

Viikkoraportti 3

Tämä viikko meni suurimmaksi osaksi algoritmin keksimiseen, joka erottaa kaikki solut toisistaan. En löytänyt siihen mitään valmista algoritmia, joten kehitin siihen oman. Algoritmin toiminta on seuraavanlainen:

Toista niin kauan kun soluja on päällekin

1. Valitse solu, joka on toisen solun kanssa päällekin, jonka keskipiste on lähimpänä ympyrän keskipistettä (jonka sisälle solut generoidaan).
2. Valitse solu, joka on kohdassa 1. valitun solun kanssa päällekin, ja jonka keskipiste on lähimpänä ympyrän keskipistettä.
3. Siiretään kohdassa 2. valittua solua (kauempaa ympyrän keskipisteestä) pois päin ympyrästä sen akselin mukaan, jolla huoneet leikkaa vähemmän.
4. Päivitetään solujen tiedot niiden päällä olevista soluista.

Algoritmi ei varmasti ole tehokkain, joka tämän ongelman ratkaisee, mutta toistaiseksi en ole keksinyt parempaa. Sen aikavaativuutta en ole vielä analysoinut. En saanut vielä toteutettua itse algoritmia, mutta nyt kun idea on selvillä, saan sen pian tehtyä.

Sain tällä viikolla toteutettua soluille collision detectionin, tein testejä ja lisäsin dokumentaatiota. Löysin myös jokusen bugin jotka sain korjattua. Tein myös oman toteutukseni ArrayListille. Tämä oli tähän mennessä hankalin viikko, sillä jouduin kehittämään solujen erottamiseen oman algoritmin. Ensi viikon tehtävät ovat tähän mennessä melko selvät. Yllä olevan algoritmin toteuttaminen, sekä minimikeko-tietorakenteen toteuttaminen, jota tulen käyttämään solujen valitsemiseen. Aikaa käytin tällä viikolla työhön noin 20 tuntia, josta suuri osa algoritmin suunnitteluun. Pyrin saamaan suunnittelemani algoritmin toteutettua ennen ensimmäistä vertaisarviointia, jotta koodissani olisi sentään jotain algoritmillista.