

Määrittelydokumentti

Käytetyt algoritmit ja tietorakenteet

Tarkoitus on luoda algoritmi, joka pelaa vindinium peliä. Tähän tarkoitukseen tarvitaan useita asioita. Ensimmäinen on A* algoritmi, jolla etsitään lyhin reitti botin ja sen kohteen välillä. Sopiva heuristiikka on tärkeä, jotta botti ei kuluta suurta määrää aikaa miettien yhtä siirtoa. Samoin täytyy välttää tilannetta, jossa botti jää ikuisesti liikkumaan kahden pisteen välillä, koska etäisyys sopivaan kohteeseen vaihtuu. Näin voi tapahtua, koska pelissä on useampi liikkuva botti, joita voidaan ajatella esteinä. Jos A* toteutuu nopeasti ja helposti alan implementoimaan Fringe search versiota koodista, joka on nopeampi kuin A* ja tehokkaampi, mutta vaatii enemmän tilaa.

Lisäksi toteutetaan algoritmi, jolla botti osaa valita seuraavan liikkeen, ja siirtyä kohti kohdetta. Tämä toteutetaan vielä toistaiseksi päättämättömällä algoritmilla. Tämän avulla botti tarkastaa tämän hetkisen tilanteen, ja yrittää saavuttaa parhaan lopputuloksen sen hetkisestä tilanteesta.

Toteutettavat tietorakenteet

Tarpeellisia tietorakenteita ovat Map, List, Queue, Set. Nämä pyritään toteuttamaan siten, että ne ovat yksinkertaisia ja tehokkaita. Jos tarvetta tulee saatetaan toteuttaa muitakin tietorakenteita. Tietorakenteita ei käytetä suoraan A* algoritmilla, Setiä lukuun ottamatta, mutta niitä käytetään botin muissa osissa, kuten esim päätöksenteossa ja palvelimen lähettämän tiedon käsittelyssä. Mapia käytetään luomaan solmut, ja merkitsemään niiden heuristiikka A* varten.

Syöte

Syöteeksi ohjelma saa pelaajan oman avaimen, jolla pelaaja tunnistautuu peliin ja pelataanko harjoitus vai areena peli. Nämä ovat ainoat syötteet jotka ohjelma tarvitsee, ohjelman muu tarvittava informaatio saadaan vindinium palvelimelta. Tähän kuuluu pelikentän sisältö, sen koko, botin olin paikka ja tietoa botin tilasta pelissä. Näiden perusteella botti laskee siirtonsa ja lähettää tiedon palvelimelle.

Lopuksi botti ilmoittaa miten pärjäsi pelissä. Tämä saadaan tietää serveriltä, joka ilmoittaa kunkin osallistuneen botin tulokset pelin päätyttyä. Tämä informaatio sitten puretaan ja annetaan käyttäjälle helppolukuisessa muodossa.

Aika- ja tilavaativuus

A* aikavaativuus on riippuvainen käytetystä heuristiikkasta. Parhaimmillaan aika ja muistivaativuus on $O(n)$, mutta tällaisen heuristiikan löytäminen saattaa tuottaa vaikeuksia. A* on periaatteessa heuristiikan ohjaama Dijkstran algoritmi joten sillä on sama aika- ja tilavaativuus jos sitä käytetään ilman heuristiikkaa, eli aikavaativuus on $O((V+E) \cdot \log(V))$ ja tilavaativuus on $O(V^2)$. Tietorakenteissa tähdätään normaaleihin aikavaativuuksiin kaikissa operaatioissa.

Lähteet:

https://en.wikipedia.org/wiki/A*_search_algorithm

vindinium.org

Tietorakenteiden aikavaativuuksia

<http://bigocheatsheet.com/>