**Projektni zadatak**

**CS367 - Razvoj igara 1**

***Super Pang***

***remake***

**Milan Radeljić 2865**

**Profesor: Miljan Milošević**

**Beograd, 2017 – 2018**

Sadržaj

[1. Uvod 2](#_Toc522075473)

[1.1 Postavka zadatka 2](#_Toc522075474)

[1.2 Gameplay 2](#_Toc522075475)

[2. Metodologija 2](#_Toc522075476)

[2.1 Teoretske osnove 2](#_Toc522075477)

[2.1.1 OpenGL i GLUT 3](#_Toc522075478)

[2.1.2 Prikazivanje slika 3](#_Toc522075479)

[2.2 Arhitektura 3](#_Toc522075480)

[2.2.1 Korišćeni alati i organizacija datoteka 3](#_Toc522075481)

[2.3 Petlja igre 4](#_Toc522075482)

[2.4 GUI 5](#_Toc522075483)

[2.5 Iscrtavanje karaktera 5](#_Toc522075484)

[2.6 Demo 6](#_Toc522075485)

[3. Diskusija i zaključak 9](#_Toc522075486)

[4. Literatura 9](#_Toc522075487)

1. Uvod

1.1 Postavka zadatka  
Za predmet CS367 Razvoj igara 1, projektni zadatak podrazumeva jednostavnu video igru u dvodimenzionalnom prostoru. Igra se razvija u programskom jeziku C/C++ i za grafički prikaz mora da koristi aplikacioni programski interfejs OpenGL i biblioteku GLUT. Igra mora biti sinhronizovana u realnom vremenu. Potrebno je implementirati pristupe razvoja igre koji se uče u okviru predmeta da bi se napravila kompletna igra.

## 1.2 Gameplay

Super pang je shooter igra nastala devedesetih od strane Capcom korporacije.  
Postoje dva moda igre u originalnoj verziji:

1. Kampanja
2. Survival

Za projekat je urađen remake survival nivoa, dakle poenta igre je izdržati što duže i napraviti što veći rezultat. Igrač kontroliše svog karaktera tako što se pomera levo/desno koristeći strelice na tastaturi i pucajući “strele“ pritiskom space tastera. Strele nisu strele u klasičnom smislu te reči, gde imaju svoje fiksne dimenzije i lete kroz prostor. U ovoj igri strela se stvara na igračevom mestu i raste vertikalno. Tokom igre će se pojavljivati kugle (baloni) u nasumičnom intervalu. Kugle se kreću tako što odskaču od poda do određene visine, a ujedno se kreću i levo/desno, dok ne udare ivice prozora, tada im se horizontalni smer menja. Postoje 4 vrste kugli. Razlikuju se u svojoj veličini i visini odskakanja od poda.  
Kada igrač ispali strelu ona će leteti dok se ne sudari sa kuglom ili sa vrhom prozora. Igrač je u stanju da se pomera nakon što je strela ispaljena, a ona će nastaviti da se raste od svoje početne pozicije. Na ekranu je moguće imati maksimalno 2 strele u isto vreme. Ako se pritisne space dok su obe strele u letu, izbrisaće se strela koja je bila ispaljena prva. Kada strela pogodi kuglu, kugla puca i stvara dve manje, proces se ponavlja sve dok se ne unište kugle najmanje veličine, one ne stvaraju ništa iza sebe. Dakle da bi se uništila najveća kugla, moraće se uništiti još 3 serije narednih manjih parova tj. 23= 8.  
Ako se kugla sudari sa likom igrača, igrač gubi život i nivo se resetuje, naravno poeni i dalje ostaju isti. Kraj igre je kada igrač izgubi sve živote.

2. Metodologija

## 2.1 Teoretske osnove

Da bi se izradila ovakva igra, bilo je neophodno razumeti osnovne koncepte koji pokreću sve računarske igre: kako napraviti petlju igre, svet igre koji se u redovnim intervalima ažurira, i kako prikazati taj svet na ekran u vidu animacije.

### **2.1.1 OpenGL i GLUT**

Grafički sistem korišćen u ovom projektu je OpenGL. Iako se često zove biblioteka, OpenGL je zapravo aplikacioni programski interfejs koji omogućava komunikaciju sa grafičkim procesorom i čija forma je standardizovana među proizvođačima grafičkih kartica. Razlika nije toliko ni bitna, i OpenGL može da se smatra bibliotekom i praktičnom smislu, jer pruža rutine pomoću kojih programer može da diktira šta će biti prikazano na ekranu. Jedino bude očigledno da zaista nije biblioteka kada se dovede do ponašanja koja nisu definisana standardom i rezultati budu drugačiji u zavisnosti od implementacije proizvođača.

### **2.1.2 Prikazivanje slika**

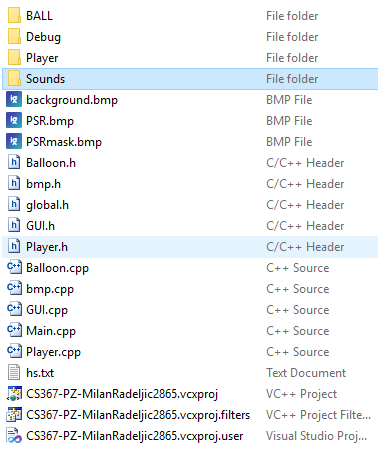
OpenGL nudi mogućnost prikazivanja slika u formi bitmapa. Bitmapa je određeni tip datoteke koji u sebi sadrži informacije o pikselima. Za projekat je korišćen format BMP, koji se kolokvijalno zove bitmapa, iako je specifičan format koji prati određeni standard. BMP format diktira određenu strukturu podataka koja se upisuje u datoteku, uključujući meta informacije o veličini slike, paleti boja, kao i enkodirani pikseli slike. Kada se učitaju podaci o pikselima u memoriju, mogu da se proslede OpenGL-u za prikazivanje. OpenGL ima više funkcija koja to mogu da izvrše, ili direktno ili preko nekog oblika u vidu teksture. Funkcije glDrawPixels i glBitmap direktno iscrtavaju sadržaj piksela na ekran. U projektu je korišćena funkcija glDrawPixels za iscrtavanje sprajtova.

2.2 Arhitektura  
Kao što je već navedeno, igra je implementirana u programskom jeziku C++. Za grafiku je korišćen aplikacioni programski interfejs OpenGL i biblioteka GLUT za rad sa korisničkim ulazom i servisima operativnog sistema, kao što je otvaranje prozora. U ovoj sekciji će biti detaljno opisan postupak po kojem su implementirani zahtevi projektnog zadatka.

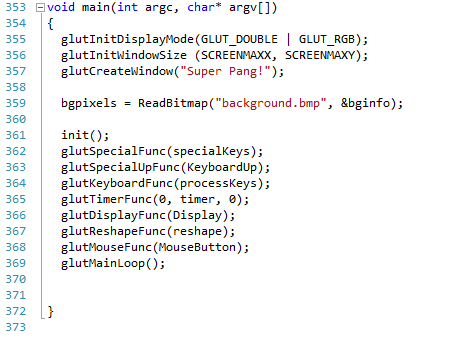
### 2.2.1 Korišćeni alati i organizacija datoteka

Korišćeno je više alata u izradi i testiranju projekta. Za pisanje i izvršavanje koda je korišćeno razvojno okruženje Visual Studio 2010 na Windows 10 operativnom sistemu. Za obradu sprajtova korišćen je Photoshop. Za isecanje zvukova korišćen je Audacity.

Struktura datoteka u projektu je prikazana na narednoj slici. Datoteka Main.cpp je ulazna tačka za celu igru i sadrži main funkciju koja se poziva kad se program pokrene. U sekciji 2.3 će se detaljno proći kroz ovu datoteku jer je ona odgovorna za postavljanje petlje igre. Direktorijumi BALL i Player sadrže slike koje su potrebne za iscrtavanje kugli i igrača tokom kretanja. Background.bmp je slika pozadine, a PSR.bmp predstavlja sliku kada se igrač ne kreće ili dok puca. Sounds direktorijum sadrži zvukove u .wav formatu. Postoje zvuci za pucanje kugle, ispaljivanje strele, sudar kugle sa igračem i zvuk za Game Over. High Score se čuva u hs.txt datoteci.



## 2.3 Petlja igre

Kod zadužen za postavljanje i izvršavanje petlje igre se nalazi u Main.cpp datoteci. Na slici se može videti sažeta verzija ove datoteke i funkcije koje su bitne za petlju igre.

Funkcija main prvo pozove GLUT funkcije koje su potrebne za otvaranje prozora i postavljanja callback funkcija. Funkciji glutDisplayFunc se prosledi funkcija Display, što znači da je ova funkcija dužna da prikaže sav sadržaj na ekran. Funkciji glutTimerFunc se prosleđuje funkcija timer, u intervalu od 30ms. Zatim se poziva funkcija glutMainLoop koja otvara prozor, inicijalizuje igru, i prekida ju kada se prozor zatvori. Ova funkcija služi kao petlja igre. Timer funkcija ima veoma jednostavno telo, proverava da li je trenutno pauzirana igra. Ako nije omogućava stvaranje kugli i ažurira strelu. Takođe dodaje 30 ms promenljivoj pod nazivom msec koja služi za prikaz koliko je igrač sekundi živ i zatim rekurzivno postavi sebe kao novu timer funkciju koja treba da se okine na 30ms. Osvežavanje igre se izvršava funkcijom glutPostRedisplay() koja se poziva unutar Display() funkcije.

## 2.4 GUI

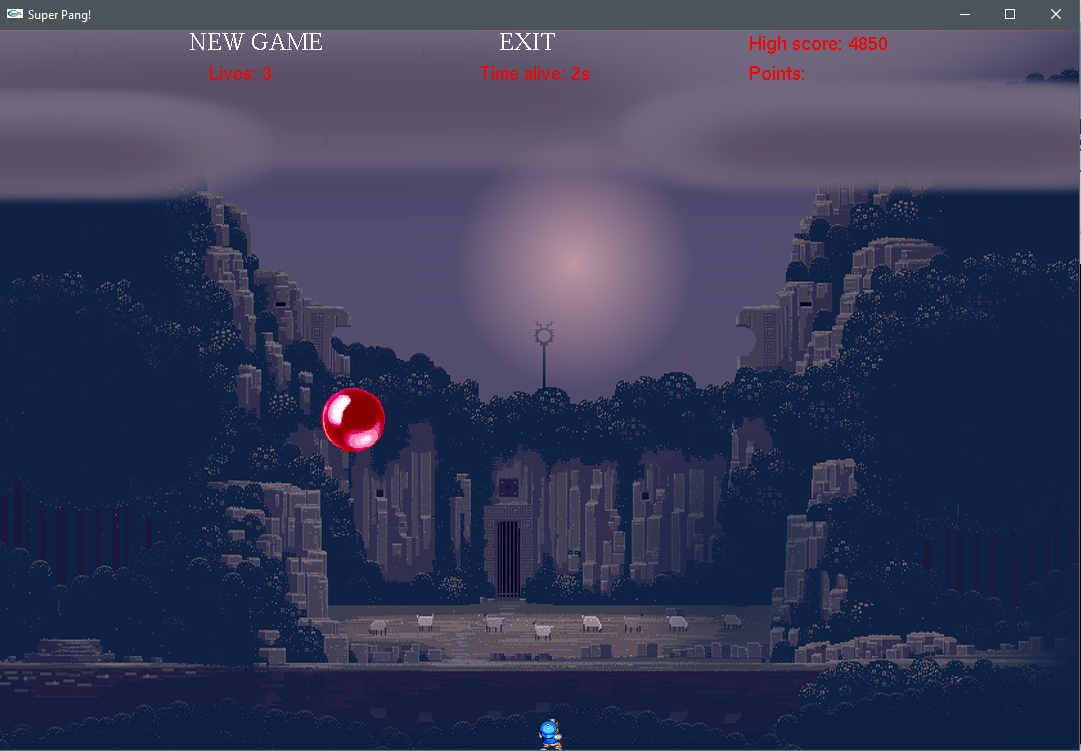
Za prikazivanje interfejsa se koristi datoteka GUI.h koja je definisana u datoteci GUI.cpp.

Svaka funkcija koja uzima int kao parameter koristi pomoćnu funkciju int2char u sebi jer se u njima poziva funkcija glutBitmapCharacter(GLUT\_BITMAP\_HELVETICA\_18, \*char); tako da može da ispiše broj kao niz karaktera. Sve potrebne informacije igraču biće ispisane u gornjem delu prozora. Pauza se prikazuje nakon što igrač pritisne ESC taster. GameOver funkcija se poziva kad igrač izgubi sve živote i prikazaće koliko je igrač sakupio poena u poslednjem igranju. Sam izgled GUI-a će biti prikazan u nastavku kroz demo igre.

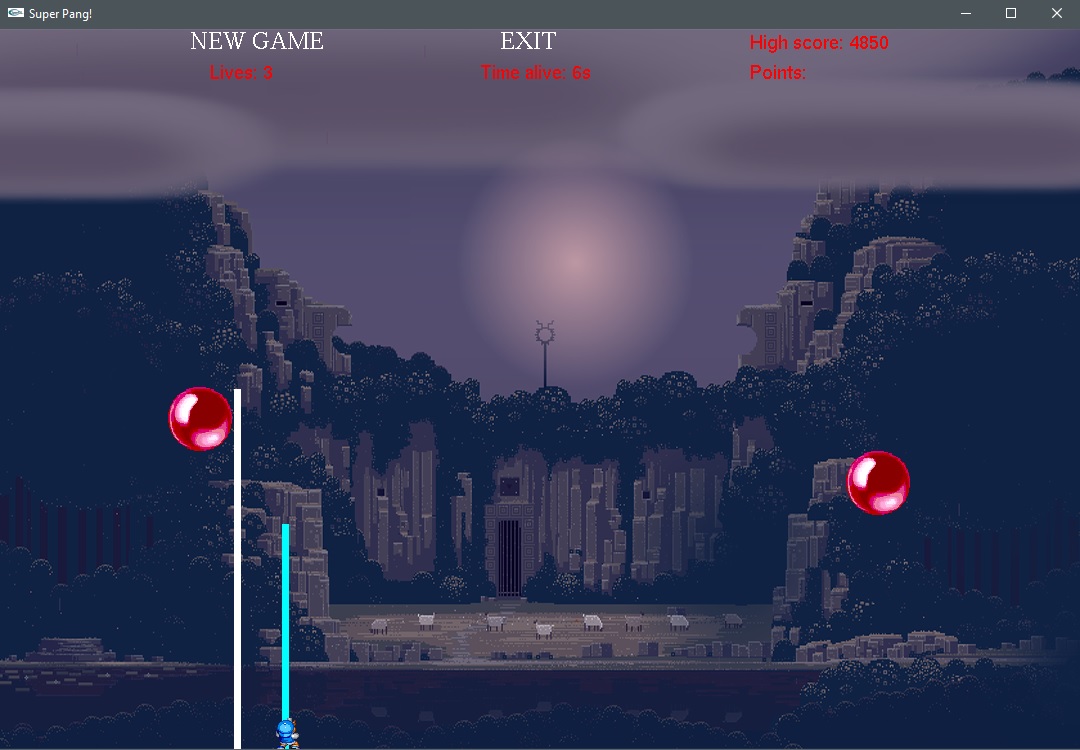
## 2.5 Iscrtavanje karaktera

Iscrtavanje karaktera se vrši pomoću glDrawPixels logičkih operacija koje se vrše nad pikselima kako bi se dobila transparentna pozadina oko lika. Svaki lik ima svoj sprajt i masku koja definiše površinu sprajta koja treba da se prikaže, odnosno pozadinu koja ne treba da se prikaže. Naredne slike prikazuju operacije potrebne za primenu maske. Pre svega potrebno je uključiti logiku nad bojama funkcijom glEnable(GL\_COLOR\_LOGIC\_OP); zatim u prvom koraku se primenjuju pikseli maske sa funkcijom glLogicOp(GL\_AND\_INVERTED);   
  
  
  
  
  
  
zatim se primenjuje na sprajt funkcija glLogicOp(GL\_OR);. Rezultat je da crna pozadina oko sprajta nestaje. Na kraju svega potrebno je isključiti logiku za boje funkcijom glDisable(GL\_COLOR\_LOGIC\_OP);.

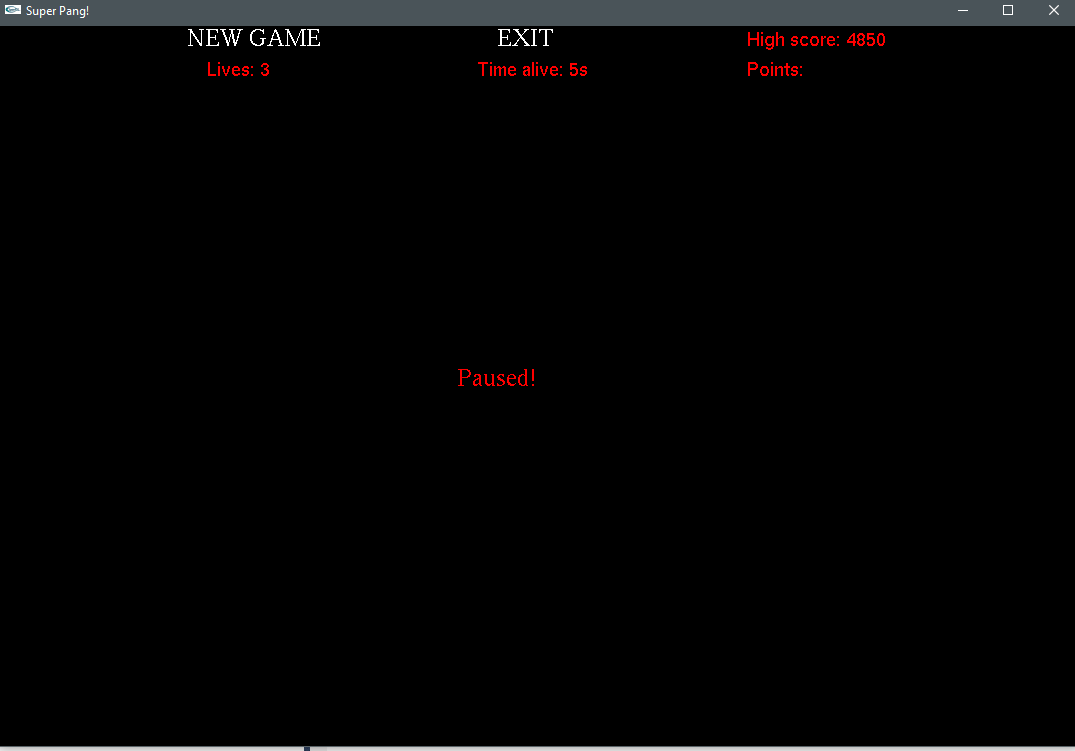
## 2.6 Demo

Kako bi se lakše opisala igra u tekstualnom formatu, sledi par slika koji opisuju gameplay i tok igre. Na sledećoj slici vidimo početak igre. Igrač se nalazi na sredini poda, i kugle kreću da se stvaraju iznad njega. U gornjem delu prozora se vidi GUI. Belim slovima je deo GUI-a koji je interaktivan i u main funkciji je definisana funkcija za proveru klika u delu belih slova.

Na narednim slikama biće prikazano ispaljivanje strele i kako funkcionišu u sudaru sa kuglom.

Na prvoj slici su ispaljene dve strele i treba napomenuti da se leva kugla kreće u desno.  
Na drugoj slici se vidi šta se dešava kada se kugla pogodi. Prva strela je prepolovila kuglu i stvorile su se dve nove. Jedna je krenula levo, druga desno. Zatim je druga strela pogodila desnu kuglu i ona je stvorila dve duplo manje kugle.

Pritiskom ESC tastera igra je pauzirana i igraču se prosleđuje informacija o tome.

Kada igrač izgubi sve živote, poziva se funkcija GameOver iz GUI klase.

# 3. Diskusija i zaključak

Igra izrađena za projekat zadovoljava osnovne zahteve zadatka. Grafika je rađena koristeći OpenGL i GLUT, u programskom jeziku C++. Tokom izrade igre, pronađeno je više mesta koja mogu biti poboljšana i optimizovana. Najpre bi trebalo poboljšati zvukove u igri, jer korišćenje PlaySound() funkcije nije dovoljno. Čak se stvara lag tokom početka igre zbog ove funkcije, ali nakon nekog vremena on nestane. No, fokus ovog kursa je bio na grafiku korišćenjem OpenGL biblioteke, opet, za zaista optimizovanu igru, potrebno je izgraditi dobar endžin koji će da apstrakuje grafiku i neće dozvoliti logiku i pravila igre da utiču na iscrtavanje i ažuriranje.

# 4. Literatura

1. *Predavanja i vežbe za predmet CS367.*
2. *Bouncing ball/gravity tuto. in allegro with c++* <https://www.youtube.com/watch?v=CGb28j6WkHg>