

Määrittelydokumentti

Monen luolastotason proseduraalinen etukäteinen luominen. Kahden eri menetelmän vertailu tuotannon nopeudessa. Luolastoon luotujen huoneiden väliset käytävät on tarkoitus toteuttaa molemmille algoritmeille samalla tavalla, ettei käytävien toteutus ns. vaikuta lopputulokseen.

Käytetty ohjelmointikieli:

Java

Muut kielet:

Python (välttävä)

Algoritmit:

BSP-puu, Voronoi

Ongelma:

Mahdollisimman monen luolastotason proseduraalinen etukäteinen luominen.

BSP-puu on ilmeisesti hyvin tyypillinen valinta luolastojen toteutukseen.

Voronoi voi olla tehokkaampi ja tuottaa visuaalisesti erilaisen lopputuloksen.

I/O:

Tulostus olisi tarkoitus toteuttaa niin, että luolastoja voi käydä läpi. Syötteitä tarvitaan hyvin luultavasti tulosteiden läpikäymisessä.

Aika- ja tilavaativuudet:

Tarkkaa aika- tai tilavaativuutta en osaa vielä tässä vaiheessa sanoa, mutta tekemäni tutkimuksen perusteella aikavaativuudet olisivat jotain tällaisia:

BSP-puu aikavaativuus: $O(n)$

Voronoi aikavaativuus: Riippuen toteutuksesta $O(n \log n)$ - $O(n^2)$

Lähteet:

<https://eskerda.com/bsp-dungeon-generation/>

<https://gamedevelopment.tutsplus.com/tutorials/how-to-use-bsp-trees-to-generate-game-maps--gamedev-12268>

<https://courses.cs.washington.edu/courses/cse326/00wi/projects/voronoi.html>

Opinto-ohjelma:

Tietojenkäsittelytieteen kandidaatti

Projektin dokumentaatioissa käytetty kieli:

Suomi (tarvittaessa mahdollisuus käyttää myös englantia)