

U programskom jeziku Python implementirati algoritme koji bi riješili sljedeće probleme:

- Raspoređivanje N skakača na šahovskoj tabli tako da se nijedan par međusobno ne napada
- Rješavanje sudoku slagalice
- https://en.wikipedia.org/wiki/Travelling_salesman_problem
- https://en.wikipedia.org/wiki/Water_pouring_puzzle

Razmotriti koliki su prostori pretrage za svaki od ovih problema, kao i koliki je prosječni faktor grananja. Koje od njih je moguće riješiti tradicionalnim algoritmima pretrage (DFS, BFS, A* i sl.), a kod kojih je potrebno vršiti lokalnu pretragu (hill climbing, simulated annealing i sl.) zbog prevelikog faktora grananja? Koji problemi se mogu formulirati u CSP obliku i riješiti backtracking pretragom uz upotrebu heuristika? U zavisnosti od odgovora na ova pitanja, za svaki problem implementirati odgovarajući algoritam (ili više njih ukoliko to ima smisla).