

## Prvi projektni zadatak – Dame

Algoritmi i strukture podataka 2023/2024

*Poslednja izmena 17.04.2024.*

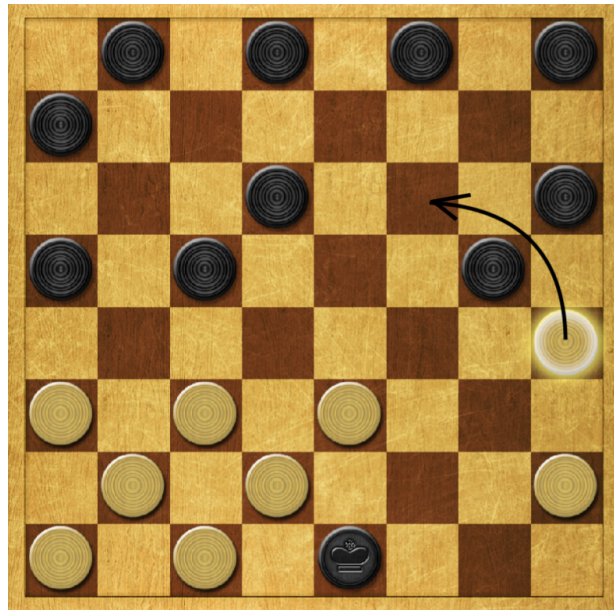
Napisati program za igranje igre dame (eng. checkers) gde bi program predstavljao jednog igrača. Ovu igru igraju 2 igrača (npr. A i B, beli i crni) sa po 12 figura na (šahovskoj) tabli kvadratnog oblika sa 8x8 polja.

**Faza 1 - postavljanje figura.** Na početku partije, figure se postavljaju kao na slici 1.



*Slika 1*

**Faza 2 - kretanje.** Igrači naizmenično igraju po jedan potez. Potez predstavlja pomeranje jedne figure na susedno slobodno mesto. Figure na tabli mogu da stoje samo na crnim poljima, a kretanje figura na tabli je moguće samo "ukoso", odnosno na druga crna polja i u smeru ka protivničkim figurama ("vraćanje" nije dozvoljeno). Cilj kretanja je da se kreira situacija kao na slici 2 kada je moguće preskočiti protivničku figuru posle čega se ona uklanja. U pravilima se javlja razlika da li je uklanjanje protivničke figure obavezno ukoliko se prilika pojavi ili igrač može da odluči da ne preskoči tj. "pojede" protivničku figuru (pomeranjem druge figure ili pomeranjem figure na neko drugo polje). Na početku igre je potrebno obezbediti odabir jednog od ova dva režima rada.



Slika 2

Kada figura jednog od igrača stigne do protivničke ivice, ona postaje "dama" čime dobija sposobnost kretanja unazad.

Igra se završava kada jedan od igrača ostane bez figura.

Javljaju se male varijacije u pravilima u različitim verzijama igre (u veličini table, broju figura i sl.) koje nije potrebno podržati.

Potrebno je implementirati prikazivanje trenutne table i odigranih poteza u konzoli (kroz pojednostavljeno iscrtavanje table).

Maksimalno vreme za odigravanje poteza za program je 5 sekundi.

## Bodovanje

Da bi se projekat smatrao odbranjenim, potrebno je modelovati stanje, obezbediti heuristiku i odigravati, odnosno odabrati sledeći potez na osnovu heuristike (dakle ne prvi pronađen, nasumično odabran i sl.). Najjednostavnija implementacija, dovoljna za prolaz, nosi 10 poena.

Za osvajanje preko 20 poena, računar mora da odigra svoj potez za manje od 3 sekunde. Primenjuje se skaliranje.

Da bi igranje bilo ugodno, potrebno je:

- Prikazati tablu u konzoli posle svakog odigranog poteza (2 poena)
- Ponuditi korisniku polja koja su mu dostupna za odigravanje sledećeg poteza (2 poena)
- Implementirati intuitivan interfejs (2 poena) koji obuhvata:
  - pregledno prikazivanje tabele

- **efikasan odabir ponuđenih poteza** (npr. odabir rednog broja ispred ponuđenog poteza)
  - označavanje potencijalnih, novih pozicija figure posle odigranog poteza
  - **jasno označavanje polja u tabeli**
  - Naglašavanje odigranog poteza protivnika
- Implementacija:
    - Heuristika: 5 poena
    - Varijabilna dubina: 2 poena
    - Stanje: 2 poena
    - 2 režima rada: 1 poen
    - Određivanje sledećeg stanja: 2 poena
    - Hash mapa za optimizaciju odigravanja: 4 poena
    - Minimax: 8 poena
    - Alfa beta rezovi: 5 poena
    - Višestruko preskakanje: 2 poena

#### Korisni linkovi

Online primer igre: <https://www.coolmathgames.com/0-checkers>  
<https://www.247checkers.com>

[https://en.wikipedia.org/wiki/Alpha-beta\\_pruning](https://en.wikipedia.org/wiki/Alpha-beta_pruning)  
<https://kartikkukreja.wordpress.com/2014/06/29/alphabetasearch/>  
<https://www.youtube.com/watch?v=xBXHz4Gbdo>  
<http://www.cs.huji.ac.il/~ai/projects/old/English-Draughts.pdf>