

Zadatak 1.

Pomeranje loptice po ekranu.

 Setanje loptice

— □ ×

```
import random
import pygame as pg
import pygamebg

(sirina, visina) = (400, 400)    # otvaramo prozor
prozor = pygamebg.open_window(sirina, visina, "Setanje loptice")

# podešavamo događaje tastaturom - prvi događaj se generiše nakon
# 50ms, a svaki naredni nakon 25ms
pg.key.set_repeat(50, 25)

# funkcija koja vraća nasumično odredjenu boju
def nasumicna_boja():
    return (random.randint(0, 255), random.randint(0, 255), random.randint(0, 255))

(x, y) = (sirina / 2, visina / 2)    # koordinate centra prozora
boja = nasumicna_boja()              # boja loptice se odredjuje nasumično
r = 40                               # poluprečnik loptice
(dx, dy) = (10, 10)                  # pomeraji po x i y koordinati

def crtaj():
    prozor.fill(pg.Color("white"))    # bojimo prozor u belo
    pg.draw.circle(prozor, boja, (round(x), round(y)), r) # crtamo lopticu

def obradi_dogadjaj(dogadjaj):
    global x, y, boja

    if dogadjaj.type == pg.KEYDOWN:
```



```

# strelica na levo
if dogadjaj.key == pg.K_LEFT:
    x -= dx
    if x - r < 0:
        x = r
        boja = nasumicna_boja()
    return True
# strelica na desno
if dogadjaj.key == pg.K_RIGHT:
    x += dx
    if x + r > sirina:
        x = sirina-r
        boja = nasumicna_boja()
    return True
# strelica na gore
if dogadjaj.key == pg.K_UP:
    y -= dy
    if y - r < 0:
        y = r
        boja = nasumicna_boja()
    return True
# strelica na dole
if dogadjaj.key == pg.K_DOWN:
    y += dy
    if y + r > visina:
        y = visina-r
        boja = nasumicna_boja()
    return True
return True

```

```

# ne treba ponovo nacrtati scenu

```

```

pygamebg.event_loop(crtaj, obradi_dogadjaj)

```

Zadatak 2.

Primer sa balonima i pogađanjem slova.

```
import random
import pygame as pg
import pygamebg

(sirina, visina) = (800, 600)
prozor = pygamebg.open_window(sirina, visina, "Igrica - baloni")

# funkcija ispisuje dati tekst, date veličine tako da mu je gornje levo teme na datoj poziciji
def tekst_levo(x, y, tekst, velicina):
    font = pg.font.SysFont("Arial", velicina) # font kojim će biti prikazan broj poena
    tekst = font.render(tekst, True, pg.Color("black"))
    prozor.blit(tekst, (x, y))

# funkcija ispisuje dati tekst, date veličine tako da mu je centar na datoj poziciji
def tekst_centar(x, y, tekst, velicina):
    font = pg.font.SysFont("Arial", velicina)
    tekst = font.render(tekst, True, pg.Color("black"))
    (sirina_teksta, visina_teksta) = (tekst.get_width(), tekst.get_height())
    (x, y) = (x - sirina_teksta / 2, y - visina_teksta / 2)
    prozor.blit(tekst, (x, y))

balon = pg.image.load("balon.png") # učitavamo sliku balona
zivot = pg.transform.scale(balon, (36, 50)) # sličica života je umanjena slika balona

MAKS_BALONA = 3 # maksimalni broj slova koje mogu biti istovremeno na ekranu
```

Igrica - baloni
Поени: 0



```
# baloni = [] # baloni koji su trenutno na ekranu (trojke koje čine x koordinata, y koordinata i slovo)
poeni = 0 # trenutni broj poena
zivoti = 3 # trenutni broj života

# dodajemo balon na vrh ekrana, ako su za to ispunjeni uslovi
def dodaj_balone():
    # ako trenutno ne pada ni jedno slovo tada obavezno dodajemo novo slovo
    # ako pada manje od MAKS_SLOVA tada bacamo kockicu tako da očekujemo da
    # se slovo doda u 1% slučajeva
    if len(baloni) == 0 or (len(baloni) < MAKS_BALONA and random.randint(1, 100) == 1):
        x = random.randint(0, sirina - balon.get_width()) # nasumično biramo x koordinatu gornjeg levog ugla balona, tako da balon bude unutar prozora
        y = 25 # ostavljamo prostor za statusnu liniju na vrhu (u njoj se prikazuju poeni i životi)
        slovo = chr(ord('A') + random.randint(0, 25)) # nasumično biramo slovo abecede
        # raspoloziva_slova = ['A', 'S', 'D', 'F', 'G', 'H', 'J', 'K', 'L']
        # slovo = raspoloziva_slova[random.randint(0, len(raspoloziva_slova) - 1)]
        baloni.append([x, y, slovo]) # dodajemo balon u listu

# sva slova pomeramo naniže i uklanjamo ona koja su pala na zemlju
def pomeri_balone():
    global baloni, zivoti

    brzina = 0.5 * (poeni // 10 + 1) # na svakih 10 poena povećava se brzina
    for balon in baloni: # svim balonima u listi
        balon[1] += brzina # povećavamo y koordinatu

    preostali_baloni = [] # u listu preostali_baloni prepisujemo sve balone koji nisu ispali sa ekrana
    for balon in baloni:
        if balon[1] > visina: # ako je balon ispao sa ekrana
            zivoti -= 1 # smanjujemo broj života
```

```

        else:
            preostali_baloni.append(balon)
            baloni = preostali_baloni
            # u suprotnom
            # ubacujemo ga u listu preostalih balona
            # nastavljamo da radimo samo sa preostalim balonima

# prikazujemo sadržaj prozora
def crtaj():
    prozor.fill(pg.Color("white"))
    if zivoti > 0: # ako igra nije završena
        # prikazujemo broj poena
        tekst_levo(5, 5, "Поени: " + str(poeni), 20)
        # prikazujemo preostale živote (iscrtavanjem umanjених sličica balona)
        for i in range(1, zivoti + 1):
            prozor.blit(zivot, (sirina - 5 - i*zivot.get_width(), 5))
        # prikazujemo sve balone
        for [x, y, slovo] in baloni:
            prozor.blit(balon, (x, y)) # prikazujemo sliku balona
            (cx, cy) = (x + balon.get_width() / 2, y + 2 * balon.get_height() / 5) # određujemo centar
teksta
            tekst_centar(cx, cy, slovo, 40) # prikazujemo tekst
        else: # ako igra jeste završena
            tekst_centar(sirina // 2, visina // 2, "КРАЈ! БРОЈ ПОЕНА: " + str(poeni), 60) # prikazujemo
rezultat

def obradi_dogadjaj(dogadjaj):
    global baloni, poeni, zivoti
    if dogadjaj.type == pg.KEYDOWN:
        if pg.K_a <= dogadjaj.key and dogadjaj.key <= pg.K_z:
            # pritisnut je taster
            # na tasteru je neko slovo
abecede
            pritisnuto_slovo = chr(ord('A') + (dogadjaj.key - pg.K_a)) # određujemo koje je
slovo u pitanju
            preostali_baloni = [[x, y, slovo] for [x, y, slovo] in baloni
                                if slovo != pritisnuto_slovo] # preostali su oni
baloni koji ne sadrže pritisnuto slovo

```

<pre> poeni += len(baloni) - len(preostali_baloni) za broj izbačenih slova baloni = preostali_baloni samo sa preostalim balonima elif dogadjaj.type == pg.MOUSEBUTTONDOWN: if zivoti <= 0: zivoti = 3 baloni = [] poeni = 0 def novi_frejm(): dodaj_balone() # dodajemo balone ako su za to ispunjeni uslovi pomeri_balone() # pomeramo balone crtaj() pygamebg.frame_loop(50, novi_frejm, obradi_dogadjaj) </pre>	<pre> # uvećavamo broj poena # nastavljamo da radimo # kliknuto je dugme miša # ako je igra završena # započinjemo novu </pre>
--	--