Gimnazija Petra Preradovića

Virovitica

Geometry Rush

Seminarski rad iz informatike

Profesor: Učenik:

Tomislav Šipek Vedran Bedeković

Virovitica, 8.4.2019.

Tema rada: Video igra (Title: Geometry Rush)

Programski jezik izvedbe: Python 3

Third party module: PyGame

**Općenito o programu:**

Geometry Rush je video igra, sastavljena u Pythonu uz pomoć Pygame module-a. Igra ima start meni, upisivanje rezultata nakon završene igre, prikazivanje samih tih rezultata i prozor s informacijama o „moćima“ koje igrač može pokupiti i koristit tijekom igre. Glavni dio programa je naravno sama video igra. Igrač kontrolira kvadratić koji se nalazi na listu papira matematičke bilježnice omeđen granicom nacrtano crvenim markerom i crvenom kemijskom. Ta granica označava granice izvađanja igre. Igrača napada horda neprijateljskih kvadratića koji iz početka napadaju igrača u manjim brojevima ali se tijekom igre poveća njihov broj i brzina. Igrač se brani pucanjem gumica kojima briše kvadratić ako je pogođen. Prilikom njihovog brisanja postoji šansa da bace neki „*power-up*“ . Ako ga igrač pokupi on će dobiti neku vrstu pomoći u obliku pojačanog pucanja ili uvećane brzine i drugo. Cilj igre je preživjeti što dulje i izbrisati što više kvadratića. Za vrijeme trajanja igre, ona se može pauzirati pritiskom na „*escape*“ tipku. Sve slike korištene u programu su nacrtane/uređivane na stranici [www.pixilart.com](http://www.pixilart.com) i programima MS Paint i Photoshop. Zvukovi su skinuti sa [www.youtube.com/audiolibrary/](http://www.youtube.com/audiolibrary/) i freesound.org

**Program se sastoji od:**

* pygame i random module
* inicijalizacije pygame-a

line[[1]](#footnote-1) 10-15:

* određen je RGB za par boja koje će biti korištene u programu

line 18-32:

* određen je prozor igre sa svojom širinom i dužinom
* naslov prozora
* ikona prozora je definirana i postavljena na prozor igre
* loadana je pozadina i njezine inačice
* jpg pozadine su konvertirane kako bi pikseli pozadine bili istog tipa kao i pikseli prozora igre (inače dolazi do usporavanja programa)
* loadan je poluproziran pravokutnik koji će kasnije služiti za zatamnjenje prozora igre tijekom pauze

line 34:

* određen je sat pygame igre koji prati vrijeme u milisekundama

line 36-44:

* definirane su funkcije koje izračunavaju x i y dužine u određenom kutu ( ili ) prema dobivenom vektoru (za brzinu)

line 46-69:

* definirane su funkcije za prikazivanje teksta na ekran
* **text\_objects** rendera dani tekst u željenoj boji i vraća dimenzije okvira tog teksta; on se koristi u **title** i **display\_text** funkcijama
* **title** se koristi za prikazivanje naslova na main meni, on ga centrira i crta na ekran
* **display\_text** koristi se za crtanje centriranog teksta s određenom veličinom fonta na određenoj visini na prozor igre ; koristi se samostalno, a i u **button** funkciji
* **multiline\_text** se koristi za pisanje teksta s pomakom u novi red (poput „*\n*“); odredi se lista koja sadrži teks, početna pozicija na ekran i veličina fonta

line 72-81:

* loadane su slike za okvir gumba, okvir kada je miš iznad gumba i crveni okvir
* **button** nacrta centrirani gumb na ekran, pa na njega stavi tekst uz pomoć **display\_text** funkcije

line 85-108:

* loadane su slike koje prikazuju glavnog lika u različitim smjerovima gledanja i oblicima
* **mc** („*main character“)* funkcija u koja prima pogled i stanje lika te ga crta a ekran po dobivenim parametrima

line 111-182:

* loadana je slika metka (tj. okrugle gumice)
* class **Metak**:
  + funkcija **\_\_init\_\_** se pokreće prilikom stvaranja novog objekta Metak; određuje početni položaj, brzinu, stanje i tip metka
  + **shoot** funkcija crta na ekran novi položaj i zatim kalkulira novi položaj metka; ovisno o tome jeli je metak ispucan iz „sačme“ ili nije mijenja se kalkulacija i koriste se funkcije za malo prije određene kutove 🡪 time se zadrži brzina metka i kada upada pod kut
  + **status** provjerava položaj metka, ako se nalazi iznad granica djelovanja igre, metak će biti postavljen za brisanje
  + **pogodak** provjerava položaj metka i danog neprijateljskog kvadratića; vraća je li je metak pogodio kvadratića ili nije

line 185-427:

* loadane su slike neprijateljskih kvadratića ( kada su živi i njihove faze brisanja)
* class **Enemy**:
  + funkcija **\_\_init\_\_** se pokreće prilikom stvaranja novog objekta Enemy; određuje početni položaj, dimenzije, smjer kretanja (neutralan za sad), brzinu, njegovu šifru (radi provjere sudara), upiti s nasumičnim rezultatom koji će odrediti ponašanje prilikom sudara s ostalim kvadratićima, različite brojače i tip pomicanja prema igraču
  + **movement\_a, movement\_b, movement\_c** funkcije prate poziciju igrača i usmjeruju neprijatelja prema njemu; razlikuju se po točki na igraču koju prate, koriste se funkcije za kut kako bi se očuvala brzina
  + **collision\_with\_other\_enemy** provjerava je li se sami neprijatelj sudario s nekim drugim
  + **check\_collision\_with\_other\_enemy** šifra sad služi kako ne bi provjerio sudar sa samim sobom, lista kroz sve neprijatelje i daje ih u **collision\_with\_other\_enemy** funkciju
  + **avoid\_movement** isto kao ostali **movement** funkcije ali dobiveni smjerovi se dobivaju drugačije
  + **avoid\_other\_enemy\_movement** funkcija služi za određivanje smjera neprijatelja kako bi zaobišao druge neprijatelje, za to vrijeme zanemaruje igrača; cijela funkcija ima nasumične elemente i poziva **check\_collision\_with\_other\_enemy**  i **avoid\_movement** više puta tijekom izvođenja; ako izvede sve izračune i svaki rezultira sudarom, neprijatelj će stajati na mjestu
  + **draw** funkcija crta neprijateljski kvadratić na određenim koordinatama
  + **erased** funkcija je jednostavna animacija brisanja neprijatelja na određenim koordinatama

line 429-494:

* loadane su slike za „kavu, *shotgun, machine gun,* krug, *ghost mode,* gumicu i zvijezdu“
* class **Powerup**:
  + funkcija **\_\_init\_\_** se pokreće prilikom stvaranja novog objekta Powerup; određuje položaj, o kojem je riječ, njegove dimenzije, status, i vrijeme postojanja
  + **draw** funkcija crta „*power-up*“ ovisno o kojemu je riječ; zadnje dvije sekunde postojanja simulirat će se blicanje slike
  + **collid\_with\_mc** funkcijaprovjeri je li se dogodio sudar između „*power-up*“-a i igrača
  + **collected** funkcija zove **collid\_with\_mc** i ako se dogodio sudar, status se mijenja na brisanje, vraća se koji je to „*power-up*“ pokupljen
  + **status** funkcija odbrojava vrijeme postojanja „*power-up*“-a
* **ghost\_vrijeme** funkcija crta tijekom igre koliko sekunda je preostalo na „*ghost mode“* (on je izvan class jer se koristi držanjme tipke space); zadnje tri sekunde tekst postaje crven

line 497-535:

* **enemies\_erased** funkcijacrta broj izbrisanih neprijatelja na prozor igre
* **scoreboard\_dat** funkcija otvara datoteku 'scoreboard.txt' i čita njen sadržaj, nadoda na to ime i rezultat igrača, sortira po rezultatu, sprema ponovo u datoteku
* **high\_score\_beaten** funkcija vraća je li dobiveni rezultat novi najbolji rezultat

line 538-544:

* **sat** funkcija određuje minutu i sekundu trajanja igre i crta na prozor igre

line 547-554:

* loadani su zvukovi i glazba koji će se koristi u programu preko pygame.mixer.music.play ili pygame.mixer.Sound.play()

line 557-625 **MAIN MENU LOOP**:

* odredimo varijablu za while loop, blokadu za klik i status klika
* line 561 početak while loop-a
  + line 561-572: pritiskom x na pygame prozoru izađe iz programa; pritiskom na space započinje igru
  + line 575-625: crta se pozadina na prozor igre, naslov igre; dobiva se pozicija miša, dobiva se je li pritisnut lijevi klik na mišu, crtaju se 4 gumba pomoću **button** funkcijama; izgled gumba se mijenja ovisno je li miš iznad njega; pritiskom na bilo koji gumb mijenja određene varijable i izbacuje iz while loopa; vraća promijenjene varijable

line 628-1166 **GAME LOOP:**

* postave se varijable za vrijeme, odbrojavanje, rezultat i ostale akcije
* line 635-644: početne varijable za igrača
* line 646-652: početne varijable za pucanje
* line 654-663: početne varijable za neprijatelej
* line 665-680: početne varijable za „*power-ups“*
* line 682-704: while loop prije glavnog while loop-a; simulira loading polja igre i prikazuje kontrole i početni položaj igrača
* line 710 početak while loop-a
  + line 712-786: **tipke;** pritiskom na x izlazi iz programa, w, a, s, d za pomicanje igrača (u 8 smjerova), strjelice za pucanje u željenom smjeru (8 mogućih), držanjem tipke space za aktiviranje *ghost mode*-a, puštanjem se deaktivira i escape za pauziranje igre
  + line 789-797: određuje granice za igrača i zaustavlja ga ako i želi prekoračiti
  + line 800-884: izračun za stvaranje neprijatelja izvan vidljivog dijela prozora igre, ima 24 moguća početna položaja koja se određuju nasumično, kod se izvršava svakih „spawn\_rate“ sekundi (početno je 5)
  + line 886-915: provjerava je su li aktivni „*power-up“*-i koji mijenjaju varijable (brzinu igrača, meci po sekundi) i promijeni ih i odbrojava trajanje; također odbrojava trajanje za one koje mijenjaju oblik pucanja
  + line 917-974: prema dobivenom inputu igrača stvorit će metak određenog smjera, ovisno o „*power-up“*-ima može se stvoriti i više od jednom; na kraju postoji štoperica koja blokira pucanje neko određeno vrijeme (kao pola sekunde nakon pucanja se ne može pucati)
  + line 977-984: izračun promjene položaja igrača
  + line 986 i 988: crtanje pozadine na prozor igre
  + line 991-1004: izvršavamo funkcije crtanja i provjere statusa „*power-up“*-ov, ovisno koji se pokupi stavimo određeni string u aktivacijsku listu kojom ćemo izvršiti određene naredbe; stvara se nova skupina (lista) „*power-up“*- koji su još postoje, oni kojima je vrijeme isteklo ili su pokupljeni se brišu
  + line 1007-1038: provjeri koji „*power-up“*-i su aktivirani i postavi njihove varijable na aktiviranje ili ne i započne njihovu štopericu (ako ju imaju), 'extra ghost time' nadoda sekunde za korištenje tog moda, a gumica briše sve neprijatelje na prozoru igre (dodaje se najviše 10 na broj izbrisanih neprijatelja pa i makar ih bilo više)
  + line 1041-1055: kod provjerava status metka i pokreće funkciju za njihovo kretanje po prozoru igre; provjerava se je li došlo do pogotka i na dodaje na broj izbrisanih neprijatelja; briše metke s prozora igre ako im je brisanje postavljeno na True
  + line 1058-1063: kod pretvara milisekunde u sekunde (svake 60-e milisekunde)
  + line 1066-1070: odbrojava sekunde preostale za *ghost mode*
  + line 1073-1084: prolaskom vremena kod ubrzava stvaranje neprijatelja (smanjuje spawn\_rate variajblu, najmanja vrijednost je 0.4); svaki peti obrisani neprijatelj poveća brzinu neprijatelja (najviše na 2.5)
  + line 1088-1098: kod koji prođe za svaki neprijatelj i izvrši njihovo kretanje i crtanje na prozor igre
  + line 1100-1127: pogođeni neprijatelji prolaze kroz kod koji crta na ekran njihovo brisanje, nakon izvršene animacije postoji šansa da bace „*power-up“* i izbrišu se iz igre do daljnjega
  + line 1130-1132: provjeri jeli igrač stoji na mjestu i postavi string za sliku stajanja; provodi crtanje igrača na ekran
  + line 1134-1150: provjerava je li se igrač sudario sa neprijateljem, ako je, prikaže se game over screen i nacrta se dobiveni rezultat množenjem izbrisanih neprijatelja i izdržanih sekundi, game over prozor traje sekundu i pol
  + 1152-1157 i 1167-1208 **PAUSE GAME LOOP**: pritiskom *esc* se pokrene; zasjeni prozor igre i pauzira igru, prikazuje trenutni rezultat i omogućuje povratak na main menu pomoću gumba

line 1211-1351 **PLAY AGAIN LOOP**:

* postavimo varijablu je li nešto odabrano i prazan string za ime
* line 1214 početak while loop-a:
  + line 1216-1308: pritiskom na x izlazi iz programa, **tipke** za unos malih slova engleske abecede, space razmak i backspace brisanje jednog slova, ime može imati najviše 10 slova
  + line 1312-1351: crta se pozadina, crta se tekst „GAME OVER“, ako je postignut novi najbolji rezultat biti će to prikazano inače će samo naglasiti rezultat, prikazuje unos imena, prikazuje svako pritisnuto slovo; imamo gumbe za ponovo igranje ili izlazak na main menu

line 1354-1422 **SCOREBOARD LOOP**:

* line 1355-1376: otvaramo datoteku i pročitamo njezin sadržaj (trenutna tablica s rezultatima); iz pročitanoga izdvojimo bitne podatke (ime, rezultat); odredimo granice koje se mogu pomicati (time prikazujemo najviše 10 osoba i njihov rezultat na prozor igre
* line 1369 početak while loop-a:
  + pročita sadržaj datoteke
  + line 1371-1376: pritiskom na x izlazimo iz programa
  + line 1379-1380: dobivamo položaj miša i je li pritisnut lijevi klik
  + line 1383-1412: crtamo pozadinu i gumbe, pritiskom na koordinate određenog gumba će pomaknuti pregled ljestvice dolje ili gore, još imamo i gumb za povratak u main menu
  + line 1415-1417: crta se poziciju i imena na jednoj strani a rezultate na drugoj prema određenim granicama prilikom izmjene te varijable pritiskom na gumbe; crta i naslov prozora

line 1425-1498 **INFO LOOP**:

* line 1426 početak while loopa-a:
  + line 1429-1434: izlaz iz programa pritiskom na x
  + line 1436-1437: dobivamo položaj miša i je li pritisnut lijevi klik
  + line 1440-1483: nacrta se naslov prozora; nacrta se slika „*power-up*" i pored njega njegovo objašnjenje
  + line 1486-1492: nacrta se gumb za povratak u main meni koji prati položaj i reagira na klik miša

line 1501-1518 **GLAVNI WHILE LOOP:**

* postavljeni su „prekidači“ koji puštaju pojedini dio koda u izvedbu
* ovisno o inputu igrača klikom na gumbe ili „*game over“*-a ranije definirani while loopovi vraćaju njihove nove vrijednosti i ovisno o tome koji je postavljen kao True pokrenut će novi dio koda i tako se izvršava ciklus programa
* ako se izbaci iz glavnog loopa, izvršit će se koda za gašenje cijelog programa

1. line – linija koda [↑](#footnote-ref-1)