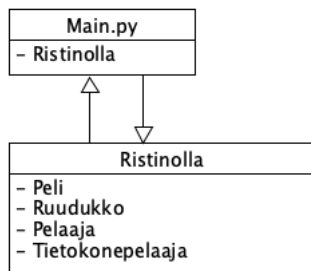


Toteutusdokumentti 27.11.2021

- Ohjelman yleisrakenne

Ohjelman suoritettava runko ja pääluokka (Ristinolla) löytyy tiedostosta main.py. Tämä luokka toteuttaa tavallaan rajapintaa pygamelle, jota peli hyödyntää ikkunan, käyttöliittymän ja ”peli-flown” toteutuksessa.



Varsinainen ristinollapeli on toteutettu paketissa Ristinolla, jonka luokat Peli ja Ruudukko toteuttaa pelilogiikan ja luokat Pelaaja ja Tietokonepelaaja sisältävät, nimiensä mukaisesti, pelaaja ja tietokonepelaaja -abstraktiot. Tietokonepelaaja-luokka vastaa varsinaisen tekoälyn ja minimax-algoritmin toteutuksesta.

Tietokonepelaajan tekoäly on toteutettu minimax-algoritmillä, joka käyttää alpha-beta -karsintaa. Alpha-beta -karsinnan tehostamiseksi siirrot järjestetään aina ennen niiden läpikäyntiä. Järjestämisen pohjana käytetään ”transposition-taulukkoa” ja heuristiikkaa joilla pyritään asettamaan läpikäytävät siirrot mahdollisimman parhaaseen järjestykseen niin, että algoritmi voisi karsia mahdollisimman paljon pelipuusta pois. Tällä saatiin hyvinkin paljon nopeutettua seuraavan siirron hakua varsinkin silloin kun siirroista on jo entuudestaan hyvää dataa saatavilla.

Transposition-taulukko on toteutettu niin, että se tallentaa pythonin ”sanakirjaan” (dictionary) merkintä-tyyppisen olion, joka indeksoidaan zonbrist-hashilla. Merkintä-objekti on toteutettu pitkälti Bruce Morelandin hash-rakenteen mukaisesti. Tosin hieman mukautettuna niin, että tietorakenteeseen tallennetaan tieto siitä onko kyseessä maksimoijan tai minimoijan tallentama pisteytys.

<https://web.archive.org/web/20070822204120/http://www.seanet.com/~brucemo/topics/hashing.htm>

https://en.wikipedia.org/wiki/Transposition_table

https://www.chessprogramming.org/Transposition_Table

- Saavutetut aika- ja tilavaativuudet (m.m. O-analyysit pseudokoodista)
- Työn mahdolliset puutteet ja parannusehdotukset
- Lähteet

Tavoitteena olevat aika- ja tilavaativuudet (m.m. O-analyysit)

Minimax-algoritmin aikatilavuus on $O(b^m)$ ja tilavaativuus $O(bm)$ missä b on sallittujen siirtojen määrä ja m on pelipuun maksimisyvyys.

Alpha-beta-karsinnan aikatilavuus voi käytännössä olla $O(b^{(2/d)})$ missä b on sallittujen siirtojen lukumäärä ja d on tietty syvyys jossa haku katkaistaan.

(Megalooikonomou, 200)

Lähteet

Vasileios Megalooikonomou, 2003, "The CIS603-Artificial Intelligence"-kurssimateriaali.

<https://cis.temple.edu/~vasilis/Courses/CIS603/Lectures/l7.html> (haettu 6.11.2021)