

Aiheenmäärittely

Labyrinth- ohjelman tarkoituksena on löytää nopein reitti labyrinthista ulos. Käyttäjä saa eteensä tyhjän labyrinthin eli "maa-alueen." Käyttäjä asettaa laudalle esteitä (seiniä), joiden läpi ei voi liikkua, luoden näin oman labyrinthin. Tämän jälkeen pelaajan tehtävänä on asettaa labyrinthin uloskäynti sekä lähtöpiste, josta ohjelma lähtee liikkeelle.

Ohjelman on tarkoitus etsiä näitä tietoja hyväksi käyttäen jokin nopeimmista reiteistä ulos labyrinthista. Kun ohjelma on löytänyt nopeimman reitin, antaa se käyttäjälle tiedon reitin käyttämisestä "maapaloista" värjäämällä kuljetun polun.

Algoritmina käytän A*- algoritmia, joka toimii seuraavasti:
Ohjelma arvottaa pelin "maapalat" etäisyyden mukaan eli asettaa niille heuristiset arvot. Tämän jälkeen ohjelma tarkistaa tarkasteltavan maapalan naapureista kaikista parhaimman eli valitsee naapurin, jolla on pienin heuristinen arvo. Algoritmi tarkistaa vain ylä- ja alapuolella olevat naapurit sekä oikealla ja vasemmalla puolella olevat naapurit. Labyrinthissa voi siis liikkua ainoastaan vertikaalisesti tai horisontaalisesti. Jokaisen maapalan kohdalla sen naapurit lisätään "avoimeen listaan" ja käsitelty maapala lisätään "suljettuun listaan." Näin ollen ohjelmalla on tieto kokoajan siitä, mitkä maapalat on jo tutkittu ja mitkä maapalat odottavat vielä vuoroaan. Näiden listojen avulla algoritmin tehokkuus saadaan $O(|E|)$ aikaan eli kaikki solmujenväliset siirtymät käsitellään korkeintaan kerran nopeimman reitin löytämiseksi.

Tietorakenteet:

Tietorakenteina käytän ohjelmassani taulukoita sekä linkitettyjä listoja, jotka sisältävät labyrinthin maapaloja. Näillä maapaloilla on tieto omista vanhemmistaan "parenteista", joita pitkin kulkien lopullinen reitti voidaan saada kätevästi esiin. Linkitetyt listat toteutan itse, eli toteutan listalle lisäämistoiminnon, listalta poistamistoiminnon, sekä listalta etsimistoiminnon.

Labyrinttina toimii siis käytännössä 2D- taulukko, joka sisältää verkon solmut (maapalat.)