

Kivi-paperi-sakset tekoäly

Joonatan Vuorela

1. Määrittely

1.1 Toteutettavat algoritmit ja tietorakenteet

Ohjelmaan toteutetaan algoritmi pelissä tapahtuvien siirtojen tulkitsemiseen ja niissä havaittujen ”kaavojen” havaitsemiseen. Tekoälyn pitää algoritmin avulla huomata jos jokin tietty siirto toistuu jatkuvasti joidenkin tiettyjen tapahtumien jälkeen. Ei siis riitä että katsotaan vaan edellisiä siirtoja vaan myös kierroksen lopputulos, voitto tai häviö, pitää huomioida seuraavaa siirtoa valittaessa.

Siirrot on tarkoitus tallettaa round olioon joka taas talletetaan LinkedList tyyliin tietorakenteeseen jossa jokainen round olio tietää edellisen ja seuraavan olion. Näin voidaan toteuttaa tietorakenteeseen next ja prev metodit ja sen läpikäymisestä tulee helppoa. Toteutettu myös indexillä haku Esimerkiksi jos halutaan aloittaa selaaminen 50 siirtoa sitten tapahtuneesta round oliosta ja sitten edetä sieltä kohti tuoreinta oliota.

1.2 Ratkaistavat ongelmat ja valitut tietorakenteen

Oma tietorakenne on kaksisuuntainen linkitetty lista joka käyttää jokaisen olion next ja prev muuttujia seuraavaa oliota hakiessaan. Tietorakenteesta voi hakea indexillä ja poistaa myös indexistä jolloin tietorakenne osaa muuttaa indexit ja viittaukset oikeiksi.

1.3 Mitä syötteitä ohjelma saa ja miten niitä käytetään

Ohjelma saa syötteenä string-muuttujia. Pelissä käytetään vain käskyjä k,p,s joilla viitataan siirtoihin. Myös pelin lopettamiselle on myös käsky ”stop”

1.4 Tavoitteena olevat aika- ja tilavaativuudet

Aikavaativuus on tarkoitus olla $O(n)$. Round olioiden lukumäärä n on ratkaiseva tekijä niiden läpikäyntiä ajatellen. Algoritmin aikavaativuus ei koskaan ole eksponentiaalisesti kasvava.

Tilavaativuus olisi myös tällöin $O(n)$ koska alkioden määrä kasvaa ja niitä talletetaan kun uusi round olio luodaan. Periaatteessa aluksi on varattu muistia yhdelle oliolle ja sen jälkeen $n+1$:lle jossa n on olioiden lukumäärä

1.5 Lähteet

Tietoa haettu tira:n materiaalista ja ohjelmointiputkasta