

Penerapan STIMA dalam Final Fantasy Tactics

Kevin Indra S / 13510022

Program Studi Teknik Informatika

Sekolah Teknik Elektro dan Informatika

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

¹author@itb.ac.id

Abstract—Penerapan algoritma tidak hanya dalam pemrograman saja, dalam bermain pun prinsip algoritma dapat diterapkan. Dalam hal ini game strategi merupakan game yang paling mudah untuk menerapkan algoritma. Salah satu contoh game strategi yang dapat menggunakan prinsip algoritma adalah *Final Fantasy Tactics*. Ada beberapa algoritma yang dapat diterapkan dalam game ini, namun yang akan di bahas hanya 2, yaitu algoritma *Greedy* dan algoritma *Back-tracking*.

Index Terms—.

I. INTRODUCTION



Final Fantasy Tactics atau yang dikenal sebagai FFT merupakan game bertipe *Tactical RPG*. Final Fantasy Tactics bercerita tentang seorang keturunan bangsawan yang terlibat dalam konflik militer antara dua kubu yang memperebutkan tahta kerajaan *Ivalice*.

FFT memiliki fitur-fitur umum seperti game strategi lainnya seperti *Experience Point* (Exp) dan *Job System*, dan fitur-fitur khusus dalam game ini seperti *Charge Time* (CT) dan *Job Points* (JP). CT digunakan untuk melakukan suatu aksi atau *Act*. Ketika CT suatu karakter telah mencapai 100 point, karakter tersebut dapat

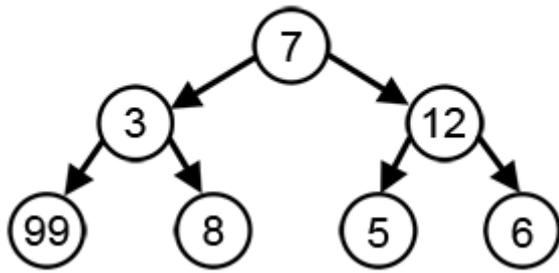
melakukan *Act*. CT akan bertambah setiap putaran atau *Turn*, penambahan CT akan bergantung pada atribut *speed* milik karakter.

Setiap karakter dalam FFT memiliki 2 macam level. Level yang pertama adalah level karakter lalu yang kedua adalah *Job level*. Atribut karakter dipengaruhi oleh level karakter dan *Job* dari karakter tersebut. Exp digunakan untuk meningkatkan level karakter, sedangkan JP digunakan untuk meningkatkan *Job level*. Misalkan seorang karakter dengan job Knight akan memiliki kekuatan fisik lebih tinggi dari karakter yang memiliki job Magician, tetapi Magician memiliki kekuatan sihir lebih tinggi dari job Knight. Level karakter akan mempengaruhi seberapa besar penambahan kekuatan fisik bagi knight dan penambahan kekuatan sihir bagi Magician. Jika seorang Knight yang sudah mencapai level 30 ingin berubah menjadi seorang Magician hal ini sangat tidak menguntungkan, karena sejak awal penambahan kekuatan fisik dari karakter tersebut lebih besar dari kekuatan sihir karakter tersebut. Namun hal tersebut bukan berarti tidak mungkin.

Hal unik lainnya dalam FFT adalah sistem *skill* yang digunakan dapat merupakan kombinasi *skill* dari beberapa *job*, misal seorang *knight* dapat memiliki *skill* samurai atau sebaliknya. Pemilihan kombinasi dari *skill* yang tepat akan menghasilkan karakter yang hebat. Untuk memiliki *skill* dari job lain kita harus menjadi master dalam *job* tersebut dengan cara mengumpulkan JP sebanyak mungkin untuk membuka *skill-skill* dari job tersebut. Setiap kali kita berhasil melakukan sebuah *Act* kita akan mendapatkan Exp, dan JP.

II. DASAR TEORI

Algoritma Greedy merupakan algoritma yang mencari optimum lokal dengan harapan mendapat optimum global.



Contoh sederhana Greedy. Dengan menggunakan algoritma Greedy solusi maksimum yang didapat adalah $7 + 12 + 6 = 25$, namun solusi maksimum yang sesungguhnya adalah $7 + 3 + 99 = 109$.

Algoritma *back-tracking* atau runut balik adalah algoritma yang berbasis pada DFS. Algoritma *back-tracking* dapat mencari solusi lebih cepat daripada algoritma *brute-force* karena algoritma ini tidak mencari semua kemungkinan solusi yang ada.

Prinsip *back-tracking* :

1. Solusi dicari dari akar ke daun seperti DFS. Simpul-simpul yang sudah terbentuk dinamakan simpul hidup. Simpul hidup yang diperluas dinamakan simpul-E.
2. Setiap simpul hidup yang tidak mengarah ke solusi menjadi simpul mati, simpul mati tidak akan diperluas lagi.
3. Bila pencarian berakhir pada simpul mati, proses pencarian akan kembali ke simpul sebelumnya dan memulai pencarian yang baru.
4. Pencarian dihentikan ketika solusi ditemukan atau tidak ada simpul hidup yang dapat di perluas lagi.

III. PENERAPAN ALGORITMA DALAM FFT

A. Penerapan Greedy dalam pemilihan Tim

Dalam game FFT kita dapat memiliki hingga 32 karakter, namun dalam bertarung kita hanya boleh memilih 3 – 6 dari 32 karakter.

Dalam hal ini algoritma Greedy dapat digunakan untuk pemilihan karakter dalam 1 tim. Pemilihan karakter dibagi 3, 2 orang pertama sebagai penyerang lalu 2 orang sebagai bertahan kemudian 2 orang sisanya sebagai pendukung dalam tim (Komposisi dari 2 tipe karakter dapat disesuaikan dengan kondisi).

Untuk memilih karakter bertahan penggunaan algoritma Greedy dipakai untuk memilih karakter dengan *Hit Points* (HP) terbesar, jika ada dua karakter memiliki HP sama besar maka dipilih karakter dengan *Defense* (Def) yang terbesar. Sedangkan untuk penyerang dipilih karakter dengan serangan (Atk) dan akurasi yang besar, dalam hal ini Atk dan akurasi sama

penting karena bila memiliki serangan yang besar namun tidak dapat mengenai lawan akan percuma demikian juga sebaliknya jika serangan tepat tetapi tidak melukai lawan juga percuma. Sebagai pendukung dalam tim dibutuhkan karakter dengan *Magic Points* (MP) dan *Magic Attack* (Mgc Atk) yang besar.



B. Penggunaan *Back-tracking* dalam pemilihan Job

Pada Gambar *Job Tree*, Job dibagian kiri (dimulai dari *Squire*) merupakan job dengan kekuatan fisik yang lebih besar daripada job dibagian kanan (dimulai dari *Chemist*), namun job dibagian kanan memiliki kekuatan sihir yang lebih besar dari job di sebelah kiri. Angka pada panah merupakan job level yang dibutuhkan untuk membuka job baru. Job awal yang diberikan hanya ada 2 yaitu *Squire* dan *Chemist*. Sedangkan Mime memiliki status yang seimbang dan job yang paling sulit didapat, karena memiliki *requirement* paling banyak dan job yang “*serba bisa*”, kemampuan *mime* dalam meniru kemampuan lawan menjadikan *mime* karakter yang sulit dibaca.

Penerapan algoritma *back-tracking* dilakukan untuk cara tercepat mencapai *job* yang diinginkan. Misal jika ingin memiliki *job lancer* maka yang diperlukan adalah mencapai *job thief* dengan job level 3, *archer* job level 2 dan *squire* job level 2. Kita tidak perlu menjadi *knight*, *monk*, atau *geomancer*.

IV. KESIMPULAN

Penerapan Strategi Algoritma tidak hanya pada programming saja, banyak hal dapat dilakukan dengan menggunakan prinsip-prinsip algoritma bahkan dalam bermain game pun dapat diterapkan.

SUMBER

http://en.wikipedia.org/wiki/Final_Fantasy_Tactics
http://images1.wikia.nocookie.net/__cb20121027063907/finalfantasy/images/5/51/FFT-Job-Tree.png
<http://www.popgunchaos.com/wp-content/uploads/2010/11/Final-Fantasy-Tactics-2.jpg>
<http://upload.wikimedia.org/wikipedia/commons/8/8c/Greedy-search-path-example.gif>
Algoritma dan Struktur Data – Rinaldi Munir.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, 21 Desember 2012

ttd

A handwritten signature in black ink that reads "Kevin". The signature is stylized with a vertical line on the left and a horizontal line extending to the right.

Kevin Indra S.
13510022

