

**SVEUČILIŠTE U RIJECI  
ODJEL ZA INFORMATIKU**

**Preddiplomski studij informatike**

**Završni rad iz kolegija  
OBJEKTNO PROGRAMIRANJE**

**FROGGER PROGRAMIRAN NA  
OBJEKTNO ORIJENTIRAN NAČIN**

**Autor:**      **Bojan Radulović**

**Mentori:**    doc. dr. sc. Miran Pobar  
                 Milan Petrović, mag. inf.

U Rijeci, veljača 2022.

## Sadržaj

1. Uvod .....	3
2. Opći pregled programa .....	4
3. Opis strukture programa .....	6
3.1. Korištene biblioteke .....	6
3.2. Klase .....	7
4. Upute za buildanje i korištenje programa .....	10
4.1. Buildanje programa .....	10
4.2. Korištenje programa .....	10
5. Zaključak .....	12
6. Izvori .....	13

## 1. Uvod

“Frogger” akcijska je arkadna igra koju je 1981. godine razvila Japanska tvrtka Konami u kojoj igrač upravlja žabom koja na siguran način mora doći od dna do vrha ekrana, pritom izbjegavajući jureće aute i utapanje u rijeci skačući po plutajućim balvanima.

Cilj ovog završnog rada rekreirati je funkcionalnosti ove igre koristeći pritom koncepte objektno orijentiranog programiranja u programskom jeziku C++.

## 2. Opći pregled programa

Prilikom pokretanja aplikacije korisniku se prikazuje glavni izbornik programa. Pri vrhu ekrana na sredini nalazi se naslov “Frogger” te moje ime kao stvaratelja aplikacije, a ispod toga nalaze se gumbi za pokretanje igre i izlaz iz programa. Na samom dnu ekrana u lijevom kutu nalaze se podatci o zadnjem i najvećem postignutom rezultatu koji se iščitavaju iz vanjske tekstualne datoteke “FroggerBodovi.txt” (ako ona postoji, a inače se vrijednosti postavljaju na nulu).



Slika 1, Glavni izbornik programa

Klikom na gumb “Izlaz”, pritiskom tipke Esc ili pritiskom slova “i” na tipkovnici, program se zatvara.

Klikom na gumb Kreni, pritiskom tipke Enter ili pritiskom slova “k” na tipkovnici igra se pokreće. Na dnu ekranu se u lijevom kutu prikazuju bodovi stečeni tijekom aktualne igre, a u desnom žabe čiji broj predstavlja preostale živote igrača (na početku igre taj broj je tri).

Iznad toga nalazi se žaba kojom igrač može upravljati pritiskom tipaka strelica na tipkovnici. Žaba se nalazi na nogostupu – sivoj horizontalnoj traci na kojoj je igrač siguran od svih opasnosti.

Iznad nogostupa nalaze se tri crne trake koje predstavljaju cestu. Po svakoj od tih traka kreću se po dva auta, različitih boja, različitim brzinama i smjerovima. Nakon ceste slijedi još jedan nogostup.

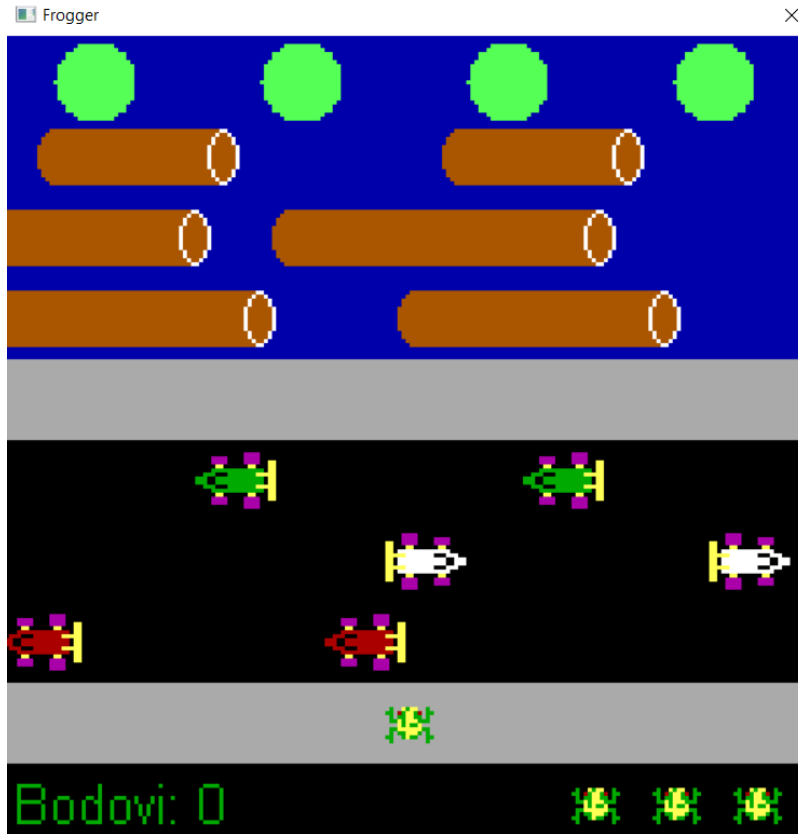
Nakon drugog nogostupa slijede četiri plave trake koje predstavljaju rijeku. Po svakoj od prvih tri od tih traka kreću se balvani, različitih dužina, različitim brzinama i smjerovima. Ako igrač dodiruje balvan, kreće se zajedno s njim istom brzinom i u istom smjeru. U zadnjoj traci pri samom vrhu ekrana nalaze se četiri zelena kruga koji predstavljaju lopoče.

Kada igrač dotakne lopoč, on nestaje, igrač se vraća na početnu poziciju i pridodaje se jedan bod. Cilj igrača doći je do svakog od tih četiriju lopoča, nakon čega se igra resetira (osim bodova i života igrača) s time da se auti i balvani kreću brže.

Ako igrač dotakne auto, upadne u rijeku (nalazi se na plavim trakama, a pritom nije na balvanu ili lopoču) ili ga odnese rijeka (igrač nestane s ekrana), igrač se vraća na početnu poziciju te se uklanja jedan život. Kada broj života dosegne nulu, postignuti rezultat se zapisuje u vanjsku tekstualnu datoteku "FroggerBodovi.txt" te se korisnik vraća na glavni izbornik.

Također, korisnik se tijekom igre u bilo kojem trenutku može vratiti na glavni izbornik pritiskom tipke Esc na tipkovnici.

Na početku svake nove igre te nakon sakupljanja sva četiri lopoča, pozicije autiju i balvana postavljaju se na nasumične vrijednosti.



Slika 2, Ekran neposrednu nakon pokretanja nove igre

### 3. Opis strukture programa

#### 3.1. Korištene biblioteke

Za ispravan rad programa potrebno je nekoliko standardnih biblioteka:

- Biblioteke `<iostream>` i `<fstream>` korištene su za zapis i čitanje podataka iz vanjske tekstualne datoteke
- Biblioteka `<vector>` korištena je za kontejnere dinamičkih veličina
- Biblioteka `<string>` korištena je za lakšu upotrebu nizova znakova
- Biblioteka `<chrono>` korištena je za mjerenje vremena potrebnog za izvršavanje glavne petlje pokrenute igre kako bi se ta vrijednost mogla oduzeti od vremena čekanja programa između iteracija petlje čime se postiže fiksna stopa crtanja na različitim računalima
- Biblioteke `<ctime>` i `<cstdlib>` korištene su za generiranje nasumičnih pozicija autiju i balvana na početku svake nove igre

Od ne standardnih biblioteka korištena je isključivo biblioteka “immediate2d”, vrlo jednostavna biblioteka za grafiku (i zvuk) koja se sastoji od samo jedne header datoteke.

### 3.2. Klase

Programski kod aplikacije podijeljen je u sedam klasa.

Klasa “Igra” sadrži privatne podatke o težini igre te preostalim životima i rezultatima igrača i metode potrebne za prikazivanje različitih dijelova igre i izbornika, periodično pomicanje autiju i balvana tijekom igre, uklanjanje života, učitavanje i zapisivanje rezultata igre u vanjsku datoteku, pokretanje igre itd. Korištenje nekih od ovih metoda ima smisla samo prilikom trajanja igre te su zato privatne (pomicanje autiju i balvana, uklanjanje života...) dok se druge mogu ispravno koristiti u bilo kojem trenutku te su zato javne (pokretanje igre, otvaranje glavnog izbornika...).

Klasa “ObjektIgre” temeljna je klasa ovog programa te sadrži zaštićene atribute o poziciji objekta (instance klase) te njegovim dimenzijama. Sve metode ove klase su javne te služe za postavljanje i dobivanje vrijednosti atributa objekta te detektiranje sudara (dodirivanja) među objektima i provjeru nalazi li se objekt unutar dimenzija ekrana.

Pošto je biblioteka “immediate2d” koja je korištena za grafiku vrlo jednostavna, za crtanje kompleksnijih oblika poput autiju, balvana i žaba korišten je vector string-ova “Nacrt” čiji znakovi predstavljaju boje piksela koji se trebaju nacrtati na ekran (npr. znak ‘g’ označava piksel zelene boje, ‘y’ žute, ‘ ’ označava da na toj poziciji nije potrebno nacrtati piksel...).

```
{ " g  ygyy  g ",  
  "gg rgyygr gg",  
  " g ggyygg g ",  
  " ggygyyyggg ",  
  "   ygyggy   ",  
  " ggyyyyyyygg ",  
  " g ggyyyg  g ",  
  "gg  gygg  gg",  
  " g          g "}
```

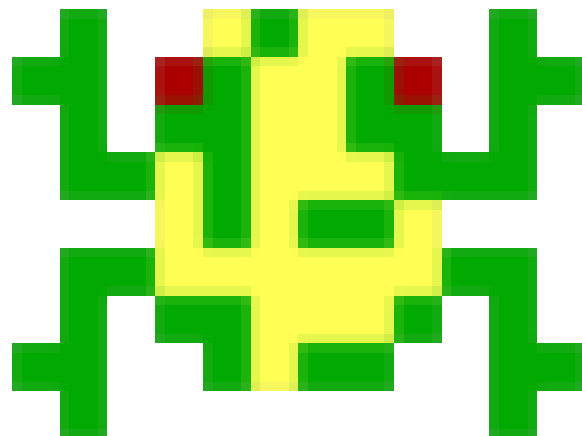


Figure 3, Primjer koji prikazuje nacrt žabe te žabu onakvu kakva će se prikazati na ekranu

Iako bi se svi objekti mogli crtati pomoću ove metode, lopoče, koji imaju oblik običnog kruga, i tekst gumbiju lakše je crtati pomoću gotovih funkcija iz biblioteke “immediate2d”. Stoga, klasa `ObjektIgre` ima tri podklase: `ObjektSaNacrtom`, `Lopoc` i `Gumb`.

Klasa “`Gumb`” podklasa je klase “`ObjektIgre`” te ima samo jedan atribut; privatni atribut “`Tekst`” koji označava tekst gumba. Ova klasa također ima dvije javne metode za crtanje gumba i provjeru je li gumb kliknut.

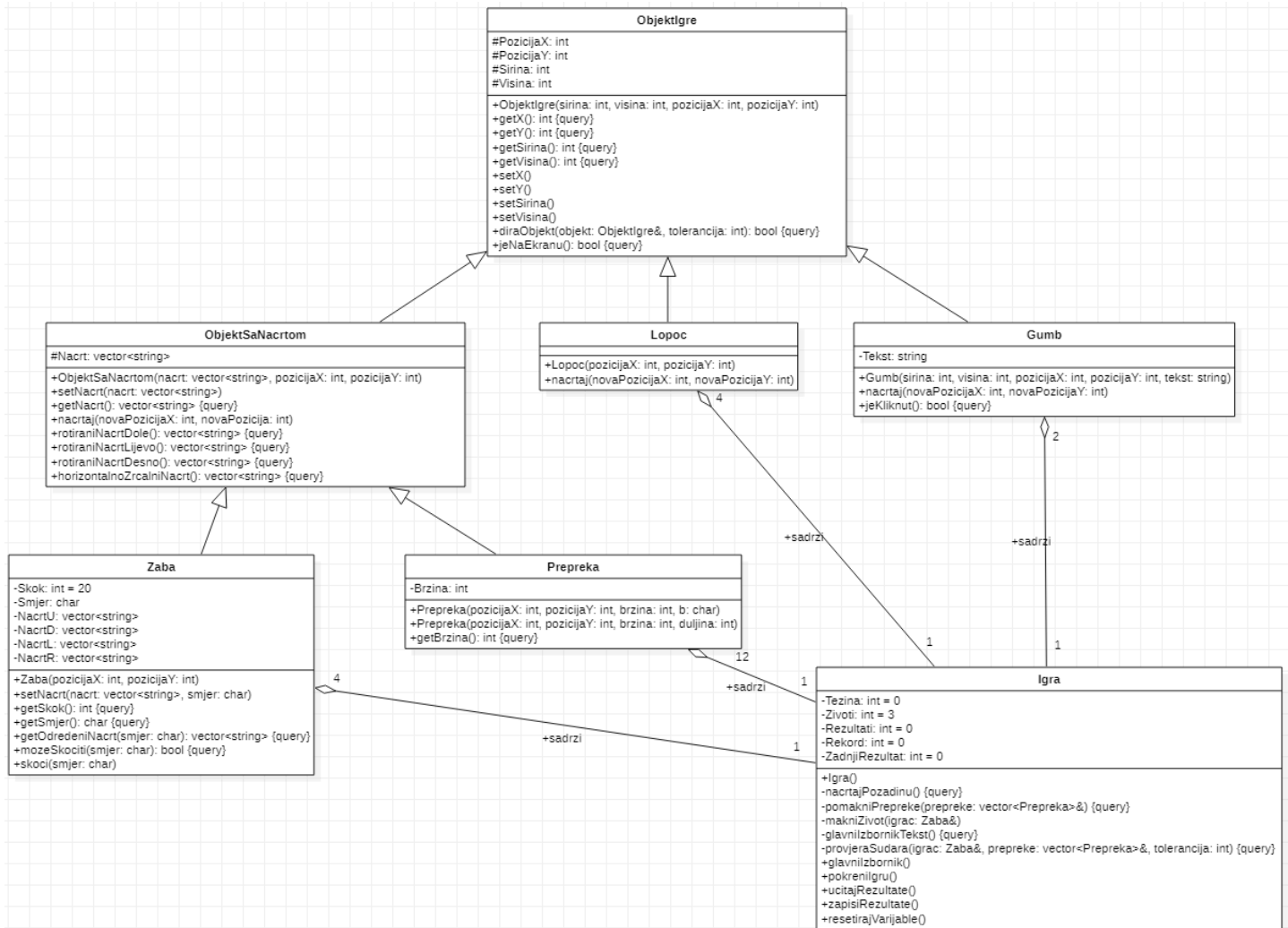
Klasa “`Lopoc`” podklasa je klase “`ObjektIgre`”. Ovo je vrlo jednostavna klasa koja sadrži samo jednu javnu metodu čija je zadaća crtanje lopoča.

Klasa “`ObjektSaNacrtom`” također je podklasa klase “`ObjektIgre`”. Sadrži nacrt objekta kao zaštićen atribut, javne metode za postavljanje i dobivanje nacrt, crtanje objekta te nekoliko metoda koje vraćaju kopiju nacrt manipuliranu na različite načine (horizontalno zrcaljenu ili rotiranu za devedeset, sto osamdeset ili dvjesto sedamdeset stupnjeva)

Klasa “`Prepreka`” posljednja je podklasa klase “`ObjektSaNacrtom`”. Ova klasa sadrži privatni atribut koji predstavlja brzinu prepreke te javnu metodu za pristup tom atributu. I auti i balvani instance su ove klase, pošto imaju identično ponašanje (kreću se određenom brzinom te se nešto događa ako ih igrač dodiruje/ ne dodiruje). Ova klasa ima dva inicijalizatora. Razlika među njima je da jedan prima znak koji predstavlja boju auta, a drugi broj (integer) koji predstavlja duljinu tijela balvana. Ako pri stvaranju instance ove klase koristimo prvi inicijalizator, instance poprima izgled auta određene boje, a ako koristimo drugi instance će imati izgled balvana određene duljine.

Posljednja klasa je klasa “`Zaba`”. Ona sadrži privatne attribute za duljinu skoka žabe, smjer u kojemu je žaba okrenuta te nacrt za svaku od četiri moguće orijentacije žabe. Također sadrži javne metode za dobivanje nekog od nacrt žabe, dobivanje smjera u kojem je okrenuta žaba, dobivanje duljine skoka žabe, postavljanje nacrt, skok žabe u određenom smjeru za duljinu definiranu vrijednošću atributa skoka, te provjeru ima li žaba dovoljno mjesta za skok u određenom smjeru.





Slika 4, Dijagram klasa programa

Svaka instanca klase “Igra” u sebi sadrži četiri instance klase “Zaba” (jedan igrač i tri ikone), dvanaest instanci klase “Prepreka” (šest autiju i šest balvana), četiri objekta klase “Lopoc” te dvije instance klase “Gumb” (gumb za pokretanje igre i izlaz iz programa).

## 4. Upute za buildanje i korištenje programa

### 4.1. Buildanje programa

Program je napravljen za operacijski sustav Windows te je uspješno testiran na Windows verzijama 7 i 10.

Korišteno integrirano razvojno okruženje je Microsoft Visual Studio Community 2019 verzija 16.11.5.

Pošto program koristi samo jednu nestandardnu biblioteku, koja se sastoji od samo jedne header datoteke, buildanje programa je izuzetno jednostavno te ne zahtijeva nikakvu instalaciju. Dovoljno je otvoriti Visual Studio soluciju programa i pokrenuti debugging, nakon čega se u direktoriju projekta stvara mapa “Debug” u kojoj se nalazi .exe datoteka programa pod nazivom “Frogger.exe”.

Nadalje, datoteka “Frogger.exe” priložena uz završni rad buildana je s Runtime bibliotekom “Multi-threaded Debug (/MTd)” te se može pokrenuti na bilo kojem Windows računalu bez potrebe za ijednom drugom datotekom.

### 4.2. Korištenje programa

Program je vrlo jednostavan za koristiti. Nakon pokretanja programa otvara se glavni izbornik programa. Ovdje korisnik ima opciju zatvoriti program klikom na gumb “Izlaz”, pritiskom tipke Esc ili pritiskom slova “i” na tipkovnici ili pokrenuti novu igru klikom na gumb “Kreni”, pritiskom tipke Enter ili pritiskom slova “k” na tipkovnici.

Nakon pokretanja igre korisnik može koristiti tipke strelica na tipkovnici za pokretanje žabe u određenom smjeru.

Na dnu ekrana može se pratiti broj osvojenih bodova i preostalih života (pomoću broja prikazanih žaba).

Cilj igre prijeći je cestu bez da žabu dotakne auto, prijeći rijeku skačući od balvana do balvana te doći do svakog od lopoča na vrhu ekrana. Svaki put kada igrač dođe do lopoča, dodjeljuje mu se bod. Kada igrač dođe do sva četiri lopoča, cilj je ponoviti isto kako bi se skupilo što više bodova, ali ovaj put će se auti i balvani kretati nešto brže.

Svaki put kada igrač dotakne auto, upadne u rijeku ili nestane s ekrana (u slučaju da ga odnese rijeka) gubi jedan život. Igra završava kada igrač izgubi sva tri života te se tada ponovno otvara

glavni izbornik. Također, igrač u svakom trenutku tijekom igre ima opciju vratiti se na glavni izbornik pritiskom tipke Esc.

## 5. Zaključak

Objektnim programiranjem moguće je napraviti rekreaciju klasične arkadne igre “Frogger” tvrtke Konami u programskom jeziku C++. Objektno orijentiran pristup posebno je prikladan za izradu ovog projekta zbog većeg broja objekata igre sličnih osobina (npr. svi imaju neku poziciju i dimenzije) koji međusobno interagiraju (npr. žaba može stajati na balvanu, može ju udariti auto itd.).

Objedinjenjem zajedničkih struktura i ponašanja, procesom generalizacije, moguće je na jednostavan i intuitivan način stvoriti nadklase objekta, dok se procesom specijalizacije na jednako intuitivan način mogu stvoriti podklase s njima specifičnim metodama (npr. skakanje žabe) i atributima (npr. brzina prepreka, duljina skoka žabe).

## 6. Izvori

Malik, D.S., C++ Programming From Problem Analysis to Program Design, 6th edDatoteka

Rao S., Liberty J. Jones B., Sams teach yourself C++ in one hour a day, 6th edDatoteka

Sams Teach Yourself UML in 24 Hours, Third Edition By: Joseph Schmuller Publisher: Sams PublishingDatoteka

The Unified Modeling Language User Guide By: Booch G., Rumbaugh J., Jacobson I. Publisher: Addison WesleyDatoteka

Piegdon N., immediate2D QuickReference

<https://github.com/npiegdon/immediate2d/blob/master/QuickReference.pdf> (Zadnje izmijenjeno: 23.11.2017., Pristupljeno: 18.2.2022)