

Aineopintojen harjoitustyö: Algoritmit ja tekoäly viikkoraportti 2

Tietojenkäsittelytieteen kandidaattiohjelma (TKT)

1. Olen tällä viikolla pääosin ohjelmoinut Reversi-pelin rakenteen ja säänöt niin että se toimii moitteetta tekstipohjaisena pelattavana pelinä. Tässä sivussa olen myös kaiken aikaa suunnitellut työtä eteenpäin ja seuraavia vaiheita niin että sovelliukseen tämänhetkinen rakenne sopisi hyvin myös tuleviin vaiheisiin. Tämä on vaatinut myös jonkin verran aikaa ja perehtymistä niin testaukseen kuin algoritmeihinkin. Aikaa olen käyttänyt vaihtelevasti noin 1-5 tuntia aktiivista aikaa per päivä lauantaihin mennessä, yhteensä kuitenkin yli 12 tuntia tällä viikolla. Tämän lisäksi olen myös pohdiskellut aihetta ja suunnitellut työtä, vaikka en sitä laskekaan aktiiviseksi ajankäytöksi.
2. Reversin rakentaminen tekstipohjaiseen matriisiin vei hieman enemmän aikaa kuin olin ajatellut mutta mielestäni se on nyt tehty niin että se soveltuu hyvin seuraaviin vaiheisiin ja toimii vakaasti. Olen myös hahmotellut hieman jatkoa koodille tekoälyvastuksen suhteen. Tein ensin minimaxin joka optimoi yksinkertaisesti parasta pistemääriä ja joka otetaan käyttöön pelin loppuvaiheessa, mutta se on vielä hyvin raaka testiversio ja sitä voi tehostaa edelleen ja lisäksi sen toimintaa on tärkeä testata että se löytää parhaan ratkaisun (tässä yksi selkeä testiaihe joka täytyy jotenkin saada mukaan). Testaukseen liittyvä koodia ei vielä juuri ole mutta toivon pääseväni sen pariin ensi viikolla. Mutta toisaalta tämän hetken pelirunko laskee varmuuden vuoksi erilaisia pelituloja muutamalla eri tavalla tarkistuksena pelin moottorin sisällä, ehkä nämä voidaan erottaa omaksi testausosioksi myöhemmin.
3. Eniten ehkä olen oppinut uutta testauksesta, sillä olen lukenut kurssimateriaalia testaukseen liittyen ja pyrkinyt tekemään sellaisen ohjelman, että se toimii hyvin myös testauksen kannalta ja soveltuu testaukseen ilman muutoksia.
4. Olen miettinyt testausta noin yleisellä tasolla, että kuinka testata koodilla koodia. Tämän vaikuttaa jotenkin hankalalta konseptilta. Miten tietää, että testauksessa käytetty koodi on oikein, ellei sitä vuorostaan testata. Ehkä on mahdollista laskea joitain aina päteviä tarkistussummia, joita käyttää "aksioomina" ja muuten luottaa siihen, että kun tarpeeksi monipuolisesti testaa niin poikkeamat löytyvät jossain testissä koska eri tavalla laskettuna voidaan saada poikkeamia tuloksissa ja niitä voi tutkia sitten tarkemmin.

Toinen asia mitä mietin, että olisi kiehtovaa rakentaa aluksi yksinkertaisia algoritmejä jotka pelaavat reversiä toisiaan vastaan ja ajaa testipelejä sekä valita parhaat ominaisuudet jatkoon. Tämä olisi jopa hyvin nopeaa toteuttaa, mutta ilmeisesti pelkkä mahdollisimman vahva pelialgoritmi ei ole kurssin arvioinnin kannalta olennaisin asia, jos oikein olen ymmärtänyt kurssin kriteereitä.
5. Seuraavana jatkan tekoälyn kehitystä tehostamalla minimax-algorimiä loppupeliin ja lisäämällä hieman erilaisen heuristiikan, jota käytetään pelin aiemmissa vaiheissa. Lisäksi suunnittelen ja alan kehittelemään testausta.