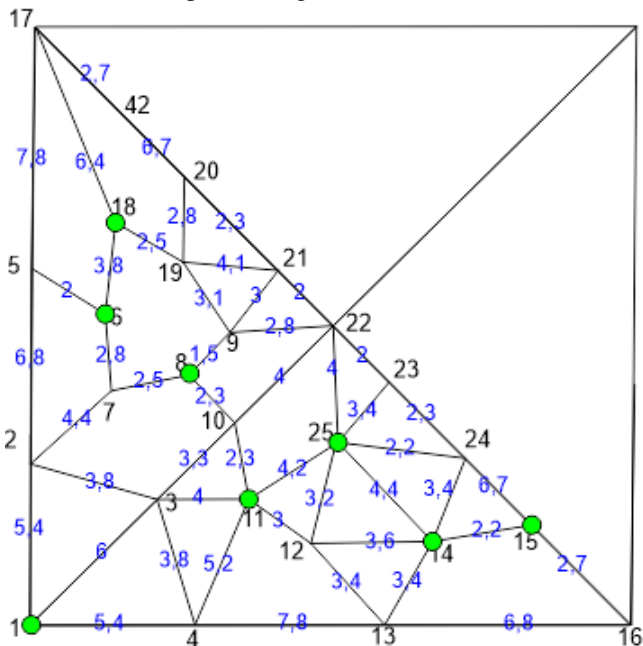


Gambar 14 Hasil Pohon Merentang Minimum (sumber: arsip penulis)

#### A. Optimalisasi gold jungler dengan pohon merentang minimum

##### 1. Kondisi normal

*Jungler* bertugas memburu monster di hutan, berikut adalah peta letak monster hutan dan simpul 1 sebagai daerah mulai.



Gambar 13 Peta Jungle (sumber: arsip penulis)

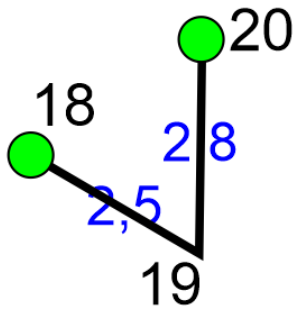
Dari pohon merentang minimum tersebut, dapat dipilih dengan *greedy* simpul mana yang akan dilalui terlebih dahulu. Didapat urutan simpul sebagai berikut.

1-11-25-14-15-14-25-11-8-6-18

Sehingga *jungler* memiliki waktu berburu 52,5 detik di perjalanan ditambah dengan waktu berburu yaitu  $10 \times 7 = 70$  detik. Oleh karena itu, waktu totalnya adalah 122,5 detik.

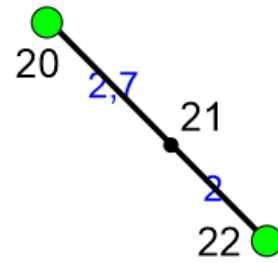
##### 2. Kondisi ada *turtle* atau *lord*

Dari poin pertama didapat bahwa *jungler* membutuhkan waktu 122,5 detik untuk memburu semua monster hutan. Selanjutnya akan muncul monster hutan yang lebih kuat yaitu *turtle*. Waktu berburu dari poin 1 bertepatan dengan munculnya *turtle* ini, sehingga sangat optimal jika *jungler* langsung berburu disana. Di peta, *turtle* akan muncul pada simpul nomor 20. Titik akhir perburuan *jungler* ada di simpul 18. Dengan menggunakan algoritma *dijkstra* didapat rute terpendek sebagai berikut.



Gambar 15 Rute Terpendek Jungler dengan Turtle (sumber: arsip penulis)

Sehingga jarak terpendeknya 5,3 detik.



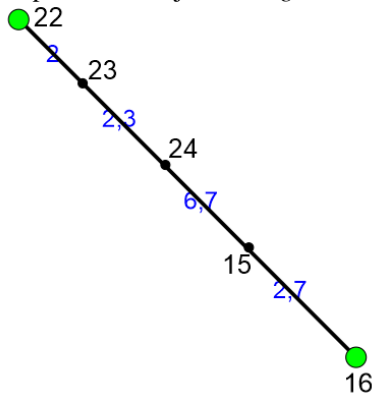
Gambar 17 Rute Terpendek Midlaner dengan Turtle (sumber: arsip penulis)

Dari hasil tersebut butuh waktu 4,7 detik untuk bolak-balik menuju simpul 20. Oleh karena itu, *midlaner* memiliki waktu 25,3 detik untuk membantu berburu *turtle* dan *lord*.

## B. Optimalisasi gold midlaner dengan jarak terpendek

### 1. Kondisi normal

Tugas utama seorang *midlaner* adalah menjaga daerah tengah (simpul 22). *midlaner* akan ebrburu *minions* lawan di sana. Setiap 30 detik ada *minions* yang harus diburu. Oleh karena itu ada jeda 30 detik untuk seorang *midlaner* berotasi ke daerah *gold* (simpul 16) atau *exp* (simpul 17). kedua daerah tersebut memiliki jarak yang sama karena peta Mobile Legends simetris. Dengan menggunakan djikstra berikut adalah jarak terpendek menuju daerah *gold* atau *exp*.



Gambar 16 Rute Terpendek Midlaner Berotasi (sumber: arsip penulis)

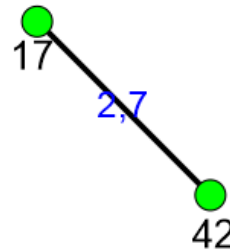
Sehingga waktu untuk bolak-balik rotasi adalah 13,7 detik. Sehingga masih ada waktu 26,3 detik untuk membantu tim berburu *hero* lawan atau pun menghancurkan turret lawan tanpa harus kehilangan *gold* di daerah *mid*.

### 2. Kondisi ada *turtle* atau *lord*

Pada suatu saat akan muncul *turtle* dan *lord* pada simpul 20. *Midlaner* sebisn mungkin harus membantu dalam memburu *turtle* dan *lord*. Dengan menggunakan djikstra, didapat jarak terpendek sebagai berikut.

## C. Optimalisasi gold explaner dengan jarak terpendek

*Explaner* berada pada simpul 17. Pada dasarnya bertugas menjaga daerah tersebut dan berburu *gold* di sana dari *minions*. *Minions* akan datang setiap 30 detik sekali, sehingga dalam jeda waktu *explaner* dapat berburu monster hutan yang belum diburu *jungler* ataupun monster di daerah musuh. Dengan *greedy* didapat monster terdekat sebagai berikut.

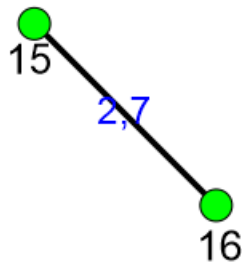


Gambar 18 Rute Terpendek Explaner dengan Monster (sumber: arsip penulis)

Dengan berburu monster tersebut *explaner* akan memerlukan waktu 5,4+10 untuk memburu monster tersebut dan masih mendapat *gold* dari *minions*.

## D. Optimalisasi gold goldlaner dengan jarak terpendek

*Goldlaner* pada simpul 16. Pada dasarnya bertugas untuk menjaga daerah tersebut dan berburu *gold* di sana dari *minions*. *Minions* akan datang setiap 30 detik sekali, sehingga dalam jeda waktu tersebut *goldlaner* dapat berburu monster hutan yang belum diburu *jungler* ataupun monster di daerah lawan. Dengan *greedy* monster terdekat sebagai berikut.

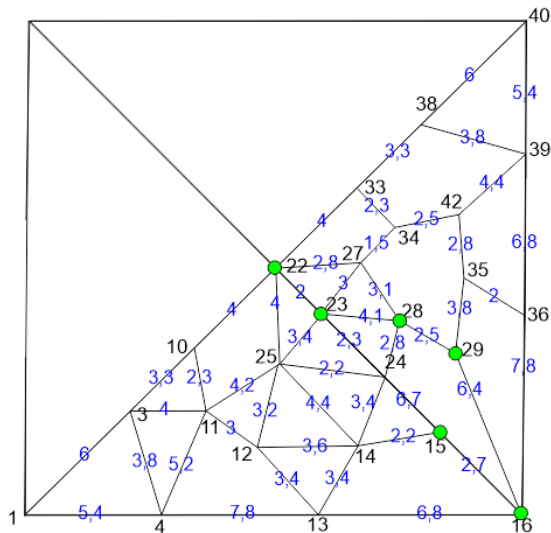


Gambar 19 Rute Terpendek Goldlaner dengan Monster  
(sumber: arsip penulis)

Dengan berburu monster tersebut *goldlaner* akan memerlukan waktu  $5,4+10$  untuk memburu monster tersebut dan masih mendapat *gold* dari *minions*.

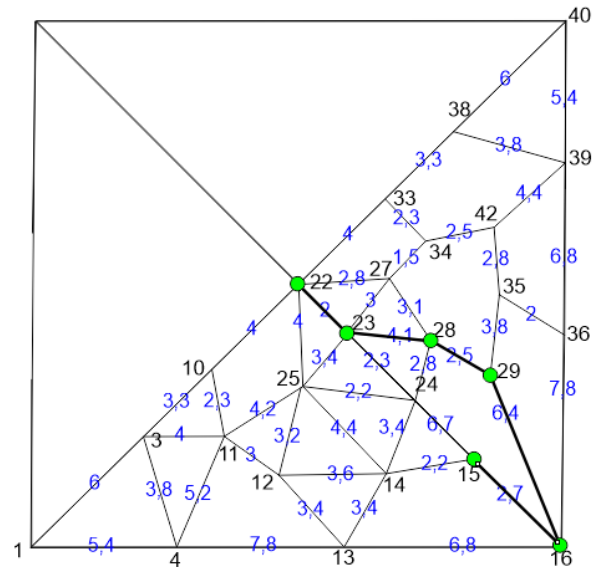
### E. Optimalisasi roamer dengan dengan pohon merentang minimum

*Roamer* sendiri memiliki tugas untuk menjaga timnya, sehingga *roamer* optimal jika dapat berotasi dengan cepat dan jangkauan peta yang luas. Dengan begitu informasi *hero* lawan akan diketahui oleh rekan tim. Berikut adalah simpul-simpul kunci yang biasanya terdapat lawan bersembunyi dari daerah *gold* sampai *mid*.



Gambar 20 Simpul Kunci Roamer  
(sumber: arsip penulis)

Dengan algoritma Kruskal didapat pohon merentang minimal sebagai berikut.



Gambar 21 Pohon Merentang Minimum Roamer  
(sumber: arsip penulis)

Dari hasil diatas, *roamer* dapat membuka peta setiap 17,7 detik.

## IV. KESIMPULAN

Graf sangat banyak pengaplikasiannya dalam berbagai bidang. Salah satunya dalam gim *Mobile Legends*. Dalam *Mobile Legends* tim harus mengumpulkan sebanyak-banyaknya *gold*. Dengan menerapkan graf sebuah tim dapat memaksimalkan pendapatan *gold* nya. Berikut adalah ringkasannya.

1. *Jungler*  
*Jungler* dapat mengikuti pohon merentang minimum pada gambar
2. *Midlaner*  
*Midlaner* dapat memanfaatkan jeda 30 detik gelombang *minions* dengan berotasi ke *gold* atau *exp*, berdasarkan gambar.
3. *Goldlaner* dan *Explaner*  
Kedua tipe ini dapat berburu monster hutan dalam jeda gelombang *minions*.
4. *Roamer*  
*Roamer* dapat berotasi untuk memantau peta sehinggarekan tim dapat menambah *gold* dengan naman berdasarkan gambar.

## V. PENUTUP

Puji syukur kepada Tuhan Yang Maha Esa atas berkat dan rahmat-Nya sehingga penyusunan makalah ini dapat selesai dengan tepat waktu. Ucapan terima kasih juga penulis sampaikan kepada keluarga, teman-teman, dan semua pihak yang telah mendukung keberjalanan studi pada mata kuliah Matematika Diskrit selama satu semester ini. Yang utama, penulis juga menyampaikan terima kasih kepada para dosen mata kuliah IF2120 Matematika Diskrit terutama kepada Ibu Ulfa Maulidevi selaku pengajar K01 atas ilmu yang telah ibu

disampaikan.

#### REFERENCES

- [1] <https://informatika.stei.itb.ac.id/~rinaldi.munir>  
diakses pada 10 Desember 2022
- [2] [https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2021-2022/Makalah2021/Makalah-Matdis-2021%20\(152\).pdf](https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2021-2022/Makalah2021/Makalah-Matdis-2021%20(152).pdf)  
diakses pada 10 Desember 2022
- [3] “Graph”, “Tree” (2012) in *Discrete mathematics and its applications*.  
New York: McGraw-Hill, pp. 641–803.
- [4] <https://www.adatah.com/lifestyle/pr-2445546399/yuk-kenali-map-dan-lane-di-game-mobile-legends-berikut-penjelasan-penjelasan>  
diakses pada 10 Desember 2022
- [5] <https://www.tembolok.id/game-mekanik-dan-map-jungle-mobile-legend-arena-valor>  
diakses pada 10 Desember 2022

#### 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, 11 Desember 2022



Mutawally Nawwar, 13521065