

Aplikasi *Decision Tree* dalam Perancangan Pergerakan dan Kemunculan Musuh dalam *Zombie Survival Game*

Billy Julius 13519094¹

Program Studi Teknik Informatika

Sekolah Teknik Elektro dan Informatika

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

¹13519094@std.stei.itb.ac.id

Abstract—*Decision Tree* atau pohon keputusan memiliki beragam aplikasi dalam dunia nyata. Namun, fakta bahwa *Decision Tree* juga dapat diaplikasikan pada perancangan *game* (permainan gawai) masih belum diketahui banyak orang. Sebagai dasar pemikiran dalam pembuatan suatu *game*, *Decision Tree* memiliki peran yang cukup besar dalam menggambarkan alur kerja *game* tersebut. Salah satu *genre* dari *game* yang memiliki aplikasi *Decision Tree* adalah *Zombie Survival Game*. *Decision Tree* dapat dipakai dalam merancang pergerakan musuh dalam arena permainan dan proses kemunculan musuh secara periodik.

Keywords—*Decision Tree*, perancangan, *game*, alur

I. PENDAHULUAN

Secara singkat, *Decision Tree* pada *video games* adalah saat satu atau lebih pilihan diberikan kepada pemain. Ketika pemain sudah memilih, maka alur keberjalanan *game* tersebut akan bergerak berbeda sesuai dengan pilihan yang dipilih oleh pemain. Faktanya, *Decision Tree* sudah sering digunakan dan dapat dijumpai aplikasinya pada banyak *game* yang sudah beredar di masyarakat.



Gambar 1 : Contoh *Decision Tree* dalam salah satu *game* terkenal *Zelda*

(Sumber : <https://www.zeldadungeon.net/why-a-choice-driven-story-game-would-make-a-great-zelda-spin-off/>)

Selain diaplikasikan pada pemain, *Decision Tree* juga dapat dijadikan alat bantu perumusan dasar perancangan suatu *game*. Dalam proses perancangan sebuah *game*, diperlukan perumusan alur *game* yang kokoh agar *game* juga dapat dibuat

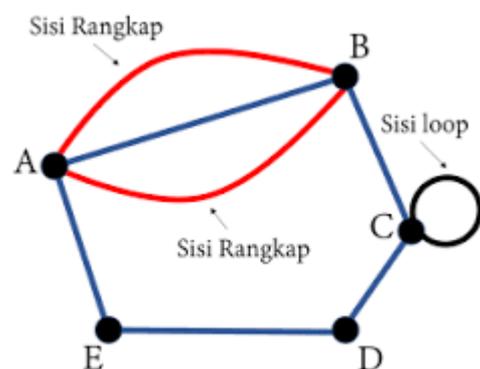
secara baik dan terstruktur. Dengan dasar yang kuat, para *programmer* yang bertugas untuk membuat algoritma-algoritma dapat membuat *code* yang bersih dan teratur. Salah satu metode yang dapat digunakan untuk membantu perumusan alur kerja *game* adalah *Decision Tree*. Pada makalah ini, penggunaan *Decision Tree* akan lebih dipusatkan kepada perancangan dasar alur *game*, yaitu *Zombie Survival Game*.

II. GRAF

A. Definisi Graf

Graf adalah salah satu bentuk visualisasi yang berpusat pada simpul dan sisi. Graf $G = (V, E)$ didefinisikan dengan V adalah himpunan tidak kosong dari simpul-simpul dan E adalah himpunan sisi yang menghubungkan sepasang simpul.

Sisi gelang (*loop*) didefinisikan sebagai sisi yang menghubungkan suatu simpul ke simpul itu sendiri. Sisi ganda (*rangkap*) didefinisikan sebagai pasangan dua sisi atau lebih yang menghubungkan dua simpul berbeda. Perhatikan Gambar 2 dibawah!



Gambar 2 : Ilustrasi sisi ganda dan sisi gelang
(Sumber : <https://mathcyber1997.com/soal-latihan-dan-penyelesaian-teori-dasar-graf-graph-basic-theory/>)

Ada beberapa jenis graf :

1. Graf sederhana

Graf sederhana adalah graf yang tidak memiliki sisi ganda ataupun gelang.

2. Graf tak sederhana

Graf tak sederhana adalah graf yang memiliki sisi ganda ataupun sisi gelang.

3. Graf ganda

Graf ganda adalah graf tak sederhana yang memiliki sisi ganda.

4. Graf semu

Graf semu adalah graf tak sederhana yang memiliki sisi gelang.

5. Graf berarah

Graf berarah adalah graf yang sisi-sisinya diberi arah dari simpul awal menuju simpul tujuan.

6. Graf tak berarah

Graf tak berarah adalah graf yang sisi-sisinya tidak diberi arah.

7. Graf berarah-ganda

Graf berarah-ganda adalah graf berarah yang memiliki sisi ganda.

B. Terminologi Graf

1. Ketetanggaan

Dua simpul dikatakan bertetangga jika kedua simpul tersebut dihubungkan oleh sisi yang sama.

2. Bersisian

Sebuah sisi dikatakan bersisian dengan simpul s_1 dan s_2 jika sisi tersebut menghubungkan s_1 dengan s_2 .

3. Simpul Terpencil

Sebuah simpul disebut simpul terpencil jika simpul tersebut tidak terhubung dengan sisi apapun.

4. Graf kosong

Graf kosong adalah graf yang tidak memiliki himpunan sisi.

5. Derajat

Derajat dari sebuah simpul adalah banyaknya sisi yang bersisian dengan simpul tersebut. Pada graf berarah, ada yang disebut dengan derajat masuk dan derajat keluar, sesuai dengan arah sisi yang bersisian dengan simpul tersebut.

6. Lintasan

Lintasan adalah rangkaian simpul-simpul yang membentuk jalan dari suatu simpul awal menuju simpul tujuan. Panjang dari suatu lintasan adalah jumlah sisi yang dilewati dari lintasan tersebut.

7. Sirkuit

Sirkuit adalah lintasan yang memiliki simpul awal dan simpul tujuan yang sama.

8. Keterhubungan

Dua buah simpul dikatakan terhubung jika ada lintasan yang menghubungkan kedua simpul tersebut. Pada graf berarah, ada dua simpul dikatakan terhubung kuat jika ada lintasan dari dan menuju masing-masing simpul.

9. Upagraf

Upagraf adalah bagian dari graf yang juga merupakan graf. Komponen dari sebuah graf didefinisikan sebagai jumlah maksimum upagraf terhubung dari graf tersebut.

10. Cut Set

Cut Set dari graf terhubung G adalah himpunan sisi yang bila dihapus akan menyebabkan G menjadi tidak terhubung.

11. Graf Berbobot

Graf berbobot adalah graf yang setiap sisinya diberi sebuah harga (bobot).

Untuk keperluan makalah ini, teori graf yang dipakai hanya akan mencakup pada jenis-jenis graf dan terminologi graf yang nantinya akan dipakai pada penjelasan teori pohon, maupun saat penjelasan aplikasi *Decision Tree* pada perancangan *Zombie Survival Game*.

III. POHON

Sebuah *pohon* adalah graf tak-berarah terhubung yang tidak mengandung sirkuit. Adapun kumpulan pohon saling lepas disebut dengan hutan.

Misalkan $G = (V,E)$ sebagai sebuah graf tak-berarah sederhana dengan jumlah simpul sebanyak n , maka pernyataan-pernyataan dibawah memiliki hubungan ekuivalen :

1. G adalah sebuah pohon
2. Setiap pasang simpul di G terhubung dengan lintasan tunggal
3. G terhubung dan memiliki $(n-1)$ buah sisi
4. G tidak mengandung sirkuit
5. Penambahan satu sisi pada G akan menghasilkan satu sirkuit
6. Semua sisi yang terdapat pada G adalah jembatan, yang berarti jika satu sisi dihilangkan maka akan ada simpul yang lepas.

A. Pohon Merentang

Pohon merentang dari sebuah graf terhubung adalah upagraf merentang yang berupa pohon. Pohon merentang dapat diperoleh dengan memutuskan sirkuit yang terdapat pada graf terhubung tersebut. Setiap graf terhubung memiliki minimal satu pohon merentang. Graf tak terhubung dengan k komponen memiliki k pohon merentang. Adapun kumpulan pohon merentang disebut dengan pohon merentang.

B. Pohon Berakar

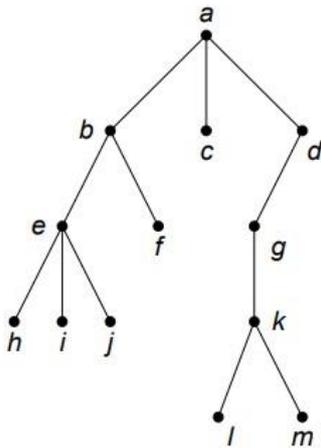
Pohon berakar adalah pohon yang salah satu simpulnya memiliki peran sebagai akar utama. Alur jalan dari pohon adalah teratur dari akar utama menuju daun terakhir.

Istilah-istilah yang terdapat pada pohon adalah sebagai berikut :

1. Anak : simpul yang memiliki parent
2. Parent : simpul yang memiliki sisi menuju anaknya
3. Sibling : dua simpul dikatakan memiliki hubungan sibling jika dan hanya jika kedua simpul tersebut memiliki parent yang sama
4. Daun : simpul yang tidak memiliki anak
5. Simpul Dalam : simpul yang memiliki anak
6. Derajat : derajat dari sebuah simpul adalah jumlah anak yang dimiliki oleh simpul tersebut.
7. Aras (Level) : Aras dari sebuah simpul adalah

tingkatan simpul tersebut dihitung dari akar utama

8. Tinggi : Aras maksimum dari sebuah pohon
9. Upapohon : Bagian dari pohon yang juga merupakan pohon (sub-pohon)



Rinaldi Munir/IF2120 Matematika Diskrit

Gambar 3 : Pohon Berakar

(Sumber :

<https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Pohon-2020-Bag2.pdf>)

Untuk keperluan penjelasan, perhatikan Gambar 3! Berdasarkan Gambar 3, a adalah parent dari b,c, dan d. Dengan demikian, b,c, dan d adalah anak dari a. Hubungan sibling dapat dilihat dari e dan f, namun e dan g bukan sibling. Daun-daun yang terdapat dari pohon tersebut adalah h,i,j,l, dan m. Derajat dari b adalah 2 dan derajat dari e adalah 3. Aras dari b adalah 2, sedangkan aras dari h adalah 4. Tinggi dari pohon tersebut adalah 5. Contoh upapohon dapat dilihat dari upapohon dengan akar utama e dan daun-daun h,i,j.

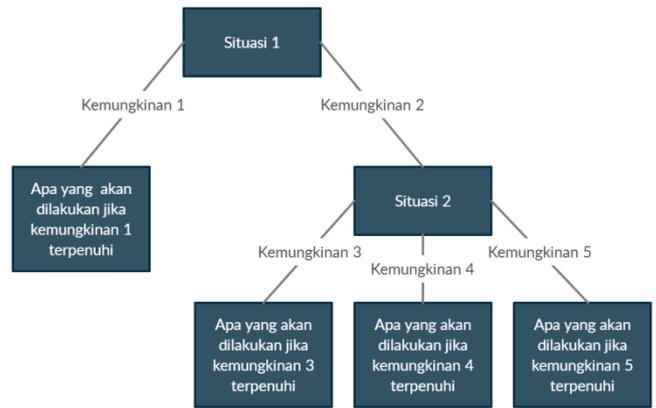
C. Pohon N-Ary

Pohon n-ary adalah pohon dengan jumlah anak maksimum yang dimiliki oleh sebuah simpul adalah n. Pohon n-ary yang penuh adalah pohon yang semua simpulnya memiliki n buah anak.

C. Decision Tree

Pohon keputusan (*Decision Tree*) adalah salah satu aplikasi pohon n-ary dan pohon berakar yang digunakan untuk menentukan alur keputusan dari situasi-situasi yang ada. Adapun pohon keputusan inilah yang akan digunakan untuk diaplikasikan sebagai perancangan *zombie survival game*.

Pohon keputusan menggunakan simpul-simpulnya sebagai input situasi yang akan ditanyakan dan sisi-sisi yang bercabang dari simpul tersebut adalah semua kemungkinan kondisi yang dapat muncul dari situasi tersebut. Untuk lebih jelasnya, perhatikan ilustrasi berikut!



Gambar 4 : Ilustrasi Decision Tree

(Sumber : Ilustrasi Penulis)

IV. PERANCANGAN ZOMBIE SURVIVAL GAME

A. Perancangan Game secara Umum

Game yang dirancang pada makalah ini adalah *zombie survival game*, dimana pemain berusaha bertahan hidup dengan menggunakan senjata untuk menembaki zombie yang berusaha mengalahkan pemain. *Game* ini memiliki latar belakang cerita dunia di masa depan yang hancur akibat adanya serangan senjata *bio* dari planet luar yang menyebabkan semua manusia yang terinfeksi menjadi kehilangan akal sehat dan berusaha untuk menyerang manusia lainnya. Manusia yang telah terinfeksi disebut dengan *zombie* dan para *zombie* memiliki pergerakan yang lambat dan jarak pandang yang terbatas. Namun, ada beberapa *zombie* yang sudah berevolusi dan memiliki kemampuan yang lebih tinggi daripada *zombie* lainnya.

Sebagai aksi mempertahankan diri, para ilmuwan berusaha menciptakan manusia buatan dengan imunitas terhadap infeksi *zombie*. Namun, dari ribuan manusia yang dijadikan eksperimen, hanya satu yang berhasil melewati semua tahap eksperimen dan berhasil menjadi kebal terhadap infeksi *zombie*. Para ilmuwan tentunya tidak menyalahkan hal ini, mereka langsung mencoba untuk menduplikasi manusia buatan tersebut. Namun, ketika saat-saat terakhir proses duplikasi akan dilakukan, pusat penelitian diserang oleh sekumpulan besar *zombie*. Semua harapan kemudian diberikan kepada manusia buatan untuk menahan serangan *zombie* sampai proses duplikasi selesai dan manusia dapat melawan kaum *zombie*.

Adapun perancangan *game* ini dibuat singkat, yaitu sebanyak 3 (tiga) buah tingkatan, atau biasa disebut dengan *level*. Jika manusia buatan dapat bertahan hidup dan membunuh semua *zombie*, maka proses duplikasi berhasil, menandakan kemenangan dan akhir dari *game*. Tentu saja, jika sebaliknya, manusia buatan gugur dan *zombie* berhasil masuk ke dalam pusat penelitian, harapan terakhir untuk selamat dari wabah *zombie* ini akan lenyap.

B. Perancangan Player

Player akan mengambil peran sebagai manusia buatan yang mengemban tanggung jawab untuk melindungi pusat penelitian sampai proses duplikasi berhasil. *Player* akan dilengkapi

dengan senjata dan pelindung tubuh untuk proses perlawanan terhadap *zombie*. Walaupun kebal terhadap infeksi, *player*, sebagai manusia buatan, masih dapat terluka akibat serangan dari *zombie*. Oleh karena itu, *player* harus hati-hati dan menjaga jarak dengan *zombie* agar dapat berhasil melindungi pusat penelitian.

Seiring berjalannya permainan, *player* akan mendapatkan *experience point* yang akan terus bertambah. Setelah mencapai jumlah tertentu, *player* dapat memilih untuk mendapatkan kemampuan tertentu yang dapat membantunya untuk melawan *zombie*, menjadikan *game* ini memiliki sedikit unsur *roguelike*. *Roguelike* merupakan salah satu genre *game* dengan ciri khas berpusat pada pengembangan kemampuan *player* seiring melewati *level-level* yang semakin sulit, serta saat *player* kalah, maka permainan akan berakhir dan harus diulang dari awal lagi.

C. Perancangan *Zombie* sebagai Musuh

Zombie sebagai musuh dari *player* akan memiliki beberapa unsur yang perlu dipertimbangkan oleh *player* sebelum mengambil keputusan, yaitu jarak pandang dan jarak serang. *Player* yang berada pada jarak pandang *zombie* akan membuat *zombie* menjadi aktif dan bergerak mengejar *player*. Sesaat ketika *player* sudah berada dalam jarak serang, *zombie* akan melancarkan serangan kepada *player*. Berdasarkan kemampuan bertarungnya, ada empat jenis *zombie* yang masing-masing memiliki tingkat kemampuan yang berbeda, dengan ketentuan semakin tinggi *level zombie*, maka semakin tinggi pula kemampuan bertarung yang dimilikinya.

1. *Zombie Level 1*

Zombie ini merupakan *zombie* dengan kemampuan bertarung yang paling rendah. *Zombie* ini memiliki jenis serangan *Basic Scratch*, menyerang *player* dengan jarak dekat.

2. *Zombie Level 2*

Zombie level 2 memiliki kecepatan, yang paling baik dibandingkan jenis *zombie* lainnya. Selain itu, serangan dari jenis *zombie* ini adalah *Midnight Strike*, yaitu menyerang *player* dengan jarak dekat, namun memiliki tingkat kerusakan yang sangat tinggi. Selain itu, jika *player* tidak berhasil menghindari serangan ini, jarak pandang *player* akan dikurangi untuk beberapa waktu.

3. *Zombie Level 3*

Zombie ini merupakan *zombie* dengan tingkat kemampuan bertarung yang paling tinggi. Walaupun tidak memiliki kecepatan sebaik *zombie level 2*, *zombie level 3* memiliki jarak pandang dan jarak serang yang sangat jauh. Jenis serangan yang dimiliki adalah *Corrosive Spit*, yang memberikan efek memperlambat pergerakan *player* jika *player* tidak berhasil

menghindari serangan dari *zombie* ini.

4. *Boss Zombie*

Zombie ini hanya akan muncul sekali selama keberjalanan *game*, yaitu saat semua *zombie* lain telah berhasil dikalahkan. *Zombie* ini memiliki nyawa dan tingkat kerusakan yang sangat tinggi. Ia juga memiliki berbagai jenis kemampuan, selain *Basic Punch*, seperti *Jump Crush*, yaitu diam selama 1 detik, sebelum melompat ke lokasi *player* dan mengakibatkan kerusakan yang sangat besar jika *player* tidak dapat menghindarinya. Kemampuan lain yang dimiliki *Boss Zombie* adalah *Ground Stomp*, yaitu diam selama 1 detik, sebelum menginjak daerah sekitar *Boss Zombie*, *player* yang terkena serangan ini akan mengalami kerusakan yang besar dan tidak dapat bergerak selama 2 detik. Selain itu, *Boss Zombie* juga dapat memunculkan anak buahnya, yaitu *Zombie Level 1*, *Zombie Level 2*, dan *Zombie Level 3* secara random. Jika *player* berhasil mengalahkan *Boss Zombie*, maka permainan akan berakhir dengan kemenangan.

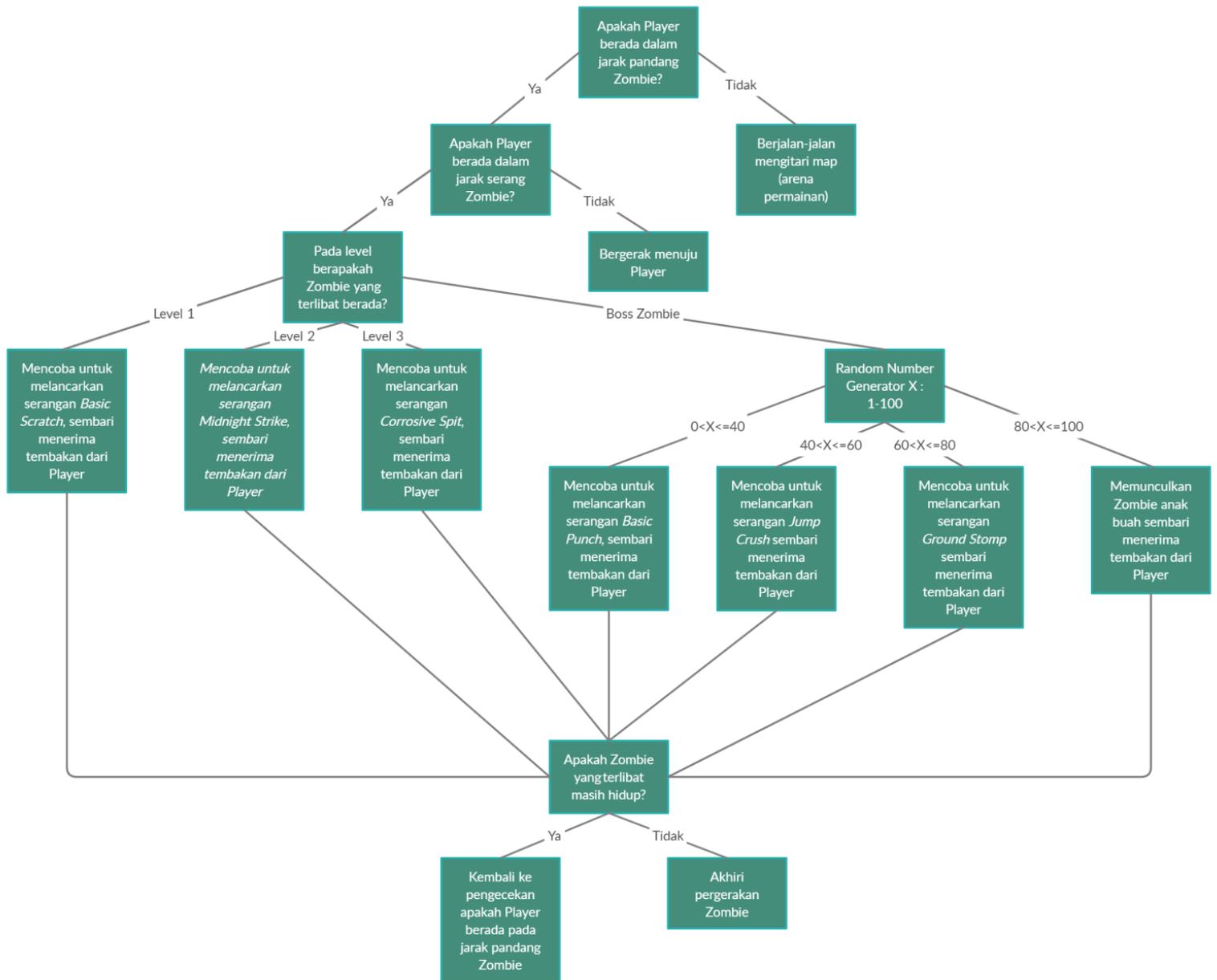
Sesuai dengan jenis *zombie*, *experience point* yang akan diberikan kepada *player* saat membunuh *zombie* akan berbeda pula. Jumlah *experience point* yang akan didapatkan *player* ketika membunuh *zombie* dengan tingkat kemampuan yang tinggi akan lebih besar dibandingkan dengan saat membunuh *zombie* dengan tingkat kemampuan yang rendah.

V. APLIKASI *DECISION TREE* DALAM PERANCANGAN PERGERAKAN DAN KEMUNCULAN MUSUH DALAM *ZOMBIE SURVIVAL GAME*

A. Pergerakan Musuh

Untuk pergerakan musuh, yaitu *zombie*, akan dilakukan beberapa tahap pemeriksaan sebelum menentukan aksi apa yang akan diambil oleh *zombie*. Pertama, pemeriksaan apakah *player* berada pada jarak pandang dari *zombie*. Jika tidak, maka *zombie* hanya akan berjalan tanpa tujuan, mengitari arena permainan. Jika ya, maka akan dilakukan pemeriksaan selanjutnya, yaitu pemeriksaan *player* berada dalam jarak serang dari *zombie*. Jika tidak, maka *zombie* akan berjalan ataupun berlari menuju *player*. Kedua pemeriksaan awal ini dilakukan karena adanya unsur jarak pandang dan jarak serang dalam perancangan *game* sebelumnya.

Jika *player* telah masuk ke dalam jarak pandang dan jarak serang *zombie*, *zombie* akan melancarkan aksi penyerangan ke *player*. Aksi ini berbeda-beda bergantung pada jenis *zombie* yang terlibat. Untuk *Zombie Level 1*, aksi penyerangan yang akan dilancarkan adalah *Basic Scratch*. Untuk *Zombie Level 2*, akan dilancarkan *Midnight Strike*. Untuk *Zombie Level 3*, akan



Gambar 5 : Decision Tree untuk Pergerakan Zombie
(Sumber : Ilustrasi Penulis)

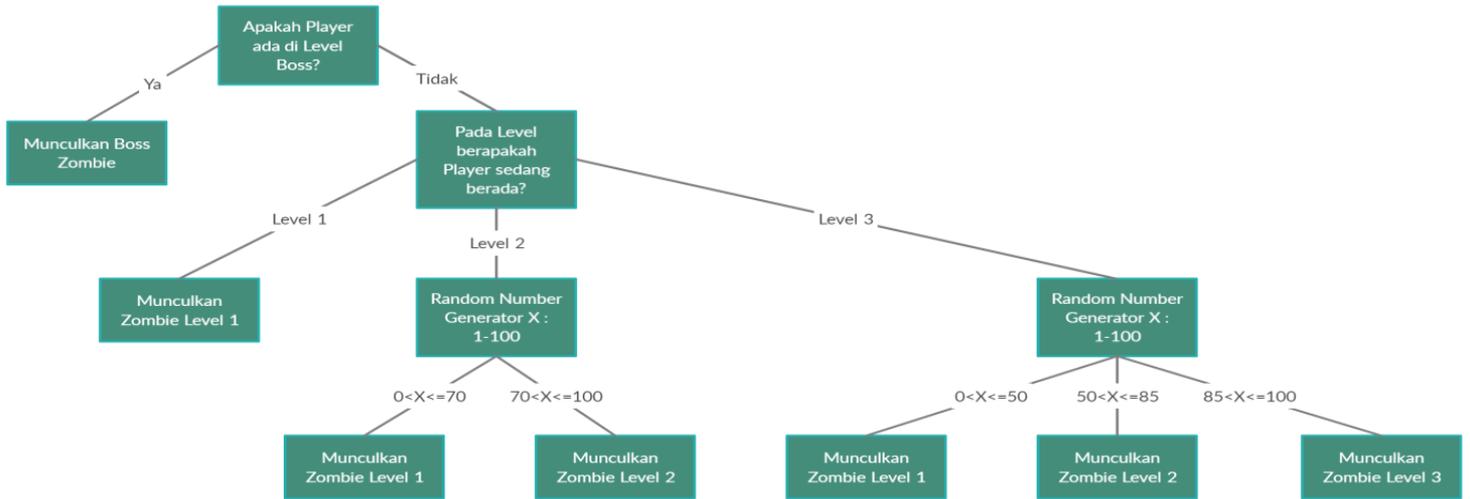
dilancarkan *Corrosive Spit*. Untuk *Boss Zombie*, aksi yang akan dilancarkan ditentukan berdasarkan peluang tertentu. Beberapa aksi yang dapat dilancarkan adalah *Basic Punch* (40%), *Jump Crush* (20%), *Ground Stomp* (20%), dan memunculkan anak buah zombie (20%). Untuk lebih jelasnya, perhatikan Gambar 5. Adapun model penggambaran pohon disederhanakan dengan sedikit menggabungkan elemen graf, agar tidak memakan tempat yang terlalu banyak.

Setelah melancarkan serangan kepada *player*, akan dilakukan pemeriksaan nyawa dari *zombie* yang terlibat. Pemeriksaan ini diperlukan karena *player* memiliki kebebasan untuk menembak *zombie* kapanpun. Saat *zombie* melancarkan serangan pun, *player* dapat menembak *zombie*. Jika *zombie* yang terlibat sudah berhasil dikalahkan oleh *player*, maka pergerakan dari *zombie* tersebut akan berakhir. Jika belum, maka pergerakan *zombie* akan diulang kembali dari awal

pemeriksaan keberadaan *player* dalam jarak pandang *zombie*. Proses ini akan terus berlangsung sampai *zombie* yang terlibat telah berhasil dikalahkan, atau *player* telah gugur. Namun, *Decision Tree* yang dirancang pada Gambar 4 hanya berpusat pada perilaku *Zombie*. Oleh karena itu, kemungkinan *player* gugur tidak dimasukkan pada *Decision Tree* tersebut. *Decision Tree* yang dijabarkan pada Gambar 5 berlaku untuk semua jenis *zombie* dan pergerakan yang akan dilakukan oleh *zombie* tersebut.

B. Kemunculan Musuh

Decision Tree juga dipakai dalam merancang proses pemunculan musuh, yaitu *zombie*. Kemunculan *zombie* akan dilakukan berdasarkan pada *level* berapa *player* sedang berada. Jika *player* berada pada tahap akhir, yaitu melawan *Boss*



Gambar 6 : Decision Tree untuk Kemuculan Zombie
(Sumber : Ilustrasi Penulis)

Zombie, maka akan dimunculkan satu *Boss Zombie* sebagai lawan dari *player*. Jika *player* berhasil mengalahkan *Boss Zombie* tersebut, maka *game* akan berakhir dengan kemenangan.

Jika *player* berada pada *Level 1*, maka *zombie* yang akan dimunculkan hanya berupa *Zombie Level 1*. Saat *player* mencapai *Level 2*, maka *zombie* yang dimunculkan akan bervariasi antara *Zombie Level 1* dan *Zombie Level 2*, dengan peluang masing-masing 70% dan 30%. Untuk *Level 3*, *zombie* akan muncul dengan lebih bervariasi lagi, yaitu *Zombie Level 1* dengan peluang 50%, *Zombie Level 2* dengan 35%, dan *Zombie Level 3* dengan peluang 15%. Proses kemunculan ini akan dipanggil pada titik-titik pada arena permainan yang akan ditentukan secara acak pula.

Dengan menggunakan *Decision Tree* untuk merancang pergerakan *zombie* dan proses kemunculan *zombie* pada *Zombie Survival Game*, konsep dasar dari *game* tersebut menjadi lebih jelas. Proses pemrograman akan menjadi lebih terstruktur dengan mengikuti alur dari *Decision Tree* yang telah dibuat.

V. KESIMPULAN

Perancangan konsep pada sebuah *game* bersifat penting dan mendasar. Hal ini dikarenakan dengan adanya konsep dasar yang kuat, proses implementasi ke dalam pemrograman akan menjadi lebih mudah, teratur, dan terstruktur. Dalam merealisasikan hal ini, *Decision Tree* dapat digunakan sebagai alat bantu dalam mempersiapkan konsep dasar sebuah *game*. Dengan menggunakan *Decision Tree* untuk memberikan daftar semua kemungkinan pilihan yang mungkin dipilih oleh pemain, ataupun untuk menentukan keberjalanan tiap bagian dari *game*, alur *game* akan menjadi lebih jelas dan mempermudah proses pemrograman dari *game* tersebut.

Hal ini membuktikan bahwa teori-teori dari mata kuliah

Matematika Diskrit memiliki banyak penerapan dalam berbagai bidang kehidupan. Salah satu teorinya, pohon keputusan (*Decision Tree*) dapat diaplikasikan pada perancangan suatu *game*. Dengan memiliki konsep dasar percabangan, *Decision Tree* menjadi alat bantu yang sangat efektif dan dapat mempermudah perancangan sebuah *game*. Dengan demikian, dapat disimpulkan bahwa teori-teori informatika sangat diperlukan dalam dunia perteknologian yang akan terus berkembang.

VI. KATA PENUTUP

Penulis mengucapkan syukur kepada Tuhan Yang Maha Esa atas berkat yang diberikannya sehingga penyusunan makalah ini dapat diselesaikan dengan baik. Penulis juga berterima kasih kepada orang tua dan keluarga yang memberi dukungan dan doa kepada penulis. Tidak lupa rasa hormat dan terima kasih juga penulis tujukan kepada Ibu Harlili dan semua dosen pengampu mata kuliah Matematika Diskrit IF2120 Semester I 2020/2021 lainnya yang telah memberikan pembelajaran kepada penulis selama satu semester dengan baik. Penulis juga berterima kasih kepada teman-teman yang senantiasa menyemangati penulis.

DAFTAR PUSTAKA

- [1] Antonette, Brandon. Decision Tree in Video Games. Diakses melalui <https://medium.com/@antoneb/decision-trees-in-video-games-3ea3f251f96e> pada 5 Desember 2020 pukul 22.32 WIB.
- [2] Santoso, Erick, dkk. Pembuatan Game dengan Menerapkan Metode Decision Tree: UCB1, untuk Menentukan Pemilihan Strategy dalam AI. Diakses melalui <https://media.neliti.com/media/publications/108803-ID-pembuatan-game-dengan-menerapkan-metode.pdf> pada 5 Desember 2020 pukul 22.41 WIB.
- [3] Munir, Rinaldi. Pohon Bagian 1. Diakses melalui <https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Pohon-2020-Bag1.pdf> pada 5 Desember 2020 pukul 22.43 WIB.
- [4] Munir, Rinaldi. Pohon Bagian 2. Diakses melalui <https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Pohon-2020-Bag2.pdf> pada 5 Desember 2020 pukul 22.43 WIB.

- [5] Munir, Rinaldi. Graf Bagian 1. Diakses melalui <https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Graf-2020-Bagian1.pdf> pada 5 Desember 2020 pukul 22.50 WIB
- [6] Munir, Rinaldi. Graf Bagian 2. Diakses melalui <https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Graf-2020-Bagian2.pdf> pada 5 Desember 2020 pukul 22.50 WIB
- [7] Munir, Rinaldi. Graf Bagian 3. Diakses melalui <https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Graf-2020-Bagian3.pdf> pada 5 Desember 2020 pukul 22.50 WIB
- [8] Mchugh, Alex. What is Roguelike. Diakses melalui <https://www.greenmangaming.com/blog/what-is-a-roguelike/> pada 8 Desember 2020 pukul 22 56.
- [9] Anonim. Why A Choice-driven Story Game would Make A Great Zelda Spin-off. Diakses melalui <https://www.zeldadungeon.net/why-a-choice-driven-story-game-would-make-a-great-zelda-spin-off/> pada 8 Desember 2020 pukul 23 01.

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.

Medan, 8 Desember 2020



Billy Julius 13519094