

# **Testausraportti**

## **Ohjelman toimivuus:**

Pelin ydin ja siirtojen generointi on testattu kattavasti Junit-testauksella, jotka löytyvät testipakkauksesta. Lisäksi ohjelmaa on testattu jatkuvasti käyttäjätestauksella eli pelaamalla shakkia. Näin on löytynyt bugeja ja niitä on korjattu ja joitakin yksikkötestejä lisätty niiden perusteella. Esimerkki tällaisesta tilanteesta: Huomasin, että selvässä mattitilanteessa ohjelma väitti, että peli loppui pattiin. Lähdin debuggaamaan vikaa ja huomasin, että LoppuTarkistus tekee tyhjän siirron, jotta voisi vaihtaa vuoroa. Tämän piti olla ihan ok, mutta ShakkiPeli tyhjensikin siirron lähtöruudun vasta kun siirrettävä nappula oli asetettu kohderuutuun ja kuningas sattui olemaan juuri samassa ruudussa. Tällä hetkellä ohjelma näyttää toimivan kaikkien shakin sääntöjen mukaan. Käyttöliittymään tekemäni pelipuun tutkimis -ominaisuus helpottaa huomattavasti tekoälyn algoritmin testausta, koska siitä näkee, miten algoritmi on edennyt ja tunnistanut mm. laittomat siirrot ja lopputilanteet.

## **Tekoälyn shakkiosaaminen:**

Tekoäly pärjää tekijälleen vahvasti jo syvyydellä 5 voittaen useimmiten. Demotilaisuudessa tekoäly matitti testaamassa pelikaverini n. 15 siirrolla. Syvyydellä 6 en ole tainnut voittaa kertaakaan. Ei tämä kuitenkaan Kasparoville vielä pärjää. Valitettavasti en ole vielä saanut pelaamaan Timoa, joka oli alunperin tavoitteena voittaa. Tekoäly syvyydellä 5 pärjää hyvin Windows 7:n mukana tulevalle shakkipelille sen tasolla 6 ja molemmat miettivät suunnilleen yhtä kauan. Koska lisäsin arviointifunktioon satunnaismuuttujan, tekoälyä ei voi huijata tekemällä juuri tiettyjä samoja siirtoja joka pelissä.

## Alfabetta-karsinnan vertailu:

Testasin, millainen vaikutus alfa-beta-karsinnalla on minmax-algoritmiin.

Vertailin minmax-algoritmia ilman alfa-beta-karsintaa ja sen kanssa.

Seuraavassa on taulukot kahdesta eri pelitilanteesta. Allaoleva taulukko kertoo, kuinka monta siirtoa algoritmi kävi läpi, eli kuinka monta solmua pelipuussa on. Ennakkotieto osoittautui oikeaksi: Karsinta tehostuu huomattavasti, mitä syvemmälle haetaan. Taulukosta nähdään, että neljän siirron syvyydessä 90 % siirroista karsiutuu pois, mutta kaksi siirtoa syvemmällä jopa 99 % siirroista karsiutuu.

tilanne 1. keskipeli – enemmän nappuloita

Syvyys	MinMax	MinMaxAlfaBeta	suhde
3	36 700	15 187	40%
4	1 170 094	134 120	10%
5	38 751 994	2 101 964	5%
6	1 251 923 762	13 842 680	1%

tilanne 2. loppupeli – vähemmän nappuloita

Syvyys	MinMax	MinMaxAlfaBeta	suhde
3	4 388	634	15%
4	90 443	9 327	10%
5	1 560 944	33 952	2%
6	34 062 346	260 533	1%