

SIMULATOR IGRICE „2048“

Smiljana Tedić i Maja Gulan

2048 je intrigantna i zarazna slagalica-igra za jednog igrača. Originalno je napisana u JavaScript programskom jeziku kao igra za igranje u web browser-u, ali danas se najčešće koristi na mobilnim uređajima. Princip igre nije revolucionaran jer je inspirisan mnogim drugim postojećim igrama kao što su 1024 ili Threes.

Kako se igra:

2048 se igra na kliznoj ploči koja predstavlja mrežu 4x4, stohastičke je prirode, a od igrača zahteva da razvije strategiju za maksimiziranje rezultata.

U svakom potezu igrač mora da izabere pravac levo, desno, gore ili dole. Potez je validan ako dovede do promena na mreži. Svakim validnim potezom na mreži dodaće se jedna nova pločica. Promene se javljaju spajanjem ćelija sa istim vrednostima. Kada igrač odabere pravac, položice u svim redovima i kolonama se poravnavaju (kližu) u tom pravcu. Spojene ćelije se ne mogu spojiti još jednom u istom potezu. Igra je moguća sve dok postoji najmanje jedna ćelija prazna ili bar dve iste koje mogu da se spoje. Sa druge strane, gotova je kada nema više validnih poteza.

Predmet rada:

Napravljen je konkurentni Java program za simulaciju i ručno igranje igrice.

Program se sastoji iz sledećih delova:

1. Dodavanje nove ćelije u ram:

- 1) prvo su zadate dimenzije svake pojedinačne slike: pozadine, brojeva, rama i izračunata je ukupna dimenzija na osnovu zadatah kao $SIZE = GRID_SIZE * BROJ_SIZE + RAM_DIMENZIJA * 2$;
- 2) pomoću rekurzivne funkcije određena je lokacija broja na slici.
- 3) po određivanju lokacije, kreirana je metoda koji vraća lokacije u vidu niza za unet broj. Korištene su originalne formule za određivanje reda i kolone koje predstavljaju elemente niza.
- 4) pomoću metode drawImage sa izvorišne lokacije se “precrtava” traženi broj na odredišnu.

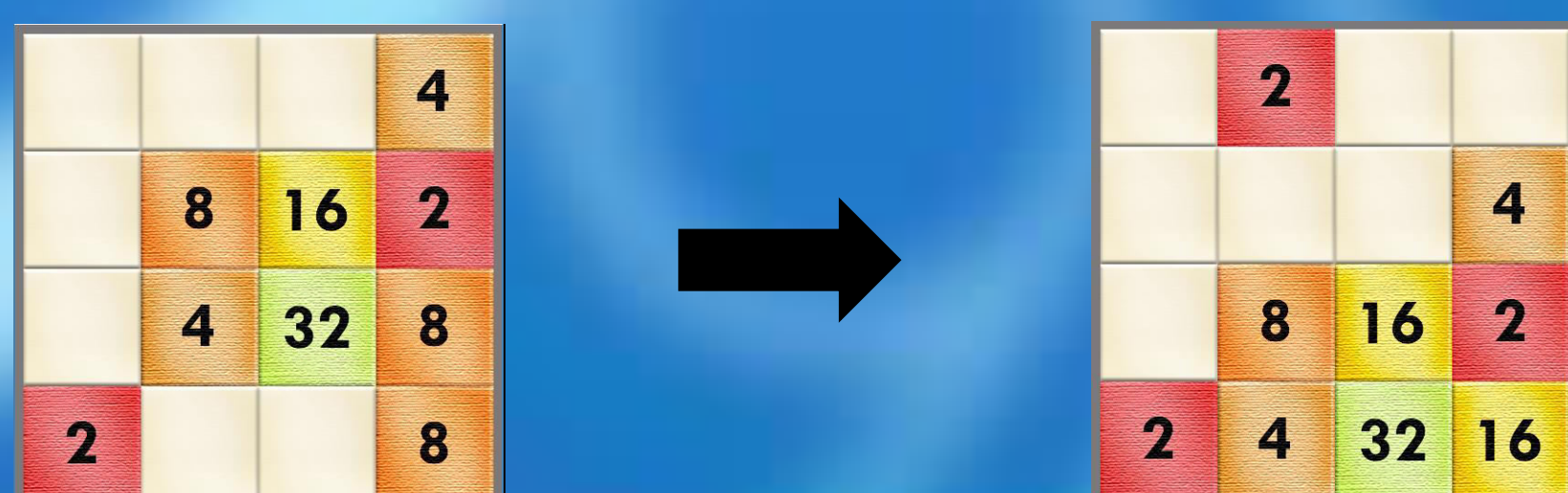
2. Pomeranje u pravcu uspeha → najboljeg rezultata:

Korištena je Monte Karlo metoda za simulaciju igrice. Monte Karlo metode su grupa računarskih algoritama koji se oslanjaju na ponavljanje slučajnih pokušaja da bi se dobili numerički rezultati. Imaju široku primenu u različitim naučnim disciplinama (matematici, računarskoj biologiji, grafici, fizici itd.). Ovaj metod se u računarskoj inteligenciji koristi za rešavanje igrica tako što pronalazi najbolji potez u igri pretražujući stanja. Za procenu dugoročnosti potencijala svakog poteza koristi se veliki broj slučajnih simulacija.

U implementaciji je kopirano trenutno stanje (napravljena je kopija mreže), zatim je učinjen korak (puštena je simulacija) i napravljen je još određen broj slučajnih okreta kako bi se procenilo koji potez daje najbolji rezultat. Na osnovu te procene, pomeranja se vrše u smeru najboljeg rezultata.

3. Spajanje i klizanje ćelija:

U mapi se čuvaju mogući smerovi kretanja i inicijalne pozicije brojeva. Kreira se kopija mreze nad kojom se vrši pomeranje i spajanje ćelija sa istim vrednostima, ukoliko one prethodno nisu bile spojene.

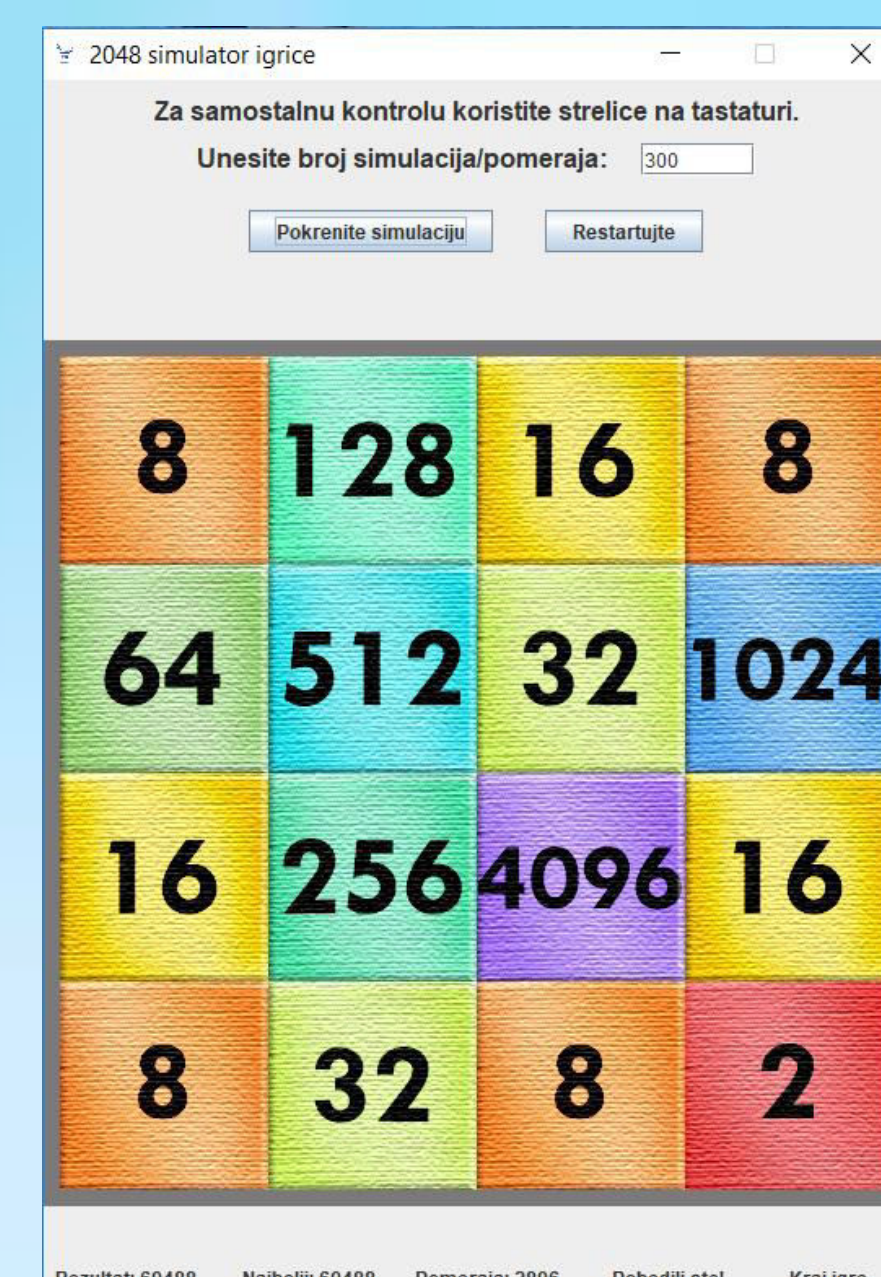


4. Određivanje slučajne (random) vrednosti dodate ćelije:

Na osnovu uslova da je verovatnoća za dodavanje 2 : 4 = 90% : 10%

5. Određivanje kraja igre:

Ako više nema mogućih smerova tj. ukoliko su kopija i trenutna mreža identične.



Prednosti Monte Karlo metode:

- Ne zahteva domensko znanje o igri da bi napravio logične poteze, stoga se može upotrebiti u velikom broju igara sa malo modifikacija.
- Može biti pokrenut bilo koji vremenski period i uvek će davati rešenje. Što je duže pokrenut rezultat će biti precizniji jer će se više puta izvršiti provera stanja.
- Ne zahteva heurističku evaluacionu funkciju, tako da je lak za implementaciju.

Nedostaci Monte Karlo metode:

- Kompromis je povećanje memorije i zahtevnosti procesora. Međutim, pošto potezi koji se koriste za procenu nastaju nasumice, moguće je da će potez koji bi bio odličan, osim za jedan specifičan odgovor protivnika, biti pogrešno ocenjen kao dobar potez. Rezultat toga su programi koji su jaki u celokupnom strateškom smislu, ali nisu savršeni taktički.
- Ovaj problem se može ublažiti dodavanjem domena znanja u generisanje poteza, kao i veći stepen dubine pretraživanja.

Poređenje parametara algoritma:

U zavisnosti od zadatog broja simulacija/ pomeraja igrica će se prelaziti efikasnije.

Metrika koja je uzeta za poređenje je ostvaren rezultat i vreme za koje ga je simulator dostigao.

U grafikonima ispod prikazani su rezultati:

