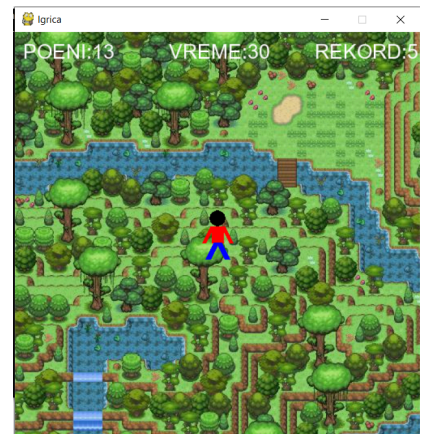

Igrica 1

OPIS IGRICE

Igrač upravlja karakterom preko strelica na tastaturi. Potrebno je da za 30 sekundi skupi što više loptica. Loptice se pojavljuju na slučajnim pozicijama na ekranu. Nakon završetka igre, ukoliko je osvojen novi rekord, ažurira se.



Dodatne napredne funkcionalnosti mogu biti:

- Implementacija zvuka
- Prikaz poruke na kraju igrice
- Igra može biti multiplayer ali na istom računaru
- Određena boja loptice može doneti negativne poene

DISKUSIJA REŠENJA

- Potrebno je da se čuvaju koordinate karaktera tokom cele igrice
- Karakter je nacrtan preko kruga i linija
- Karakterova pozicija je centar njegovog izgleda
- Uslov dodira karaktera i loptice se mora realizovati preko koordinata pozicije oba lika u igrici
- Implementacija pozadine i teksta
- Potrebno je implementirati sistem za događaje sa tastature
- Dodatni txt fajl će biti potreban za upis rekorda

POSTUPAK IMPLEMENTACIJE

- Funkcije za setapovanje scene (tekst, slike, grafički elementi koji se crtaju)
- Glavni deo programa i logika poteza
 - Hvatanje događaja tastature
 - Ažuriranje promenljivih
 - Iscrtavanje
- Nove funkcije koje će se koristiti (neophodno je detaljno ih objasniti prilikom korištenja, njihove parametre, povratnu vrednost i šta rade)
 - Za tekst:
 - Font.SysFont()
 - tekst.render()
 - Za slike:
 - load()
 - Generalno za prikaz na prozor: blit()
- Pripremiti sve neophodne objekte kroz promenljive vidljive svuda u programu

IMPLEMENTACIJA:

```
# crtanje karaktera
```

```
import pygame as pg
import pygamebg as pgb
```

```
# boje
```

```
crna = (0,0,0)
```

```
crvena = (255, 0, 0)
```

```
plava = (0,0,255)
```

```
bela = (255, 255, 255)
```

```
def nacrtajLika(x, y):
```

```
    # glava
```

```
    pg.draw.circle(prozor, crna, (x, y-15), 8)
```

```
    # telo
```

```
    pg.draw.line(prozor, crvena, (x, y-10), (x, y+5), 4)
```

```
    pg.draw.line(prozor, crvena, (x, y-7), (x-7, y+3), 4)
```

```
    pg.draw.line(prozor, crvena, (x, y-7), (x+7, y+3), 4)
```

```
    pg.draw.line(prozor, plava, (x, y+5), (x-10, y+20), 4)
```

```
    pg.draw.line(prozor, plava, (x, y+5), (x+10, y+20), 4)
```

```
prozor = pgb.open_window(500, 500, "Igra")
```

```
prozor.fill(bela)
```

```
# glavni deo igrice
```

```
nacrtajLika(250,250)
```

```
pgb.wait_loop()
```

```
# korak 1 - dodavanje plejsholdera za prikaz poena, vremena, rekorda
```

```
def nacrtajVreme(vreme):
```

```
    slikaVreme = fontVreme.render("Vreme: {}".format(vreme), True, plava)
```

```
    prozor.blit(slikaVreme, (10, 10))
```

```
def nacrtajPoene(poeni):
```

```
    slikaPoeni = fontPoeni.render("Poeni: {}".format(poeni), True, plava)
```

```
    prozor.blit(slikaPoeni, (200, 10))
```

```
def nacrtajMaksRezultat():
```

```
    slikaMaksRezultat = fontMaksRezultat.render("Maks rezultat:
    {}".format(27), True, plava)
```

```
    prozor.blit(slikaMaksRezultat, (330, 10))
```

```
# promenljive u glavnom programu za to, odmah ispod boja dodati
```

```
prozor = pgb.open_window(500, 500, "Igra")
```

```
prozor.fill(bela)
```

```
fontVreme = pg.font.SysFont("Arial", 20)
fontPoeni = pg.font.SysFont("Arial", 20)
fontMaksRezultat = pg.font.SysFont("Arial", 20)
```

```
# poziv datih funkcija
# glavni deo igrice
nacrtajLika(250,250)
nacrtajPoene(12)
nacrtajVreme(30)
nacrtajMaksRezultat()
```

```
# korak 2 - dodavanje da se zaista ucita vrednost iz fajla za rekord
```

```
def ucitajIzFajla():
    fajl = open("rekord.txt", "r") # r, w, a
    rez = fajl.readline()
    fajl.close()
    return rez
```

```
def nacrtajMaksRezultat():
    slikaMaksRezultat = fontMaksRezultat.render("Maks rezultat:
{}".format(ucitajIzFajla()), True, plava)
    prozor.blit(slikaMaksRezultat, (330, 10))
```

```
# korak 3 - dodavanje slike kao pozadine
```

```
# dodavanje promenljive za sliku
prozor = pgb.open_window(500, 500, "Igra")
prozor.fill(bela)
slika = pg.image.load("bg.jpg")
```

```
# dodavanje funkcije za prikaz
```

```
def prikaziPozadinu():
    prozor.blit(slika, (0,0))
```

```
# poziv funkcije za prikaz slike
# glavni deo igrice
nacrtajLika(250,250)
prikaziPozadinu()
```

POSTUPAK DALJE IMPLEMENTACIJE

- Sistem događaja u Pygameu, get() funkcija, šta vraća i kog su tipa objekti u toj listi
- Generalne promenljive za čuvanje pozicija i pomeraja glavnog karaktera

IMPLEMENTACIJA:

```
# korak 1 - dodavanje promenljivih za kretanje
# pocetna vrednost karaktera
```

```

x_koord = 250
y_koord = 250

x_brzina = 0
y_brzina = 0

# negde niže u kodu onda treba da se azuriraju te koordinate
x_koord += x_brzina
y_koord += y_brzina

# iscrtavanje
prozor.fill(bela)
nacrtajLika(x_koord, y_koord)

# korak 2 - funkcija za dohvatanje događaja

tba...

krajIgrice = False

sat = pg.time.Clock()

while krajIgrice == False:
    for event in pg.event.get():
        if event.type == pg.KEYDOWN:
            # taster na tastaturi je pritisnut
            if event.key == pg.K_LEFT:
                x_brzina = -5
            if event.key == pg.K_RIGHT:
                x_brzina = 5
            if event.key == pg.K_UP:
                y_brzina = -5
            if event.key == pg.K_DOWN:
                y_brzina = 5
        elif event.type == pg.KEYUP:
            if event.key == pg.K_LEFT or event.key == pg.K_RIGHT:
                x_brzina = 0
            elif event.key == pg.K_UP or event.key == pg.K_DOWN:
                y_brzina = 0
        elif event.type == pg.QUIT: # klik na x
            krajIgrice = True

    x_koord += x_brzina
    y_koord += y_brzina

# iscrtavanje
prozor.fill(bela)
nacrtajLika(x_koord, y_koord)
nacrtajPoene(12)
nacrtajVreme(30)
nacrtajMaksRezultat()

```

```
pg.display.update()
sat.tick(60)
```

```
pgb.wait_loop()
```

DALJE IMPLEMENTACIJA:

- Dodavanje kruga koji se kupi
- Dodavanje poena
- Dodavanje pozadine
- Crtanje timera
- Programiranje timera
- Rekord prikaz
- Rekord programiranje

PRIMER 1 Dodavanje kruga koji se kupi

```
def nacrtajKrug(x, y):
    pygame.draw.circle(prozor, narandzasta, (x, y), 7)

def generisiRandomKoord():
    x = random.randint(0, 500)
    y = random.randint(0, 500)
    return x, y

pomeraj_x = 0
pomeraj_y = 0

(krug_x, krug_y) = generisiRandomKoord()
...
#crtanje
prozor.fill(bela)
nacrtajCikicu(x, y)
nacrtajKrug(krug_x, krug_y)
```

PRIMER 2 Azuriranje poena

```
x = x + pomeraj_x
y = y + pomeraj_y

if abs(x - krug_x) < 7 and abs(y - krug_y) < 7:
    poeni += 1
    (krug_x, krug_y) = generisiRandomKoord()

#crtanje
prozor.fill(bela)
nacrtajCikicu(x, y)
nacrtajKrug(krug_x, krug_y)
nacrtajRezultat(poeni)
```

PRIMER 3 Dodavanje pozadine

```
font = pygame.font.SysFont("Arial", 20)
slika = pygame.image.load("background.png")
```

```
....
def prikaziSliku():
    prozor.blit(slika, (0,0))
```

```
#crtanje
prozor.fill(bela)
prikaziSliku()
nacrtajKrug(krug_x, krug_y)
```

PRIMER 4 Timer - crtanje

```
def nacrtajTimer(vreme):
    tekst = font_timer.render("Vreme: {}".format(vreme), True, crvena)
    prozor.blit(tekst, (400, 20))
```

```
....
font_timer = pygame.font.SysFont("Arial", 20)
```

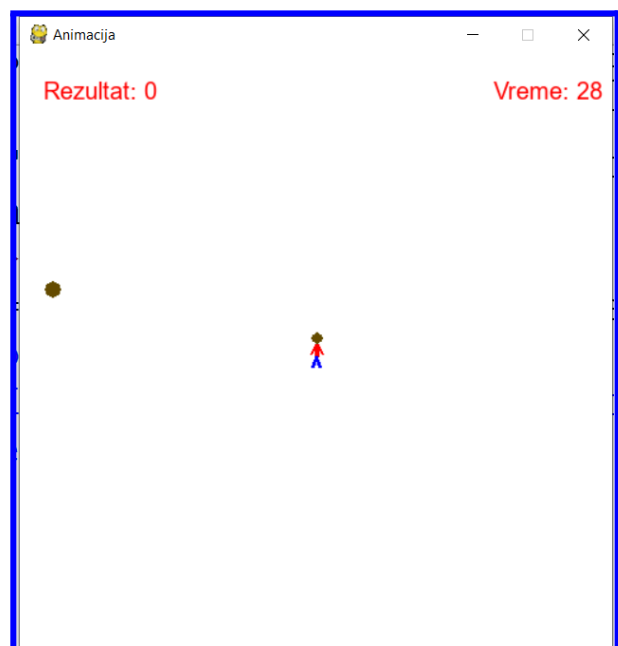
```
sat = pygame.time.Clock()
maxVreme = 30 # sekunde
```

```
#crtanje
    prozor.fill(bela)
    nacrtajCikicu(x, y)
    nacrtajKrug(krug_x, krug_y)
    nacrtajRezultat(poeni)
    nacrtajTimer(maxVreme)
```

PRIMER 5 Timer - programiranje

```
sat = pygame.time.Clock()
maxVreme = 30 # sekunde
timerEvent = pygame.USEREVENT + 1
pygame.time.set_timer(timerEvent,
1000)
```

```
for event in pygame.event.get(): #
    [strelica gore, klik, desni klik ,
        if event.type == pygame.QUIT:
            ...
        elif event.type ==
pygame.KEYDOWN:
            ...
        elif event.type == pygame.KEYUP:
            ...
        elif event.type == timerEvent:
            maxVreme = maxVreme - 1
```



```
print(maxVreme)
if maxVreme <= 0:
    kraj = True
```

PRIMER 6 Rekord - prikaz

```
def prikaziRekord(rekord):
    tekst = font.render("Rekord:"+str(rekord), True, crna)
    prozor.blit(tekst, (200,20))
```

```
font = pygame.font.SysFont("Arial", 20)
font_timer = pygame.font.SysFont("Arial", 20)
font_rekord = pygame.font.SysFont("Arial", 20)
```

```
// pre glavne while petlje
rekord = 10
```

```
// u delu gde se sve crta
    rezultat(poeni)
    nacrtajTimer(maxVreme)
    prikaziRekord(rekord)
```

PRIMER 7 Rekord - citanje iz fajla

```
def procitajRekord():
    file = open("rez.txt", "r")
    rez = file.readline()
    return rez
```

```
# poeni
poeni = 0
```

```
rekord = procitajRekord()
```

PRIMER 5 Rekord - upis u fajl

```
def upisiRekord(broj):
    file = open("rez.txt", "w")
    file.write(str(broj))
    file.close()
```

```
// ispod while petlje
    sat.tick(60)
```

```
if poeni > int(rekord):
    upisiRekord(poeni)
```

```
pygamebg.wait_loop()
```