

Pemodelan Pohon Keputusan Penentuan Akhir Cerita *Undertale*

Feralezer L. G. Tampubolon - 13519062¹

Program Studi Teknik Informatika

Sekolah Teknik Elektro dan Informatika

Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia

¹13519062@std.stei.itb.ac.id

Undertale adalah sebuah video game berbasis PC yang dirilis pada tanggal 15 September 2015. *Undertale* memiliki beberapa kemungkinan akhir cerita. Pohon keputusan dapat dipakai untuk memodelkan penentuan akhir cerita video game ini.

Keywords—Matematika diskrit, Pohon keputusan, Video game, *Undertale*.

I. PENDAHULUAN

Video game adalah permainan yang dijalankan melalui alat-alat elektronik. Manusia yang bertindak sebagai pemain dapat mengirimkan masukan kepada alat elektronik. *Video game* akan menganggap masukan tersebut sebagai perintah terhadap permainan yang sedang berlangsung dan menjalankannya. *Video game* pertama dibuat pada tahun 1947 dengan sebutan “alat hiburan berbasis tabung sinar katoda”, sangat sederhana jika dibandingkan dengan standar *video game* sekarang ini.

Undertale adalah sebuah *video game* berbasis PC yang dirilis pada tanggal 15 September 2015 [1]. *Video game* ber-genre *role playing game (RPG)* ini dibuat sendiri oleh seseorang bernama Toby Fox, yang menginginkan suatu *RPG* di mana pemainnya “tidak perlu melakukan kekerasan”. Pasca rilis, *Undertale* menjadi sangat populer, dan dengan cepat menjadi “game wajib” bagi para *gamer* (pemain *video game*). *Undertale* mempopulerkan kembali industri *game* independen, mendapat berbagai nominasi penghargaan, dan hingga saat ini tetap dianggap sebagai salah satu *video game* terbaik yang pernah ada oleh sejumlah kritikus.



Gambar 1. Logo *Undertale* (sumber: undertale.com)

Seperti kebanyakan *RPG*, *Undertale* memiliki alur cerita utama yang harus diikuti. Akan tetapi, alur cerita tersebut tidak memiliki akhir/*ending* yang linier. Rangkaian aksi tertentu yang pemain lakukan sepanjang kelangsungan *game* akan menentukan *ending* mana yang akan diperoleh pemain. Mekanisme penentuan akhir cerita ini dapat dimodelkan menggunakan pohon keputusan/*decision tree*.

II. PRINSIP DASAR MATEMATIKA DISKRIT YANG DIGUNAKAN

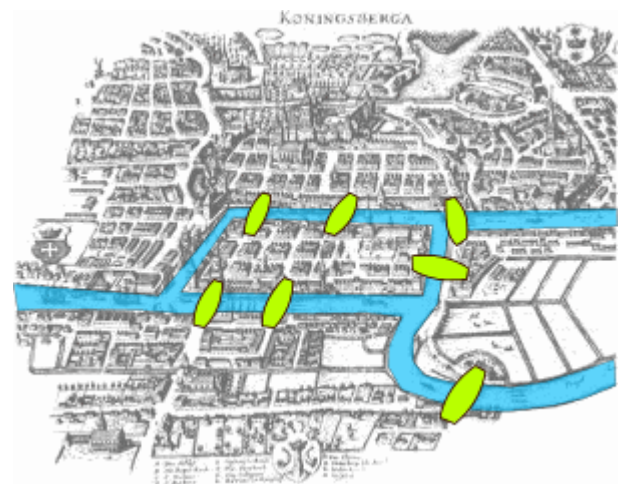
A. Graf

Beberapa konsep dasar graf diperlukan dalam pemahaman mengenai pohon keputusan. Konsep-konsep dasar tersebut akan dijabarkan secara singkat terlebih dahulu.

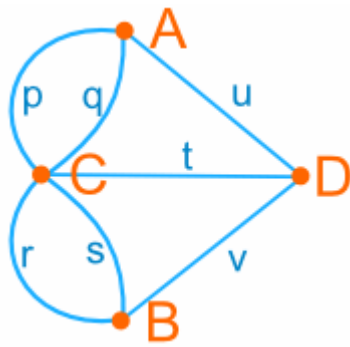
1. Definisi dan Sejarah Graf

Graf didefinisikan sebagai suatu tupel yang terdiri atas 2 anggota (V, E), di mana V adalah himpunan simpul/*vertice* pada graf, dan E himpunan sisi/*edge* yang menunjukkan relasi antar simpul pada graf. Secara visual, graf dapat direpresentasikan sebagai sekumpulan titik (simpul) yang saling dihubungkan oleh sejumlah garis (sisi).

Ide mengenai graf pertama kali timbul ketika Leonhard Euler, seorang matematikawan asal Swiss, ingin memecahkan apa yang saat ini dikenal orang sebagai “Persoalan Jembatan Königsberg”. Pada suatu kota yang bernama Königsberg, terdapat 7 buah jembatan yang menghubungkan 4 bagian berbeda dari kota tersebut. Pertanyaannya, dapatkah seseorang memulai perjalanan dari satu bagian kota, melewati 7 jembatan tersebut tepat sekali saja, lalu kembali ke tempat ia memulai perjalanannya? Euler menyimpulkan hal tersebut tidak mungkin dilakukan dengan memodelkan “Persoalan Jembatan Königsberg” sebagai suatu graf.



Gambar 2. Persoalan Jembatan Königsberg (sumber: Public Domain)

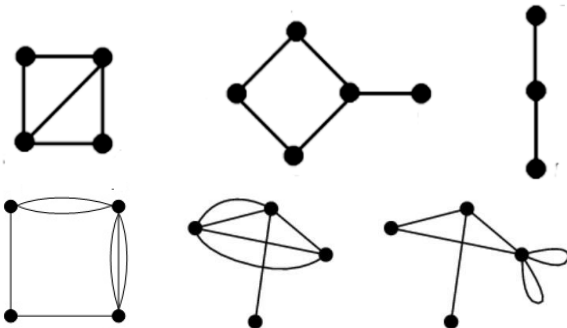


Gambar 3. Visualisasi graf “Persoalan Jembatan Königsberg” dengan 4 simpul {A, B, C, D} dan 7 sisi {p, q, r, s, t, u, v} (sumber: mathisfun.com)

2. Jenis-jenis Graf

Berdasarkan banyaknya sisi yang menghubungkan 2 simpul tertentu, serta ada/tidaknya sisi yang menghubungkan suatu simpul dengan simpul itu sendiri (sisi yang seperti ini disebut sebagai gelang/loop) graf dibedakan menjadi 2 jenis [2]:

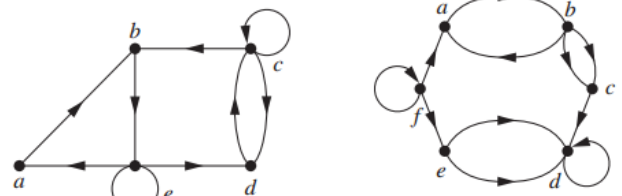
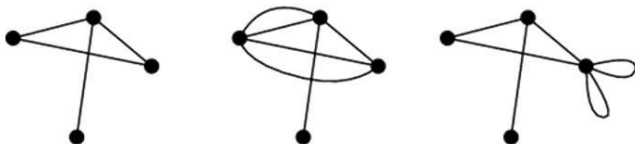
1. Graf sederhana (*simple graph*)
Graf sederhana adalah graf yang tidak memiliki gelang maupun simpul yang dihubungkan oleh 2 atau lebih sisi.
2. Graf tak sederhana (*un-simple graph*)
Graf tak sederhana adalah graf yang memiliki gelang ataupun simpul yang dihubungkan oleh 2 atau lebih sisi.



Gambar 4. Contoh graf sederhana (atas) dan tak sederhana (bawah) (sumber: informatika.stei.itb.ac.id/~rinaldi.munir)

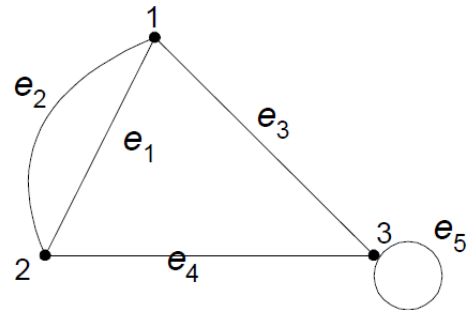
Berdasarkan orientasi arah pada sisi, graf dibedakan menjadi 2 jenis [2]:

1. Graf tak berarah (*undirected graph*)
Graf tak berarah adalah graf yang sisinya tidak mempunyai orientasi arah (dinyatakan dalam bentuk panah).
2. Graf berarah (*directed graph* atau *digraph*)
Graf berarah adalah graf yang setiap sisinya diberikan orientasi arah (dinyatakan dalam bentuk panah).



Gambar 5. Contoh graf tak berarah (kolom kiri-bawah) dan berarah (atas) (sumber: informatika.stei.itb.ac.id/~rinaldi.munir)

3. Terminologi Graf



Gambar 6. Graf dengan 3 simpul {1, 2, 3} dan 5 sisi {e1, e2, e3, e4, e5} (sumber: informatika.stei.itb.ac.id/~rinaldi.munir)

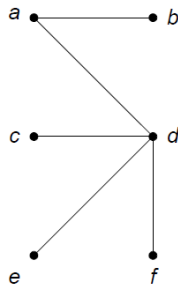
Beberapa terminologi graf [2]:

1. Derajat (*degree*)
Derajat suatu simpul adalah banyaknya sisi yang berikatan dengan simpul tersebut.
Pada graf di gambar 6, simpul 1 memiliki derajat 3, simpul 2 memiliki derajat 3, dan simpul 3 memiliki derajat 2.
2. Lintasan (*path*)
Suatu lintasan pada graf adalah barisan simpul dan sisi, di mana simpul-simpul v_1, v_2, \dots, v_n dihubungkan oleh sisi-sisi $e_1 = (v_1, v_2), e_2 = (v_2, v_3), \dots, e_{n-1} = (v_{n-1}, v_n)$.
Pada graf di gambar 6, lintasan 1,3,2 adalah lintasan dengan barisan sisi e_3, e_4 .
3. Siklus (*cycle*) atau sirkuit (*circuit*)
Siklus adalah lintasan yang berawal dan berakhir pada simpul yang sama.
Pada graf di gambar 6, lintasan 1,2,3,1 adalah sebuah sirkuit.
4. Keterhubungan (*connected*)
Dua buah simpul v_1 dan v_2 pada graf disebut terhubung apabila terdapat lintasan dari v_1 ke v_2 .
Pada graf di gambar 6, (misalnya) simpul 1 terhubung dengan simpul 2 dan 3.
5. Upagraf (*subgraph*) dan komplement upagraf
Suatu graf $G_1 = (V_1, E_1)$ adalah upagraf dari graf $G = (V, E)$ jika $V_1 \subseteq V$ dan $E_1 \subseteq E$. Sementara itu, graf $G_2 = (V_2, E_2)$ adalah komplement dari upagraf $G_1 = (V_1, E_1)$ jika $E_2 = E - E_1$ dan V_2 adalah himpunan simpul yang dihubungkan oleh sisi-sisi pada E_2 .
Pada graf di gambar 6, $G_1 = (\{1, 2\}, \{e_1, e_2\})$ dan $G_2 = (\{1, 3\}, \{e_3, e_4, e_5\})$ adalah suatu pasangan upagraf dan komplementnya.

B. Pohon

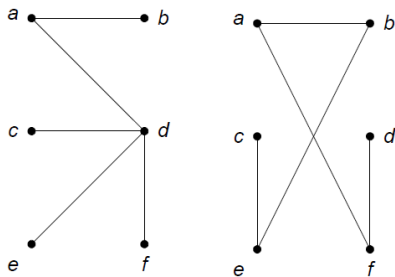
1. Definisi Pohon

Suatu pohon adalah graf tak berarah terhubung sederhana yang tidak mengandung sirkuit [3].



Gambar 7. Pohon (sumber: informatika.stei.itb.ac.id/~rinaldi.munir)

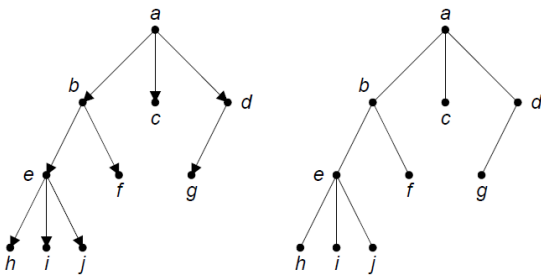
Suatu hutan adalah kumpulan pohon yang saling lepas/tidak terhubung [3].



Gambar 8. Hutan yang terdiri atas 2 pohon (sumber: informatika.stei.itb.ac.id/~rinaldi.munir)

2. Pohon Berakar

Pohon berakar memiliki definisi yang sedikit bertentangan dengan definisi pohon. Pohon berakar adalah pohon yang ditransformasi menjadi graf berarah dengan memilih satu simpul sebagai akar, dan membuat semua sisi memiliki orientasi arah yang menjauhi akar tersebut.



Gambar 9. Pohon berakar. Panah dapat dihilangkan jika akar pohon sudah jelas (sumber: informatika.stei.itb.ac.id/~rinaldi.munir)

Pohon keputusan adalah contoh pohon berakar.

2a. Terminologi Pohon Berakar

Beberapa terminologi pohon berakar [4]:

1. Anak (*child* atau *children*) dan orangtua (*parent*)

Jika ada suatu sisi $e = (v_1, v_2)$ di mana v_1 adalah simpul asal dan v_2 adalah simpul tujuan, maka v_1 adalah orangtua v_2 dan v_2 adalah anak v_1 .

Pada pohon di gambar 9, a adalah orangtua dari b, c, d dan b, c, d adalah anak-anak dari a .

2. Daun (*leaf*) dan simpul dalam (*internal nodes*)

Daun adalah simpul yang tidak memiliki anak. Simpul dalam adalah simpul yang memiliki anak.

Pada pohon di gambar 9, c, f, g, h, i, j adalah daun, b, d, e adalah simpul dalam.

3. Derajat (*degree*)

Derajat suatu simpul adalah banyaknya anak dari simpul tersebut.

Pada pohon di gambar 9, a dan e memiliki derajat 3, b memiliki derajat 2, d memiliki derajat 1, dan c, f, g, h, i, j memiliki derajat 0.

III. SEKILAS MENGENAI *UNDERTALE*

Sebelum mengulas lebih lanjut, penulis ingin memberi peringatan bahwa mulai dari sekarang, isi dari makalah ini akan dipenuhi *spoiler* jalan cerita *Undertale*. Penulis mengharapkan tanggung jawab masing-masing pembaca akan hal ini.

Latar belakang kisah *Undertale* adalah suatu dunia yang dihuni oleh manusia dan monster. Konon, keduanya pernah hidup berdampingan. Namun sekarang, hanya manusia yang hidup di permukaan, sementara monster menutup diri dan berdiam di bawah tanah.

Karakter yang dimainkan oleh pemain adalah seorang manusia yang secara tidak sengaja terperosok ke bawah tanah hingga sampai di dunia monster. Lantas dimulailah perjalanan sang pemain untuk mencari jalan kembali ke dunia luar, dunia manusia. Selama di perjalanan, sedikit demi sedikit terungkaplah cerita di balik dunia monster dan para penghuninya yang entah kenapa tidak semuanya bersikap antagonis terhadap sang pemain.

Seperti *RPG* pada umumnya, *gameplay* dari *Undertale* memiliki 2 *state*: eksplorasi dan pertarungan. Pemain bereksplorasi dengan menggerakkan karakter menggunakan kunci w, a, s, d pada *keyboard*. Pemain dapat berinteraksi dengan objek-objek tertentu yang ia temui, seperti mengambil barang yang tergeletak di tanah atau memulai pembicaraan dengan monster-monster tertentu yang bisa diajak berdialog. Ketika sedang bergerak, pemain dapat secara *random* bertemu dengan musuh (berupa monster), dan *game* akan masuk ke *state* pertarungan.



Gambar 10. Eksplorasi pada *Undertale* (sumber: undertale.com)

giliran musuh, hati merepresentasikan pemain yang harus mengelak dari serangan-serangan musuh (objek-objek bergerak yang berwarna putih di dalam kotak yang disediakan) (sumber: undertale.com)

Jika pemain berhasil mengalahkan monster yang menjadi musuh (membuat HP monster tersebut menjadi 0, bukan melakukan *mercy*), pemain akan mendapat EXP. Ketika pemain berhasil mengumpulkan EXP dalam jumlah tertentu, LV pemain akan naik dan HP/nyawa maksimum pemain akan bertambah tinggi. Jika pertarungan selesai dengan cara lain, pemain tidak akan mendapat EXP.

Jumlah monster yang dapat pemain kalahkan pada tiap-tiap daerah yang bisa dijelajahi memiliki batasan. Jika jumlah monster yang dikalahkan pada suatu daerah sudah mencapai batas, pemain tidak akan lagi menjumpai monster di daerah itu.

Pada akhir tiap daerah, akan ada satu *boss* monster yang harus dilawan pemain. Sama seperti teman-temannya, *boss* monster tidak perlu dikalahkan, namun kondisi yang diperlukan untuk mengaktifkan pilihan *mercy* akan jauh lebih sulit untuk dicapai.

Akhir Cerita

Secara umum ada 3 kemungkinan akhir cerita yang bisa diperoleh pemain:

1. Neutral Route

Pemain akan memperoleh *neutral route* jika pada akhir permainan, ada monster yang tidak dikalahkan oleh pemain. Ingat bahwa pemain tidak diwajibkan untuk mengalahkan musuh; ada cara-cara lain yang bisa ditempuh untuk mengakhiri pertarungan. Akhir cerita *neutral route* adalah pemain berhasil keluar dari dunia monster dan kembali ke dunia manusia.

2. True Pacifist Route

Pemain akan memperoleh *true pacifist route* jika dan hanya jika:

- pemain sama sekali tidak mengalahkan satupun monster (total EXP = 0, LV = 1), dan
 - pemain sudah pernah melihat akhir cerita *neutral route*.
- Jika pemain tidak mengalahkan satupun monster tapi belum pernah melihat akhir cerita *neutral route*, pemain tetap akan mendapat akhir cerita *neutral route*. Namun dengan mudah pemain bisa me-load *savegame* terakhir dan mengulang bagian akhir ceritanya untuk melihat *true pacifist route*. Akhir cerita *true pacifist route* adalah pemain berhasil menyelesaikan konflik yang terjadi antara manusia dan monster, sehingga keduanya kembali hidup berdampingan.

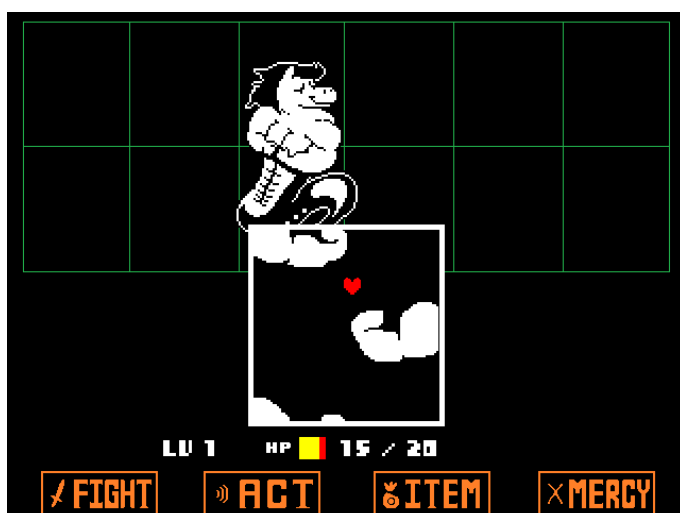
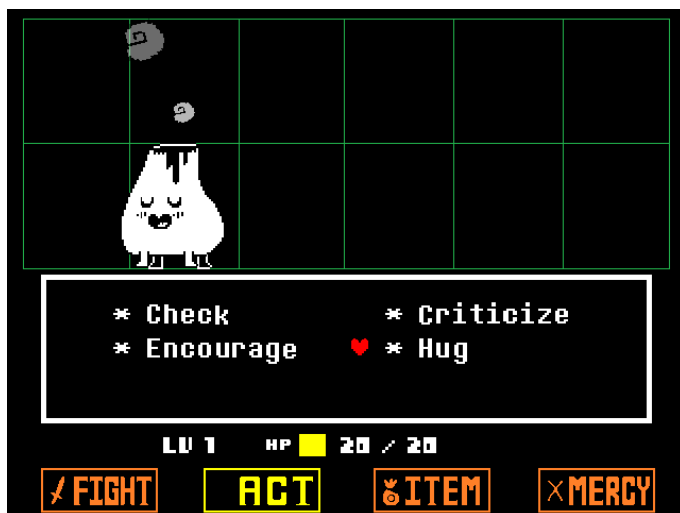
3. Genocide Route

Pemain akan memperoleh *genocide route* ketika pemain berhasil mengalahkan semua monster yang ada di bawah tanah. Ini berarti mencapai batas maksimum monster yang bisa dikalahkan pada semua daerah dan mengalahkan semua boss monster. Akhir cerita *genocide route* adalah pemain merusak dunia *Undertale*. Jika setelahnya pemain ingin memulai permainan baru, pemain akan diberikan pilihan untuk “menciptakan ulang” dunia *Undertale*. Dunia yang diciptakan ulang ini hampir identik dengan dunia yang sebelumnya. Tapi ketika di dunia yang baru ini pemain mencoba untuk mengambil akhir cerita *neutral route* atau *true pacifist route*, akan ada *plot twist* yang sifatnya permanen dan tidak bisa dicegah.

State pertarungan di *Undertale* terdiri atas 2 fase: fase giliran pemain dan fase giliran musuh. Pada fase giliran pemain, akan muncul suatu kotak dialog yang berisi aksi-aksi yang bisa dilakukan pemain. Pada fase ini, pemain dapat memilih untuk:

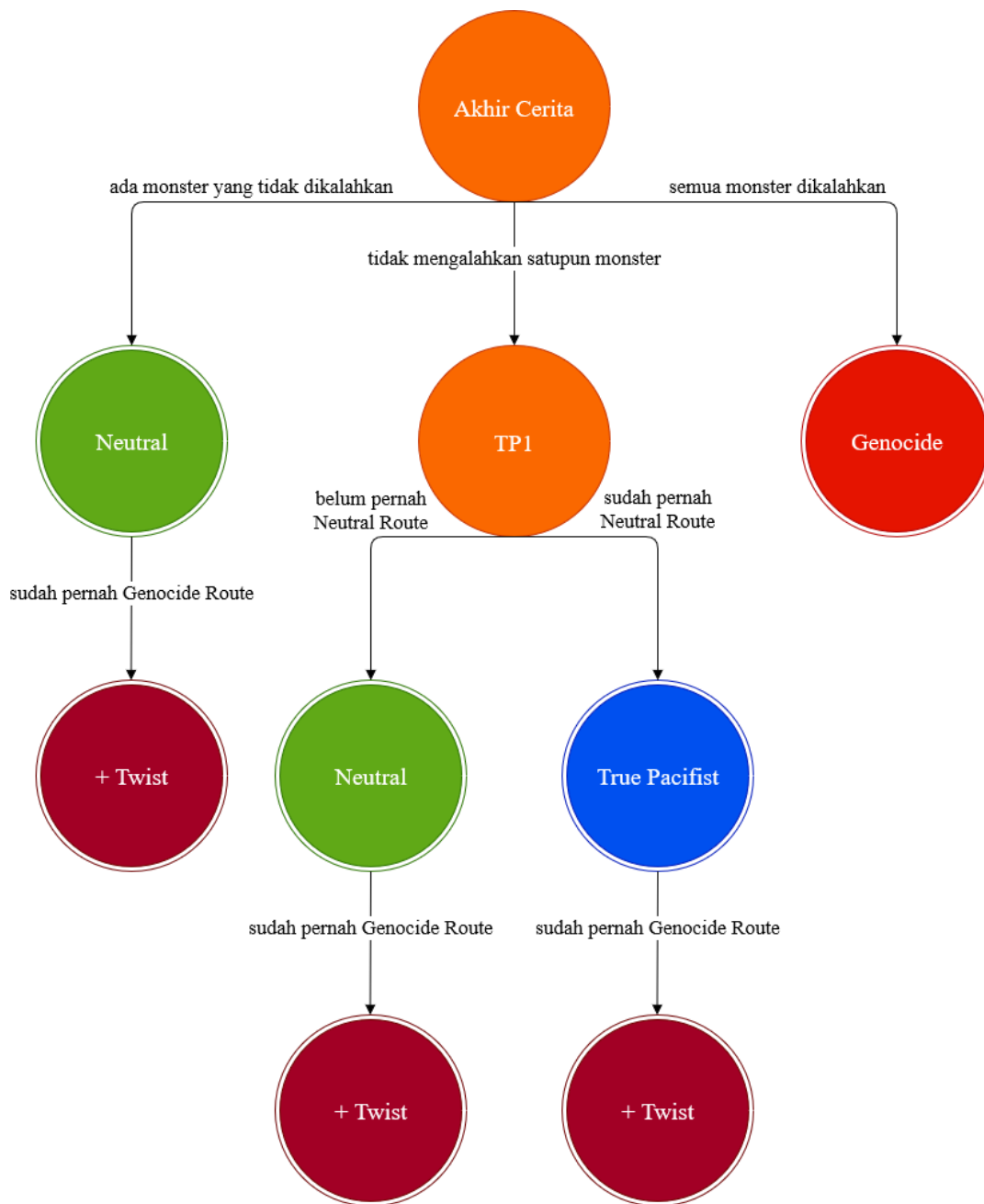
1. *fight*: menyerang musuh (pertarungan akan selesai ketika HP/nyawa musuh mencapai angka 0),
2. *act*: berinteraksi dengan musuh (berbicara atau melakukan sejumlah hal lainnya yang tidak menyakiti musuh),
3. *item*: menggunakan barang yang dikumpulkan selama eksplorasi, dan
4. *mercy*: pemain tidak menyerang musuh, dan jika musuh tidak menyerang balik, pertarungan langsung selesai. Sebagai catatan, opsi *mercy* hanya akan menyebabkan pertarungan selesai apabila pemain telah melakukan serangkaian interaksi spesifik dengan musuh.

Jika pertarungan tidak selesai setelah fase giliran pemain, akan dimulai fase giliran musuh. Musuh akan menyerang pemain dan pemain harus mengelak dari serangan-serangan musuh. Jika pemain terkena serangan musuh, maka HP/nyawa pemain akan berkurang. Permainan akan berakhir ketika nyawa pemain mencapai angka 0; permainan akan diulang dari *savegame* terakhir.



Gambar 11. Fase giliran pemain (atas) dan fase giliran musuh (bawah). Pada fase giliran pemain, simbol hati bertindak sebagai pointer, sementara pada fase

IV. PEMODELAN POHON KEPUTUSAN PENENTUAN AKHIR CERITA *UNDERTALE*



Gambar 12. Pohon keputusan penentuan akhir cerita *Undertale*

Akhir cerita *Undertale* dapat dimodelkan oleh pohon keputusan di atas.

1. Setiap lingkaran dengan *border* putih menandakan bahwa keputusan sudah dapat di ambil ketika memasuki simpul tersebut.
2. Pewarnaan dibedakan antara ending *Neutral Route*, *True Pacifist Route*, dan *Genocide Route*.
3. *Twist* adalah sesuatu yang *mungkin* (tapi tidak harus) terjadi.

V. KESIMPULAN DAN SARAN

A. Kesimpulan

Pohon keputusan dapat digunakan untuk memodelkan penentuan akhir cerita *Undertale*

B. Saran

Selain pohon keputusan, penentuan akhir cerita *Undertale* juga dapat dilakukan menggunakan graf berarah saja. Studi lanjutan dapat dilakukan untuk menentukan model mana yang lebih bisa merepresentasikan penentuan akhir cerita *Undertale*

dengan baik.

VII. UCAPAN TERIMA KASIH

Penulis pertama-tama ingin mengucapkan syukur kepada Tuhan karena tanpa bimbingan dan penyertaan-Nya makalah ini tidak dapat selesai. Penulis juga ingin berterima kasih kepada seluruh tim pengajar IF2120 Matematika Diskrit Institut Teknologi Bandung, Semester I Tahun 2020/2021 yang telah mengajarkan ilmu-ilmu yang diperlukan dalam penulisan makalah ini.

REFERENCES

- [1] Valve 2015, *Undertale on Steam*, Steam, diakses pada 11 Desember 2020 <<https://store.steampowered.com/app/391540/Undertale/>>
- [2] R. Munir 2020, *Graf (Bagian 1)*, Homepage Rinaldi Munir, diakses pada 11 Desember 2020 <<http://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Graf-2020-Bagian1.pdf>>
- [3] R. Munir 2020, *Pohon (Bagian 1)*, Homepage Rinaldi Munir, diakses pada 11 Desember 2020 <<http://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Pohon-2020-Bag1.pdf>>
- [4] R. Munir 2020, *Pohon (Bagian 2)*, Homepage Rinaldi Munir, diakses pada 11 Desember 2020 <<http://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Pohon-2020-Bag2.pdf>>

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, 3 Desember 2020



Feralezer L. G. Tampubolon - 13519062