

Pengimplementasian Teori Graf dan Pohon dalam *Video Game Pokemon*

Raden Fajar Hadria Putra - 13511076
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
fajarghosty@gmail.com

Abstrak — Makalah ini berisikan tentang bagaimana teori-teori yang dijelaskan dalam mata kuliah Struktur Diskrit digunakan di luar mata kuliah tersebut. Dalam makalah ini, dijelaskan bagaimana teori pohon, pohon biner, graf, logika, dan himpunan digunakan dalam pembuatan sekumpulan video game RPG (role-playing game) yang secara umum diberi judul Pocket Monsters (lebih dikenal dengan singkatan Pokemon).

Seperti game RPG pada umumnya, Pokemon menggunakan sistem turn-based yang merupakan karakteristik utama video game bergenre RPG. Namun berbeda dengan video game RPG lainnya, Pokemon memiliki beberapa formula yang unik, seperti penggunaan sistem evolusi, empat jenis serangan non-standar, penggunaan sistem individual value, penggunaan sistem held item, dan sebagainya. Formula-formula tersebut dapat dibuat dengan mengimplementasikan sejumlah teori yang berkaitan dengan logika, himpunan, pohon dan sebagainya.

Pada awalnya, video game Pokemon terdahulu tidak sekompleks judul-judul yang sekarang. Namun, selama lima belas tahun, pengembangan dilakukan untuk meningkatkan kompleksitas dan ketidakpastian kemungkinan yang akan terjadi. Hingga sekarang, kemungkinan-kemungkinan yang dapat terjadi dalam Pokemon mendekati milyaran kemungkinan.

I. PENDAHULUAN

A. Latar Belakang

Berawal dari ketertarikan penulis dengan *video game Pokemon* sejak lama, berdasarkan pengalaman bermain beberapa kali penulis menemukan banyak kemungkinan yang dapat terjadi dalam satu kali permainan. Penulis menemukan bahwa banyak aspek-aspek dari *Pokemon* yang merupakan implementasi dari teori-teori yang penulis pahami dari Struktur Diskrit.

Penulis memilih *video game Pokemon* sebagai pokok bahasan dari makalah ini karena banyaknya aspek-aspek yang dapat dikaji yang berkaitan dengan teori-teori dalam Struktur Diskrit. Pemilihan judul “Pengimplementasian Teori Graf dan Pohon dalam *Video Game Pokemon*” dilakukan karena penulis hanya akan memfokuskan kajian mengenai penggunaan teori graf dan pohon sehingga kajian akan lebih mendalam dan terfokus pada satu arah.

B. Tujuan

Tujuan dari pembuatan makalah ini adalah sebagai berikut:

- Memenuhi tugas makalah mata kuliah Struktur Diskrit
- Memahami bagaimana pengimplementasian teori graf dan pohon dalam *game Pokemon*.
- Membuka ruang lingkup strategi dalam *game Pokemon* dan memudahkan pemain dapat menyelesaikan berbagai masalah dalam *game Pokemon*.
- Mengambil inspirasi implementasi teori pohon dan graf dalam *game Pokemon* untuk memodelkan *video game* yang lebih inovatif.

II. TEORI DASAR

A. Teori Graf

Graf G didefinisikan sebagai pasangan himpunan (V, E) , ditulis dengan notasi $G = (V, E)$, yang dalam hal ini V adalah himpunan yang tidak kosong dari simpul-simpul (*node*) dan E merupakan himpunan sisi (*edges*) yang menghubungkan sepasang simpul.

Berdasarkan ada tidaknya gelang atau sisi ganda pada suatu graf, secara umum graf dapat digolongkan menjadi tiga jenis:

1. Graf Sederhana
Graf sederhana adalah graf yang tidak mengandung gelang maupun sisi-ganda. Pada graf sederhana, sisi adalah pasangan pasangan tak-terurut. Jadi, penulisan sisi (u, v) sama dengan sisi (v, u) .
2. Graf Ganda
Graf ganda adalah graf yang mengandung sisi ganda. Sisi ganda yang menghubungkan sepasang simpul bisa lebih dari dua buah. Sisi ganda dapat diasosiasikan sebagai pasangan tak-terurut yang sama. Setiap graf sederhana merupakan graf ganda, namun tidak semua graf ganda merupakan graf sederhana.
3. Graf Semu
Graf semu adalah graf yang mengandung gelang (*loop*). Gelang yang dimaksud adalah sisi yang menghubungkan satu simpul pada dirinya sendiri.

Berdasarkan orientasi arah pada sisi, secara umum graf dibedakan atas dua jenis:

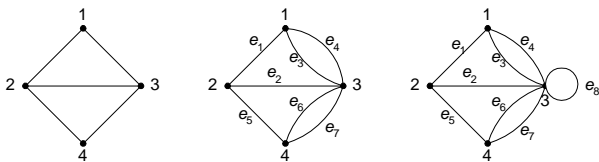
1. Graf Tak-Berarah

Graf tak-berarah adalah graf yang sisinya tidak mempunyai orientasi arah. Pada graf tak-berarah, urutan pasangan simpul yang dihubungkan oleh sisi tidak diperhatikan.

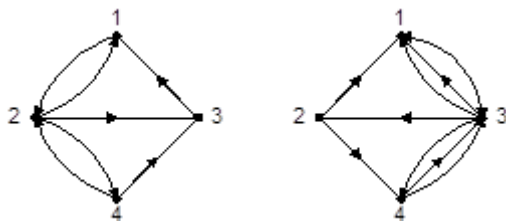
2. Graf Berarah

Graf berarah adalah graf yang setiap sisinya diberikan orientasi arah. Pada graf berarah, (v, u) dan (u, v) menyatakan dua busur (sisi berarah) yang berbeda. Untuk busur (u, v) , simpul u dinamakan simpul asal dan simpul v dinamakan simpul terminal.

Definisi graf dapat diperluas sehingga mencakup graf-ganda berarah. Pada graf-ganda berarah, gelang dan sisi ganda diperbolehkan ada. [1]



Gambar 2.1 Kiri-Kanan : Graf Sederhana, Graf Ganda dan Graf Semu [1]

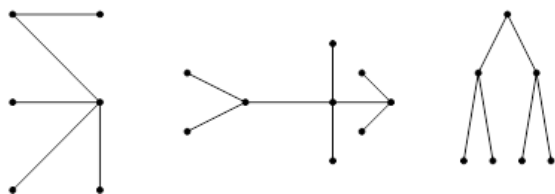


Gambar 2.2 Kiri-Kanan : Graf Berarah dan Graf-Ganda Berarah [1]

B. Teori Pohon

Pohon adalah graf tak-berarah terhubung yang tidak mengandung sirkuit. Karena definisi pohon diacu dari teori graf, maka sebuah pohon dapat mempunyai hanya sebuah simpul tanpa sebuah sisipun. Dengan kata lain, jika $G = (V, E)$ adalah pohon, maka V tidak boleh berupa himpunan kosong.

Beberapa pohon dapat membentuk hutan, Hutan adalah kumpulan pohon yang saling lepas. Hutan dapat dikatakan juga sebagai graf tak-terhubung yang tidak mengandung sirkuit, yang dalam hal ini setiap komponen di dalam graf terhubung tersebut adalah pohon.



Gambar 2.3 Contoh sebuah hutan yang terdiri dari beberapa pohon [1]

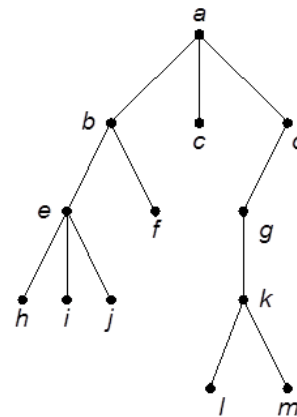
Terdapat beberapa jenis pohon, yaitu :

1. Pohon Berakar

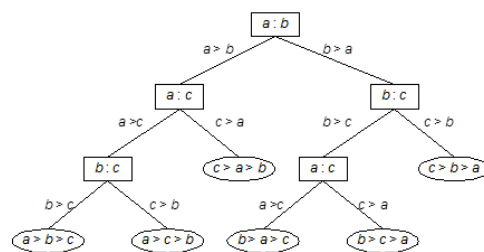
Pohon berakar adalah pohon yang sebuah simpulnya diperlakukan sebagai akar dan sisi-sisinya diberi arah menjauh dari akar. Akar mempunyai derajat-masuk sama dengan nol dan simpul-simpul lainnya berderajat-masuk sama dengan satu. Simpul yang mempunyai derajat-keluar nol disebut daun. Simpul yang mempunyai derajat-keluar tidak sama dengan nol disebut simpul dalam atau simpul cabang. Variasi pohon berakar yang semua simpul cabangnya memiliki derajat-keluar tidak lebih dari dua disebut pohon biner, sedangkan pohon yang simpul cabangnya dapat memiliki lebih dari dua derajat disebut pohon m-nary.

2. Pohon Keputusan

Pohon keputusan digunakan untuk memodelkan persoalan yang terdiri dari serangkaian keputusan yang mengarah ke solusi. Setiap cabang dari akar menentukan ke cabang manakah yang harus dituju pada kondisi tertentu. [1]



Gambar 2.4 Contoh pohon berakar, dengan a sebagai akar dari pohon, $c, f, h, i, j, l,$ dan m sebagai daun dan sisanya sebagai cabang. [1]



Gambar 2.5 Contoh pohon keputusan dengan persoalan bagaimana urutan dari terbesar antara $a, b,$ dan $c.$ [1]

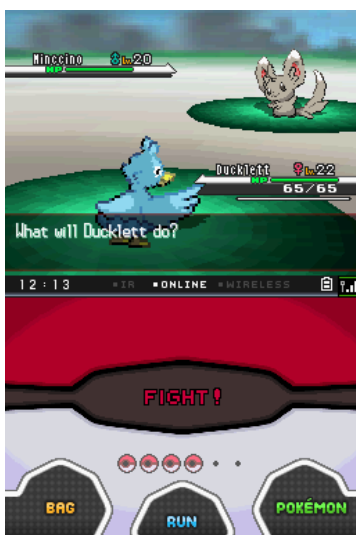
C. Pokemon

Pocket Monsters (lebih dikenal dengan singkatannya *Pokemon*) merupakan *franchise video game* yang bergenre RPG (*role-playing game*) yang dikembangkan oleh *Game Freak* dan *Nintendo* menggunakan game *handheld Game Boy, Game Boy Color, Game Boy Advance, dan Nintendo DS*. *Video game Pokemon* pertama kali dirilis pada tahun 1996 dengan judul *Pokemon Red* dan *Pokemon Green* di Jepang. Pada Tahun 1998, *Pokemon Red Version* dan *Pokemon Blue Version* dirilis di Amerika dan mulai dikenal di seluruh dunia.



Gambar 2.6 Tampak depan *Pokemon Red Version* dan *Pokemon Blue Version* (Sumber : bulbapedia.bulbagarden.net)

Pokemon menceritakan sebuah dunia yang memiliki banyak monster berwujud seperti binatang yang dapat mengeluarkan berbagai jurus elemen dan dapat berevolusi yang dinamakan *pokemon*. *Pokemon* tersebut dapat dikontrol oleh manusia melalui sebuah alat berbentuk bola yang disebut *pokeball*. *Pokemon* tersebut biasanya ditandingkan oleh *pokemon* lain untuk menentukan *pokemon* mana yang lebih kuat. Sistem pertarungan yang diprogram dalam *video game*-nya sendiri berbentuk *turn-based*, yaitu setiap *pokemon* bergiliran melancarkan serangannya masing-masing sampai salah satu *pokemon* kalah, atau disebut *fainted* (pingsan).



Gambar 2.8 Tampilan Pertarungan dalam *Pokemon* (Sumber : bulbapedia.bulbagarden.net)

Sejak *video game* tersebut dirilis, *Pokemon* mendapat berbagai sekuel yang dirilis di berbagai platform, seperti *Pokemon Gold and Silver Version* untuk *Game Boy Color*, *Pokemon Diamond and Pearl Version*, serta *Pokemon Black and White Version* untuk *Nintendo DS*. *Pokemon* memiliki banyak penggemar di seluruh dunia yang mengakibatkan *Nintendo* mengadakan pertandingan *Pokemon* di seluruh dunia setiap tahun dengan memanfaatkan koneksi *wireless* untuk menghubungkan para pemain di seluruh dunia. [2]



Gambar 2.7 Tampak depan *Pokemon Black Version* dan *Pokemon White Version* (Sumber : bulbapedia.bulbagarden.net)

III. PENGIMPLEMENTASIAN GRAF PADA POKEMON

A. Pembuatan Peta dalam Game

Sama seperti game RPG lainnya, *Pokemon* menawarkan kebebasan dalam menjelajahi tempat-tempat yang tersedia dalam game dengan batasan harus menyelesaikan cerita (*plot*) yang harus dijalani sepanjang permainan. Sebetulnya, konsep penjelajahan dari *Pokemon* tidak jauh berbeda dengan permainan ber-genre *sandbox*. Namun, perbedaannya adalah jika *genre sandbox* membebaskan pemain untuk menjelajahi seluruh sudut lokasi dalam game dengan sedikit pengecualian, *Pokemon* hanya dapat menjelajahi tempat-tempat tertentu, ditandai dengan ada jalan setapak, rerumputan, bangunan-bangunan, dan hutan terbuka. Tempat-tempat yang tertutup oleh pohon-pohon ataupun tebing-tebing tanpa ada jalan alternatif tidak dapat dijelajahi dengan cara apapun kecuali dengan *debugging*.



Gambar 3.1 Tampilan Tempat Penjelajahan (Sumber : bulbapedia.bulbagarden.net)

Jika dilihat secara garis besar, terdapat dua jenis tempat, yaitu perkotaan/area dan rute. Perkotaan/area

dapat direpresentasikan dengan simpul sebuah graf, sedangkan rute-rute direpresentasikan dengan sisi. Apabila direpresentasikan seperti ini, peta yang memuat lokasi-lokasi di *Pokemon* akan berbentuk sebuah graf sederhana seperti contoh di bawah ini:



Gambar 3.2 Peta lokasi *Unova Region* dalam game *Pokemon Black 2 and White 2 Version* (Sumber : bulbapedia.bulbagarden.net)

Jika melihat peta di atas, peta tersebut membentuk sebuah graf tak terhubung dengan bulatan biru sebagai simpul dan garis kuning sebagai sisi. Graf yang tak terhubung dengan cara spesial, karena tidak dapat diseberangi dengan berjalan, seperti meminta tolong seorang pelaut untuk menyeberangi sungai. Rute di kanan bawah pada gambar di atas dapat diasumsikan sebagai gelang, karena dengan menjelajahi rute tersebut pada akhirnya pemain akan kembali ke tempat dimana pemain mulai menjelajahi rute tersebut. Dalam graf peta lokasi, pada umumnya setiap simpul akan memiliki derajat lebih dari dua, dan apabila terdapat simpul yang hanya memiliki satu derajat, simpul tersebut merepresentasikan sebuah kota yang hanya memiliki sebuah jalur masuk dan keluar dan hanya terhubung dengan satu rute.

Dalam pengembangan peta lokasi saat pembuatan game, penggambaran peta dengan sistem graf dibutuhkan agar dapat mendesain rute-rute agar pemain tidak langsung menemukan jalan pintas yang merupakan alternatif dari rute yang harus dilalui dan dapat menyebabkan penghindaran cerita yang seharusnya terjadi di rute yang seharusnya dilalui. Sistem graf pun dapat mengurangi kemungkinan banyak lokasi yang dibuat hanya memiliki satu jalur masuk dan keluar, sehingga membingungkan pemain dalam menentukan kemana pemain harus berjalan selanjutnya.

B. Desain Efektivitas Elemen dalam Pertarungan

Pokemon memiliki sistem pertarungan yang mirip dengan permainan suit, yaitu terdapat sesuatu yang lebih kuat apabila melawan sesuatu yang lainnya. Dalam *Pokemon*, sistem ini diperluas dengan menyediakan 17 jenis elemen berbeda yang satu sama lain dapat lebih kuat maupun lebih lemah. Contohnya, serangan yang memiliki

elemen api (*fire*) memiliki lebih kuat apabila dilancarkan kepada *pokemon* yang memiliki elemen rumput (*grass*). Setiap serangan hanya dapat memiliki satu buah elemen, sedangkan *pokemon* dapat memiliki maksimal dua elemen.



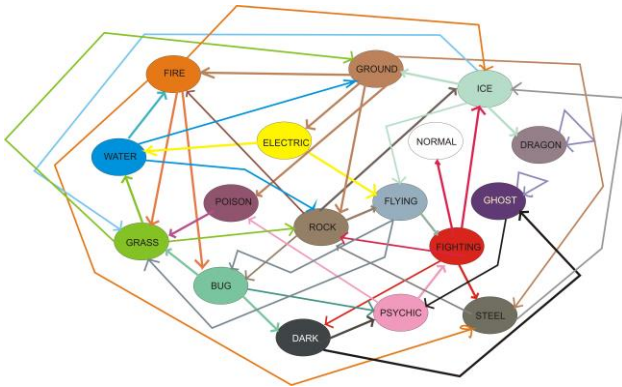
Gambar 3.3 Jumlah tipe elemen yang digunakan dalam *Pokemon* (Sumber : bulbapedia.bulbagarden.net)

Seperti yang telah disebutkan, setiap elemen memiliki kekuatan dan kelemahan masing-masing terhadap elemen lainnya. Apabila satu serangan menyerang *pokemon* yang memiliki tipe yang lemah terhadap tipe serangan tersebut, *damage* akan diberikan dua kali lebih besar dari biasanya (disebut *super effective*) dan berlaku kelipatannya. Sebaliknya, jika Contohnya, *pokemon* bernama Gyarados memiliki dua tipe, yaitu *water* dan *flying*, dan kedua elemen tersebut lemah terhadap tipe *electric*. Apabila Gyarados diserang oleh serangan *Charge Beam* yang bertipe *electric*, maka Gyarados akan menerima *damage* empat kali lebih besar dari biasanya. Berikut adalah *type chart* yang memetakan kekuatan dan kelebihan elemen terhadap satu sama lain.

	Normal	Fighting	Flying	Poison	Ground	Rock	Bug	Ghost	Steel	Fire	Water	Grass	Electric	Psychic	Ice	Dragon	Dark
Normal	1x	1x	1x	1x	1x	1x	1x	0x	1x	1x	1x	1x	1x	1x	1x	1x	1x
Fighting	2x	1x	1/2x	1/2x	1x	2x	1/2x	0x	2x	1x	1x	1x	1x	1/2x	2x	1x	2x
Flying	1x	2x	1x	1x	1x	1/2x	2x	1x	1/2x	1x	1x	2x	1/2x	1x	1x	1x	1x
Poison	1x	1x	1x	1/2x	1/2x	1/2x	1x	1/2x	0x	1x	1x	2x	1x	1x	1x	1x	1x
Ground	1x	1x	0x	2x	1x	2x	1/2x	1x	2x	1/2x	1x	1/2x	2x	1x	1x	1x	1x
Rock	1x	1/2x	2x	1x	1/2x	1x	2x	1x	1/2x	2x	1x	1x	1x	1x	2x	1x	1x
Bug	1x	1/2x	1/2x	1/2x	1x	1x	1x	1/2x	1/2x	1/2x	1x	2x	1x	2x	1x	1x	2x
Ghost	0x	1x	1x	1x	1x	1x	1x	2x	1/2x	1x	1x	1x	1x	2x	1x	1x	1/2x
Steel	1x	1x	1x	1x	1x	2x	1x	1/2x	1/2x	1/2x	1/2x	1x	1x	1x	2x	1x	1x
Fire	1x	1x	1x	1x	1x	1/2x	2x	1x	2x	1/2x	1/2x	2x	1x	1x	1x	2x	1/2x
Water	1x	1x	1x	1x	2x	2x	1x	1x	1x	2x	1/2x	1/2x	1x	1x	1x	1x	1/2x
Grass	1x	1x	1/2x	1/2x	2x	2x	1/2x	1x	1/2x	1/2x	2x	1/2x	1x	1x	1x	1x	1x
Electric	1x	1x	2x	1x	0x	1x	1x	1x	1x	1x	2x	1/2x	1/2x	1x	1x	1x	1/2x
Psychic	1x	2x	1x	2x	1x	1x	1x	1x	1/2x	1x	1x	1x	1x	1/2x	1x	1x	0x
Ice	1x	1x	2x	1x	2x	1x	1x	1x	1/2x	1/2x	1/2x	2x	1x	1x	1x	1/2x	2x
Dragon	1x	1x	1x	1x	1x	1x	1x	1x	1/2x	1x	1x	1x	1x	1x	1x	1x	2x
Dark	1x	1/2x	1x	1x	1x	1x	1x	2x	1/2x	1x	1x	1x	1x	2x	1x	1x	1/2x

Gambar 3.4 *Type chart* (Tabel keefektifan) elemen di *Pokemon* (Sumber : bulbapedia.bulbagarden.net)

Tabel tersebut menunjukkan semua relasi keefektifitas semua elemen serangan terhadap semua elemen pokemon, apakah elemen serangan tersebut lebih kuat (*super effective*), lebih lemah (*not very effective*), atau tidak berpengaruh (*not effective*) terhadap pokemon yang diserang. Masing-masing bentuk relasi dapat diubah menjadi bentuk graf berarah, dengan arahnya menunjukkan bahwa tipe serangan tersebut *super effective*, *not very effective*, atau *not effective* terhadap elemen pokemon yang diserang, tergantung relasi manakah yang diimplementasikan dalam graf. Berikut merupakan contoh bentuk graf untuk relasi lebih kuat:



Gambar 3.5 Graf keefektifan elemen terhadap elemen lain. Dalam versi ini panah arah menunjukkan bahwa serangan bertipe simpul asal *super effective* terhadap simpul terminal

Jika diperhatikan, terdapat beberapa sifat dalam graf tersebut. Graf tersebut merupakan graf terhubung lemah, karena setiap elemen berhubungan dengan minimal satu elemen lainnya, namun tidak terdapat lintasan berarah karena terdapat dua simpul yang bukan merupakan simpul asal menuju simpul lain, yaitu *normal* dan *dragon*.

Pembentukan graf keefektifan ini sangat berguna dalam pembuatan *type chart* dalam sebuah game. Karena *developer* dapat melihat dari graf apakah distribusi keefektifan sudah seimbang atau belum, apakah suatu simpul elemen merupakan simpul asal dari terlalu banyak elemen atau terlalu sedikit, menandakan bahwa apakah serangan bertipe elemen tersebut *super effective* terlalu banyak atau terlalu sedikit pokemon dengan elemen yang lemah terhadap elemen tersebut. Lalu graf keefektifan elemen versi *super effective*, *not very effective* dan versi *not effective* dapat dibandingkan, apakah serangan bertipe elemen tersebut *super effective* terhadap terlalu banyak elemen lain dan *not very effective* terhadap terlalu sedikit elemen lain atau sebaliknya. Hal ini sangat berpengaruh terhadap jalannya pertarungan, karena apabila distribusi keefektifan tidak seimbang, akan terdapat pokemon yang terlalu kuat (*overpowered*) karena elemennya *super effective* terhadap banyak elemen lainnya dan *not very effective* terhadap sedikit elemen lainnya. Penyeimbangan tersebut dapat diperkirakan pula dari kemampuan-

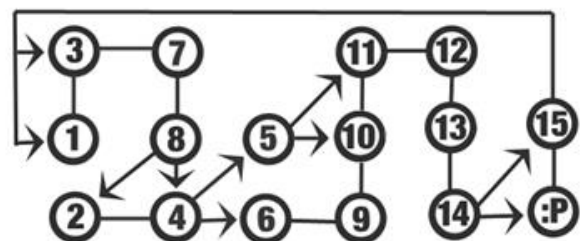
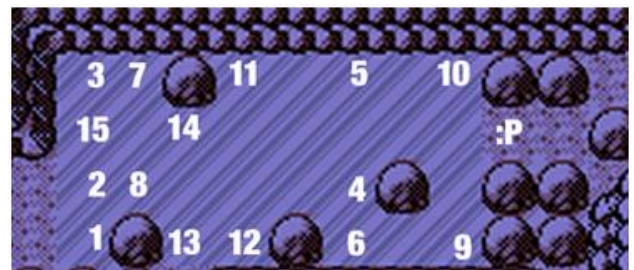
kemampuan yang elemen tersebut miliki. Contohnya elemen *poison super effective* terhadap hanya satu elemen karena sebagian besar serangan bertipe *poison* dapat mengurangi nyawa lawannya secara otomatis setiap putaran tanpa mempertimbangkan *type chart*.

Graf tersebut pun dapat membantu para pemain, karena pemain dapat mengatur strategi mengenai elemen apakah yang banyak dipakai pemain lainnya karena keunggulan elemen tersebut dan elemen apakah yang harus ia fokuskan agar dapat mengalahkan pemain lain dengan mudah.

C. Desain Puzzle dalam Penjelajahan Jalan

Dalam *Pokemon*, seringkali pemain menemukan sebuah jalan yang memiliki trik khusus (*puzzle*) agar dapat dilalui, seperti banyaknya batu penghalang jalan sempit yang dapat didorong, jalur es licin yang membuat pemain meluncur ke satu arah dan melarang pemain untuk bergerak bebas sampai menabrak suatu rintangan, dan sejumlah tanah rapuh yang membuat lubang apabila dilangkahi oleh pemain dan membuat pemain jatuh. *Puzzle* tersebut banyak ditemui dalam gua, menara tua, ataupun arena *boss*.

Untuk mendesain *puzzle* tersebut, *developer* harus dapat memperkirakan apakah *puzzle* tersebut dapat dipikirkan oleh pemain atau tidak, sehingga *puzzle* tersebut tidak boleh terlalu kompleks. Di sini teori graf akan berguna untuk mendesain *puzzle* tersebut. Contohnya, untuk mendesain *puzzle* es licin, asumsikan tempat dimana pemain dapat menabrak sesuatu dan dapat berganti arah sebagai simpul, dan jalur dimana pemain dimungkinkan meluncur sebagai sisi. Agar pemain dapat melewati *puzzle* tersebut, harus dipastikan bahwa terdapat satu atau lebih simpul yang bersisian dengan simpul keluar, yaitu suatu simpul dengan derajat satu. Lalu yang paling penting adalah *puzzle* tersebut harus memiliki lintasan berarah dari simpul mulai menuju simpul keluar.

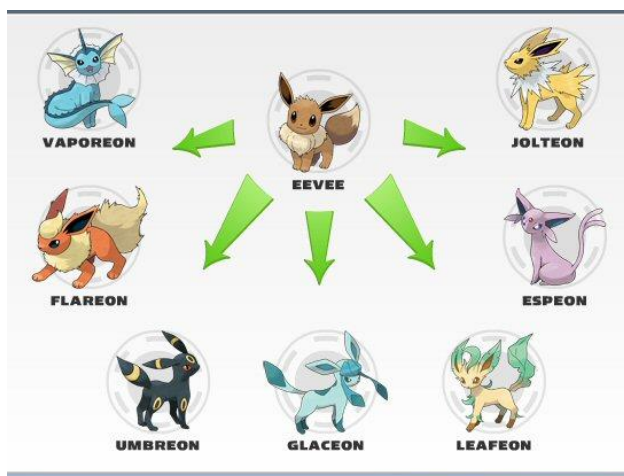


Gambar 3.6 Contoh Penggambaran graf dalam *puzzle*

es licin, dengan angka menggambarkan simpul dimana pemain dapat berhenti, sisi berarah menggambarkan bahwa pemain hanya dapat bergerak satu arah, sisi tek-berarah menggambarkan pemain dapat bergerak dua arah, dan (:P) menggambarkan titik keluar.

IV. PENGIMPLEMENTASIAN POHON PADA EVOLUSI DALAM POKEMON

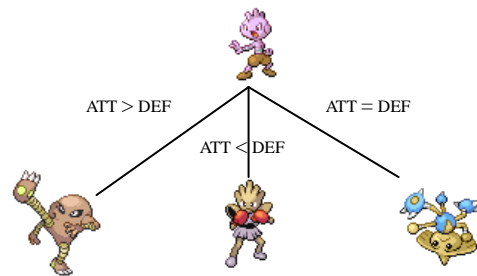
Hampir semua pokemon memiliki evolusi, artinya pokemon tersebut dapat berubah menjadi bentuk yang lebih besar dan kuat. Sebagian besar pokemon hanya dapat berubah menjadi satu bentuk yang lebih besar, namun ada beberapa pokemon yang dapat berubah menjadi lebih dari satu bentuk, tergantung kondisi apakah yang membuat pokemon tersebut berubah. Perjalanan evolusi tersebut dapat dibentuk sebagai pohon pendek, dengan pokemon tersebut berupa akar dan evolusinya berupa cabang dan daunnya. Evolusi pokemon berbentuk pohon karena pokemon yang sudah berevolusi tidak dapat kembali menjadi wujud asalnya, sama seperti cabang yang tidak dapat kembali menuju akar.



Gambar 3.7 Pokemon Eevee sebagai akar dan evolusi-evolusinya sebagai daun.

Dari semua pokemon yang memiliki evolusi bercabang, terdapat pokemon yang evolusinya berbentuk pohon keputusan pendek, artinya pokemon tersebut berevolusi menjadi pilihan pokemon yang berbeda tergantung suatu kondisi. Contohnya, pokemon Wurmpe saat mencapai level tujuh dapat berevolusi menjadi antara Cascoon atau Silcoon tergantung *personality value* (sebuah poin individual dari suatu pokemon) yang dimiliki Wurmple tersebut. Contoh lainnya adalah pokemon Tyrogue saat mencapai level 20 dapat berevolusi menjadi antara Hitmonlee, Hitmonchan, atau Hitmontop tergantung poin *attack* dan *defence* yang dimiliki Tyrogue tersebut. Apabila poin *attack* lebih tinggi daripada *defence*, Tyrogue akan berevolusi menjadi Hitmonlee, jika sebaliknya, akan berevolusi menjadi Hitmonchan, dan jika poin *attack* dan *defence* sama, maka akan berevolusi

menjadi Hitmontop.



Gambar 3.8 Pokemon keputusan evolusi pokemon Tyrogue saat mencapai level 20

V. KESIMPULAN

Kesimpulan dari makalah ini adalah sebagai berikut:

- Teori graf dalam membuat game *Pokemon* dipakai untuk membuat lokasi-lokasi dalam game, membuat *type graph* keefektifan tipe elemen dalam *Pokemon* sebagai pembandingan dan pengkalkulasi, serta untuk membuat *puzzle* di jalan-jalan yang dilalui pemain dalam *Pokemon*.
- Teori pohon dan pohon keputusan digunakan untuk mendesain evolusi pokemon yang bercabang

VI. UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada Allah SWT yang telah memberi kesempatan kepada penulis untuk menulis makalah ini, Yang Terhormat Bapak Ir. Rinaldi Munir, M.T. dan Ibu Dra. Harlili S., Msc. yang telah memberikan tugas makalah kepada penulis, serta Satoshi Tajiri, pencetus *video game Pokemon* yang telah menginspirasi penulis untuk membuat makalah ini.

REFERENSI

- [1] Munir, Rinaldi. 2005. *Matematika Diskrit*. Bandung: Penerbit Informatika
- [2] <http://bulbapedia.bulbagarden.net/wiki/Pokemon>

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, 19 Desember 2012

ttd



Raden Fajar Hadria Putra – 13511076