

# Objektno-orijentisano programiranje

## Drugi popravni kolokvijum

Vreme izrade: 120 minuta

### 1. Opis zadatka

Potrebno je implementirati Java aplikaciju koja simulira popularnu igru „Candy Crush“. Igra se odvija na tabli gde korisnik spajanjem istih simbola (slatkiša) sakuplja poene. Konačan izgled igre dat je na slici ispod.



Slika 1 Inicijalni prikaz igre

### 2. Specifikacija korisničkog interfejsa (GUI)

Aplikacija mora izgledati u skladu sa priloženim dizajnom (videti sliku iznad), uz poštovanje sledećih ograničenja:

- Glavni prozor: Naslov prozora je „Candy Crush“
- Tabla za igru:
  - Sastoji se od mreže dugmića dimenzije **8x8**.
  - Svako dugme (polje) ima dimenziju **70x70** piksela.
  - Na dugmićima se prikazuju simboli slatkiša (ukupno **6 različitih vrsta**).
- Ostalo:
  - Na dnu prozora nalazi se dugme za novu igru. Ovo dugme treba da bude **približne širine kao tabla za igru**, a visine **40 piksela**.
  - Na vrhu prozora nalazi se **broj poena igrača**, kao i **broj klikova** koje je igrač napravio na tabli.

### 3. Logika igre

Važe sledeća pravila igre:

- **Inicijalizacija:**
  - Prilikom pokretanja aplikacije ili klika na dugme „New Game“, tabla se popunjava **nasumičnim slatkišima**.
  - Algoritam generisanja početnog stanja **mora osigurati** da na početku ne postoji **3 ili više** istih slatkiša u nizu (po vrsti ili koloni). Svaka nova igra mora generisati drugačiji raspored.
- **Igranje:**
  - Igrač bira polja klikom miša.
  - Klikom na **dva različita polja**, proverava se da li su ona **susedna** (gledano po vrsti ili koloni, **ne dijagonalno**).
  - Ukoliko su polja susedna, slatkiši na tim pozicijama zamenjuju mesta.
  - Svaki klik na bilo koje polje uvećava brojač klikova.
- **Kraj igre:**
  - Igra se završava kada igrač sakupi **50 ili više poena**.
  - U tom trenutku, prikazati dijalog sa pitanjem „*Game Over! Your score: \_\_\_\_ . Do you want to start a new game?*“. Videti Slika 7.
  - Ako je odgovor **DA**, resetovati igru na početno stanje.
  - Ako je odgovor **NE**, zatvoriti aplikaciju.

### 4. Nivoi implementacije i uvećanje poena

Zadatak je podeljen na dva nivoa težine.

#### **Verzija A: Lakša verzija zadatka** – *video\_laksi\_primer.mp4*

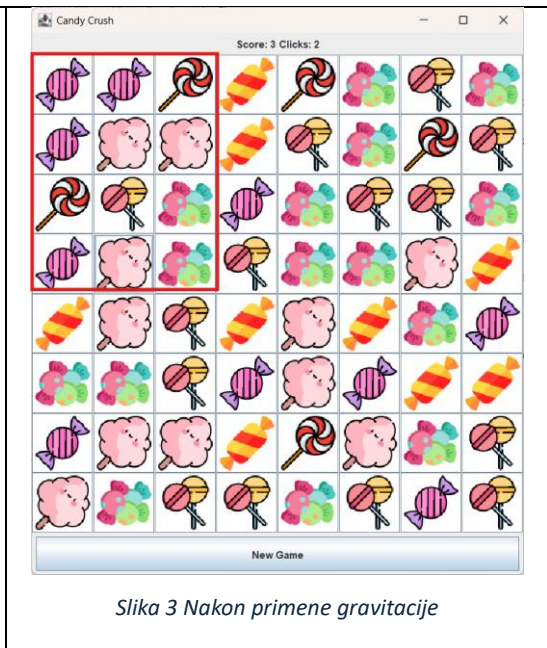
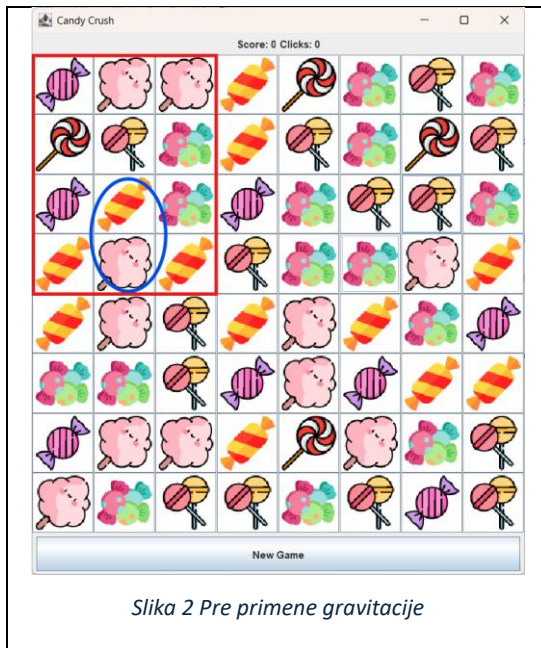
Ova verzija podrazumeva statičko uklanjanje slatkiša.

- Nakon što igrač zamenom mesta formira niz od 3 ili više slatkiša (horizontalno ili vertikalno):
  - Svi slatkiši koji čine taj niz se uklanjaju sa table.
  - Polja sa kojih su uklonjeni slatkiši ostaju prazna (nema slike slatkiša). Videti Slika 6.
  - Na tabli se ne generišu novi slatkiši na mestu praznih polja.

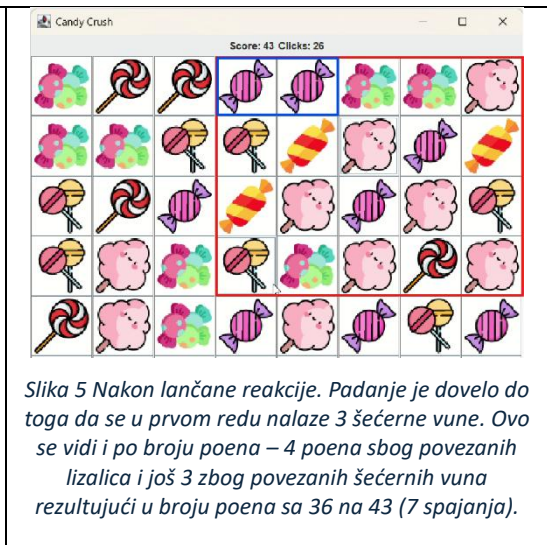
#### **Verzija B: Teža verzija** – *video\_tezi\_primer.mp4*

Ova verzija podrazumeva dinamičko pomeranje i implementaciju „gravitacije“.

- Nakon što igrač zamenom mesta formira niz od 3 ili više slatkiša (horizontalno ili vertikalno):
  - Svi slatkiši koji čine taj niz se uklanjaju sa table.
  - **Efekat gravitacije:** Svi slatkiši koji su se nalazili iznad uklonjenih polja padaju naniže kako bi popunili praznine. Efekat gravitacije vizualizovan je na slikama u tabeli ispod.

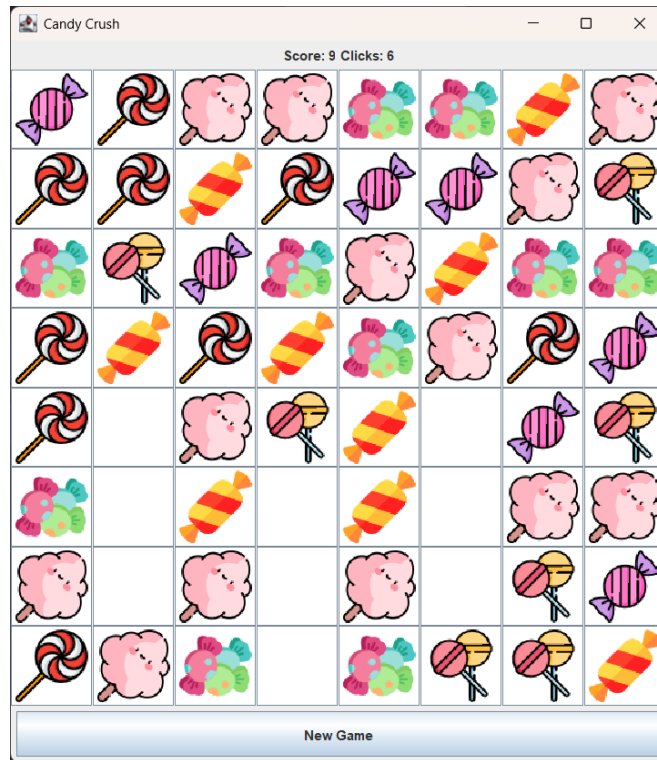


- **Dopuna table:** Na prazna mesta koja su ostala pri vrhu kolone generišu se novi, nasumični slatkiši.
- **Lančana reakcija:** Ukoliko novo stanje na tabli (nakon pada i dopune) formira nove nizove od 3+ ista slatkiša, proces se automatski ponavlja (uklanjanje -> padanje -> dopuna) sve dok na tabli više ne bude mogućih spajanja.

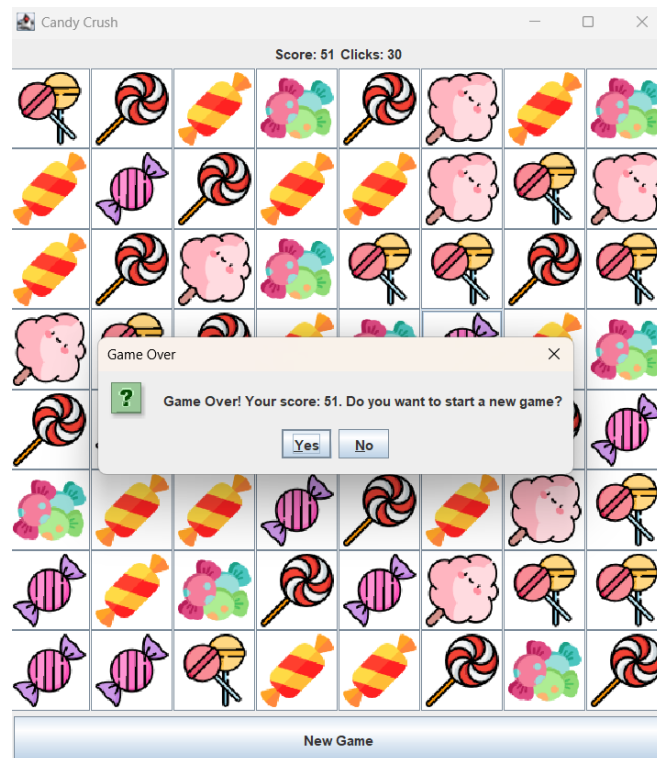


Svako uklonjeno polje (uključujući i ona iz lančanih reakcija kod teže verzije zadatka) donose **1 poen** igraču u igri.

Dodatni resursi (Screenshots):



Slika 6 Prazna polja nakon uklanjanja u lakšoj verziji zadatka



Slika 7 Obrada kraja igre