

Tällä viikolla työ ei edennyt ihan suunnitelmien mukaisesti, mutta viikko oli kuitenkin hyvinkin opettavainen.

Lähdettiin ehkä hieman liian nopeasti suoraan kehittämään varsinaisen algoritmin toteuttamista, ilman sen suurempaa suunnittelua. Ja tästä johtuen, hakattiinkin päätä seinään tuloksettomana, kunnes palattiin takaisin lähtöpisteeseen. Hyvä suunnittelu ja sen toteuttaminen ajoissa ennakkoon pitäisi aina nostaa hyvinkin tärkeäksi osaksi ohjelmointia.

Tutustuttiin entistä enemmän valittuun "Ray Casting" -laskentamenetelmään netissä, ja myös hyvin mielenkiintoiseen artikkeliin siitä, miten ensimmäiset 3D tietokonepelit (viit. Wolfenstein 3D, itsekin sitä pelanneena) hyödynsivät samaa laskentamenetelmää pelikenttensä projisoinnissa.

Tästä kaikesta innoittamana, suunnittelupöydälle otettiin nyt aivan tavallista ruutupaperia, johon hahmoteltiin yksinkertaisia polygonikarttoja piirtäen, ja sitä miten valitun pisteen olemassa olo kartan sisä- ja ulkopuolella vaikuttaisi aina polygonikartan reunoilla kulkemiseen, kun aina kuljetaan kartan kahden pisteen välisten viivojen mukaisesti. Huomattiin, että Ray Casting -menetelmän horisontaalinen leikkaus polygonin reunan kanssa tapahtuisi siis aina silloin, jos haetun pisteen y-koordinaatti kuuluisi näiden kahden kartan välisen pisteen y-koordinaattien välille. Ja taas x-akselilla liikuttaessa ei olisi väliä, paitsi tapauksessa; jossa haettavan pisteen x-koordinaatti olisi täysin sama kuin kahden polygonin välisten pisteiden väliset x-koordinaatit (ääritapaus).

Jos laskennan aloittaisi aina polygonin pienimmästä x-koordinaatin pisteestä, ja merkkaisi aina x-koordinaatin ylös kohdasta jossa haetun pisteen y-koordinaatti kuuluisi näiden kahden polygonikartan y-koordinaattien välille. Ja kun kaikki polygonin reunat olisi käyty läpi, voitaisiin suorittaa laskenta kuinka monta näistä ylös otetuista x-koordinaateista olisi x-akselilla ennen haetun pisteen koordinaattia. Huomattiin - ja tiedettiinkin jo - että jos tämä lukema olisi pariton, haettu piste olisi siis polygoni kartan sisällä. Parillisessa tapauksessa taas kartan ulkopuolella.

Näin saatiin paperilla piirtäen (hyvin suunnitteleamalla) purettua tulevan algoritmin rakenne, josta sitä on nyt paljon helpompi lähteä toteuttamaan. Seuraavaksi on siis tarkoitus paneutua algoritmin varsinaisen koodin vääntämiseen.