

INTISARI

CASE BASED REASONING UNTUK IDENTIFIKASI PENYAKIT TANAMAN KAKAO MENGGUNAKAN *NEAREST NEIGHBOR*

Oleh

Muhammad Anshar

18/433785/PPA/05600

Dungeon merupakan *level* dalam *game* yang terdiri dari kumpulan ruang dan pintu, yang terdapat rintangan di dalamnya. Untuk membuat *level* yang bagus, memerlukan waktu yang banyak. Dengan *Procedural Content Generation* (PCG), *dungeon* bisa dibuat secara otomatis. Salah satu pendekatan dalam PCG untuk membuat *level* adalah *progressive*. pendekatan *progressive* menghasilkan *timeline* sebagai representasi interaksi di dalam *game*. Representasi *graph timeline* yang berbentuk satu garis lurus bagus untuk konten *endless runner*. Tapi untuk konten *dungeon*, *level* yang terbentuk sangat linear. Pada penelitian ini representasi *timeline* diubah menjadi *cyclic graph*. *Cyclic graph* dibentuk menggunakan algoritma *graph grammar*. Penelitian ini bertujuan untuk membangun *dungeon* yang tidak linear dan minim dari jalan buntu. Untuk menghilangkan linearitas pada *dungeon*, maka percabangan dalam *dungeon* perlu dibentuk. Tahapan yang dilakukan dalam penelitian ini adalah perancangan *graph grammar rule*, *generate graph* sebanyak populasi, evaluasi *graph* berdasarkan nilai *fitness*, dan pembentukan *dungeon*. Ada empat fungsi yang digunakan untuk menentukan nilai *fitness*, yaitu *shortest vertices*, *average duration*, *replayability*, dan *variation*. Hasil pengujian pendekatan *progressive* dengan *cyclic graph* 100% selalu menghasilkan *dungeon* dengan percabangan. Sedangkan *Dungeon* yang menggunakan representasi linear tidak dapat menghasilkan percabangan dan selalu menghasilkan 2 jalan buntu.

Kata Kunci: Dungeon, Generative Grammars, Graph , Procedural Content

ABSTRACT

PROGRESSIVE CONTENT GENERATION BASED ON CYCLIC GRAPH FOR GENERATE DUNGEON

By

Muhammad Anshar

18/433785/PPA/05600

Dungeon is level in game consisting collection of rooms and doors with obstacles inside. To make good level, takes a lot of time. With Procedural Content Generation (PCG), dungeons can be created automatically. One of the approaches in PCG to create levels is progressive. Progressive approach produces timeline as representation of the interactions in the game. Timeline representation that is in the form of one straight line is good for endless runner, but for dungeon, the levels are linear. In this research, the timeline is changed to cyclic graph. Cyclic graph is formed using graph grammar algorithm. This research aims to build dungeon that has not linear and minimal dead ends. To eliminate linearity in dungeons, branching in dungeons needs to be formed. The steps carried out in this research are designing graph grammar rules, generating population of graphs, evaluating graphs based on fitness values, and building dungeons. Four functions are used to determine the fitness value: shortest vertices, average duration, replayability, and variation. The results of testing the progressive approach with a cyclic graph always produce branched dungeons. Whereas a Dungeon that uses a linear representation cannot generate branches and always generates 2 dead ends.

Keywords: Dungeon, Generative Grammars, Graph , Procedural Content