

# Connect4

Dokumentaatiossa käytetty kieli: Suomi  
Opinto-ohjelma: tietojenkäsittelytieteen kandidaatti (TKT)  
Ohjelmointi kieli: Python  
Osatut kielet: Python

Mitä työ pitää sisällään:

Toimiva Connect4 peli ja tekoäly, jota vastaan voi pelata. Ohjelmaan annetaan syöte, mihin sarakkeeseen pelaaja haluaa tiputtaa nappulansa. Tekoälyn aikaja tilavaativuus ovat  $O(b^d)$  ja  $O(b \cdot d)$ . b on haarautumisluku ja tässä tapauksessa kuinka monta mahdollista liikettä per tila eli monta kolumnia pelissä on käytettävissä (maksimissaan 7). d taas on etsinnän syvyys, eli kuinka monta siirtoa eteenpäin halutaan laskea

Ydin:

Työn ydin on Connect4 peliä pelaava tekoäly, joka toteutetaan minimax-algoritmillä. Tekoälyn tulee toteuttaa kaikki tehtäväänannon vaatimukset ja lisävaatimukset. Tekoäly siis käy läpi mahdolliset siirrot ja määrittelee niille arvon sillä perustella, kuinka hyvä siirto on, ja toteuttaa parhaimman siirron vuoron lopussa.

Lähteet joita käytän:

Wikipedia:

[https://en.wikipedia.org/wiki/Alpha%E2%80%93beta\\_pruning](https://en.wikipedia.org/wiki/Alpha%E2%80%93beta_pruning)

[https://en.wikipedia.org/wiki/Minimax#Minimax\\_algorithm\\_with\\_alternate\\_moves](https://en.wikipedia.org/wiki/Minimax#Minimax_algorithm_with_alternate_moves)

Neverstopbuilding:

<https://www.neverstopbuilding.com/blog/minimax>

Stanford university:

<https://cs.stanford.edu/people/eroberts/courses/soco/projects/2003-04/intelligent-search/minimax.html>

<https://cs.stanford.edu/people/eroberts/courses/soco/projects/2003-04/intelligent-search/alphabeta.html>

Geeksforgeeks

<https://www.geeksforgeeks.org/dsa/minimax-algorithm-in-game-theory-set-1-introduction/>