Python i Pygame

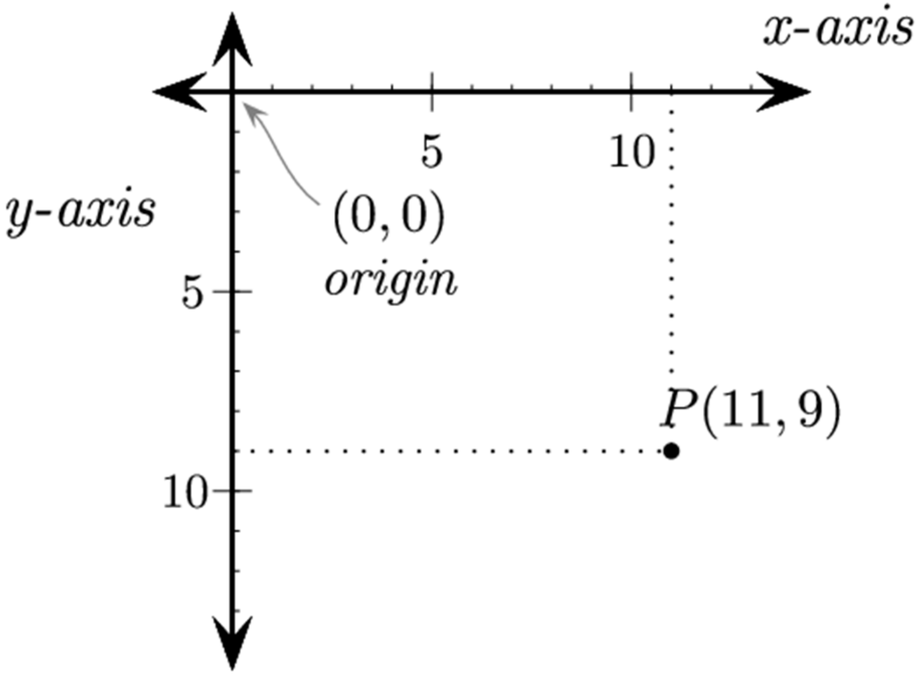
Programiranje arkadnih igrica u Pythonu koristeći Pygame

Uvod u grafiku

Koordinatni sustav

- Računalni koordinatni sustav ima točku 0,0 u gornjem lijevom kutu ekrana

- Pozitivne vrijednosti X osi koordinatnog sustava idu s lijeva na desno
- Pozitivne vrijednosti Y osi koordinatnog sustava idu odozgo prema dolje



Pygame

- Radi jednostavnijeg programiranja igrica koristit ćemo se Pygame bibliotekom funkcija
- Pygame biblioteka olakšava rad sa:
  - Grafičkim oblicima
  - Bitmap slikama
  - Animacijama
  - Mišem, tipkovnicom i gamepad-om
  - Zvukom

- Prije korištenja pygame biblikoteke u vlastitom programu potrebno je:

```
#Importiraj biblioteku funkcija Pygame
import pygame
#Inicijaliziraj game engine
pygame.init()
```

Boje

- Svaka boja se definira kao kombinacija tri osnovne boje – RGB - Red, Gree, Blue
  Udio svake osnovne boje se izražava kao broj 0-255
  Primjer:
  BLACK = (0, 0, 0)
  WHITE = (255, 255, 255)
  GREEN = (0, 255, 0)
  RED = (255, 0, 0)
  BLUE = (0, 0, 255)

- Za provjeru RGB vrijednosti mogu se koristiti aplikacije poput Colorpicker-a:
  http://www.colorpicker.com/

Otvaranje prozora

- Za otvaranje glavnog prozora igrice koriste se slijedeće naredbe:

```
# Definiranje sirine i visine prozora u pixelima
size = (700, 500)
# Otvaranje prozora
screen = pygame.display.set_mode(size)
```

- Za spremanje dva podatka u jednu varijablu koristili smo nepromjenjivi niz (Tuple)

```
size = (700, 500)
```

  Vrijednosti se pišu unutar zagrada i odvajaju zarezom

- Podešavanje naslova prozora

```
pygame.display.set_caption("Moja igrica")
```

Interakcija sa igračem

- Interakcija sa igračem odvija se unutar glavne petlje programa

```
# Varijabla koja označava kad završavamo program
done = False
# varijabla koja prati brzinu izvršavanja programa
clock = pygame.time.Clock()
# Glavna petlja
while not done:
    # Petlja glavnog eventa
    for event in pygame.event.get() # Hvatanje akcije igrača
        if event.type == pygame.QUIT # Igrač je pritisnuo close window
            done = True # postavljamo varijablu za kraj programa
    # Ovdje dolazi logika igrice
    # Ovdje dolazi crtanje ekrana

    # Postavi boju ekrana na bijelo
    screen.fill(WHITE)
    # Ažuriranje ekrana
    screen.display.flip()
    # Postavi limit na 60 FPS
    clock.tick(60)
```

Hvatanje akcije igrača

- Hvatanje akcije igrača se radi unutar event processing petlje
  Ova petlja reagira na igračev pritisak tipkovnice ili aktivnost mišem
  Primjer:

```
# Varijabla koja označava kad završavamo program
for event in pygame.event.get():
    if event.type == pygame.QUIT:
        print("Igrac zeli zavrsiti igricu")
    elif event.type == pygame.KEYDOWN:
        print("Igrac je pritisnuo tipku")
    elif event.type == pygame.KEYUP:
        print("Igrac je otpustio tipku")
    elif event.type == pygame.MOUSEBUTTONDOWN:
        print("Igrac je pritisnuo tipku na misu")
```

event je lista koja bilježi sve aktivnosti igrača
for petlja prolazi kroz listu od jedne do druge aktivnosti
if naredba se koristi za detekciju pojedinih aktivnosti

Procesiranje frejma i završetak

- Procesiranje svakog frejma se odvija slijedećom logikom:

```
while not done:

    - Za svaki event koji je igrač generirao
        - Pomoću niza if naredbi odgovori na evente
    - Izračunaj pomicanje objekata, interakciju objekata, itd.
    - Obriši ekran
    - Iscrtaj sve ponovo
```

- Na kraju programa za ispravno završavanje programa se dodaje: `pygame.quit()`

Crtanje

- Linija
  pygame.draw.line(engine, boja, [px, py], [zx, zy], debljina_linije)

```
pygame.draw.line(screen, GREEN, [0, 0], [100, 100], 5)
```

- Pravokutnik
  pygame.draw.rect(engine, boja, [px, py, sirina, visina], debljina_linije)

```
pygame.draw.rect(screen, GREEN, [0, 0, 250, 100], 2)
```

- Elipsa
  pygame.draw.ellipse(engine, boja, [cx, cy, sirina, visina], debljina_linije)

```
pygame.draw.ellipse(screen, GREEN, [20, 20, 250, 100], 2)
```

Crtanje, nastavak

- Luk (dio elipse)
  pygame.draw.arc(engine, boja, [cx, cy, sirina, visina], pkut, zkut, debljina_linije)

```
pygame.draw.arc(screen, GREEN, [100, 100, 250, 200], PI, PI/2, 5)
```

- Poligon
  pygame.draw.polygon(engine, boja, [[t1x, t1y], [t2x, t2y], [t3x, t3y]], debljina_linije)

```
pygame.draw.polygon(screen, GREEN, [[100, 100], [0, 200], [200,
200]], 2)
```

Crtanje teksta

- Tekst se iscrtava u 3 koraka
  - 1. korak: definiranje varijable koja sadrži info o fontu
    var_font = pygame.font.SysFont(font, vel, bold, italic)
    ```
    font = pygame.font.SysFont("Calibri", 25, True, False)
    ```
  - 2. korak: kreiranje slike teksta
    var_tekst = var_font.render(tekst_za_ispis, anti_aliasing, boja)
    ```
    tekst = font.render("Moj tekst", True, BLACK)
    ```
  - 3. korak: iscrtavanje slike teksta na ekranu
    screen.blit(var_tekst, [tx, ty])
    ```
    screen.blit(tekst, [250, 250])
    ```

Kviz

- Slijedi link ispod:

  http://programarcadegames.com/quiz/quiz.php?file=graphics&lang=en

Kviz odgovori

- P1: If a box is drawn starting at (x,y) coordinate (0,0), where will it be on the screen?
  - It won't display
  - Upper right
  - Lower right
  - Upper left
  - Center
- P2: If the screen width and height are both 400 pixels, and a rectangle is drawn starting at (0,400), where will it display?
  - Upper right
  - Center
  - Lower left
  - It won't display
  - Lower right

Kviz odgovori, nastavak

- P3: In computer graphics, as x and y coordinates increase in value, a point will move:
  - Down and to the left
  - Up and to the right
  - Nowhere
  - Down and to the right.
  - Up and to the left
- P4: What color would be defined by (0, 0, 0)?
  - White
  - Red
  - Blue
  - Black
  - Green

Kviz odgovori, nastavak

- P5: What color would be defined by (0, 255, 0)?
  - Black
  - Red
  - Blue
  - White
  - Green
- P6: What color would be defined by (255, 255, 255)?
  - White
  - Red
  - Blue
  - Black
  - Green

Kviz odgovori, nastavak

- P7: What code will open up a window 400 pixels high and 800 pixels wide?
  - screen = pygame.display.open_window(400, 800)
  - size = [800, 400]
    screen = pygame.display.set_mode(size)
  - size = [400, 800]
    screen = pygame.display.set_mode(size)
  - size = 800 x 400
    screen = pygame.display.set_mode(size)
  - screen = pygame.display.open_window(800, 400)
- P8: What is the main program loop?
  - It runs once for the entire game.
  - It processes user input, updates objects, and draws the screen each frame of the game.
  - It loops once for each level of the game.
  - It loops once for each life that the player has.

Kviz odgovori, nastavak

- P9: Where does this code go?
  `clock = pygame.time.Clock()`
  - This code is placed before the main program loop.
  - The code is place after the main program loop.
  - The code is placed inside the main program loop.
- P10: Where does this code go, and what does it do?
  `clock.tick(20)`
  - The code is place after the main program loop and limits the game to 20 frames per second.
  - This code is placed before the main program loop and limits the game to 20 frames per second.
  - The code is placed inside the main program loop and limits the game to 20 frames per second.
  - The code is place after the main program loop and pauses 20 milli-seconds.
  - The code is placed inside the main program loop and pauses 20 milli-seconds.

Kviz odgovori, nastavak

- P11: Changing this code from 20 to 30 will cause what to happen?
  `clock.tick(20)`
  - The game will run faster.
  - The game will run slower.
  - Nothing.
- P12: What does this code do?
  `pygame.display.flip()`
  - Nothing.
  - Flips the screen from top to bottom.
  - Displays everything that has been drawn so far.
  - Flips the screen from left to right.
  - Clears the screen.

Kviz odgovori, nastavak

- P13: What code will draw a line from x, y coordinates (0, 0) to (100, 100)?
  - pygame.draw.line(5, GREEN, [0, 0], [100, 100], screen)
  - pygame.draw.line(screen, GREEN, [0,0,100,100], 5)
  - pygame.draw.line(screen, GREEN, 0, 0, 100, 100, 5)
  - pygame.draw.line(screen, GREEN, [0, 0], [100, 100], 5)
  - pygame.draw.line(GREEN, screen, 0, 0, 100, 100, 5)
- P14: What does this code draw?

```
offset = 0
while offset < 100:
    pygame.draw.line(screen, RED, [50+offset,20], [50+offset,60], 5)
    offset = offset + 10
```

  - Ten vertical lines, 5 pixels apart, with a starting x coordinate of 50 and an ending coordinate of 100.
  - Ten vertical lines, 10 pixels apart, with a starting x coordinate of 50 and an ending coordinate of 140.
  - Ten vertical lines, 10 pixels apart, with a starting x coordinate of 10 and an ending coordinate of 110.
  - Ten lines, all drawn on top of each other.
  - Ten vertical lines, 5 pixels apart, with a starting x coordinate of 10 and an ending coordinate of 150.

Kviz odgovori, nastavak

- P15: How wide will this ellipse be?

  ```
  pygame.draw.ellipse(screen, BLACK, [0, 0, 100, 100], 2)
  ```

  - 49 pixels
  - 101 pixels
  - 99 pixels
  - 50 pixels
  - 100 pixels
- P16: Where will the center of this ellipse be?

  ```
  pygame.draw.ellipse(screen, BLACK, [1, 1, 3, 3], 1)
  ```

  - (2, 2)
  - (3, 3)
  - (1, 1)

Za vježbu

- Dopuniti kostur programa sa slajda 6 naredbama za crtanje linije, pravokutnika, elipse, luka, poligona (trokuta) i teksta
- Rješenje:

```
# Importiraj biblioteku funkcija Pygame
import pygame
# Inicijaliziraj game engine
pygame.init()

# Definiraj boje i broj PI
BLACK = (0, 0, 0)
WHITE = (255, 255, 255)
GREEN = (0, 255, 0)
RED = ( 255, 0, 0)
BLUE = (0, 0, 255)
PI = 3.141592653

# Postavi veličinu prozora
size = (400, 500)
screen = pygame.display.set_mode(size)
pygame.display.set_caption("Moja igrica")

# Varijabla koja označava kad završavamo program
done = False
# varijabla koja prati brzinu izvršavanja programa
clock = pygame.time.Clock()
```

Za vježbu, nastavak

```
# Glavna petlja
while not done:
    # Petlja glavnog eventa
    for event in pygame.event.get(): # Hvatanje akcije igrača
        if event.type == pygame.QUIT: # Igrač je pritisnuo close window
                    done = True # postavljamo varijablu za kraj programa

    # Postavi boju ekrana na bijelo
    screen.fill(WHITE)

    # Nacrtaj liniju od (0,0) do (100,100) sirine 5 pixela
    pygame.draw.line(screen, GREEN, [0, 0], [100, 100], 5)

    # Nacrtaj pravokutnik od (20, 20) sirine 250 visine 200 debljine linije 2
    pygame.draw.rect(screen, BLACK, [20, 20, 250, 100], 2)

    # Nacrtaj elipsu unutar prethodnog pravokutnika
    pygame.draw.ellipse(screen, BLACK, [20, 20, 250, 100], 2)

    # Nacrtaj luk centar u (20,220) širina 250, visina 200, početni kut 0 , zavrsni kut PI/2
    # debljina linije 2
    pygame.draw.arc(screen, BLACK, [20, 220, 250, 200], 0, PI/2, 2)
```

Za vježbu, nastavak

```
    # Nacrtaj trokut izmedju tocaka (100,100) (0,200) (200,200) crne boje i debljine linije 5
    pygame.draw.polygon(screen, BLACK, [[100, 100], [0, 200], [200, 200]], 5)

    # Nacrtaj tekst "Moj tekst", font Colibri, velicina 25, boja crna, bold, ne italic,
    # anti-aliasing
    font = pygame.font.SysFont("Calibri", 25, True, False)
    text = font.render("Moj tekst", True, BLACK)
    screen.blit(text, [250, 250])

    # Ažuriranje ekrana
    pygame.display.flip()

    # Postavi limit na 60 FPS
    clock.tick(60)

# Po izlasku iz petlje zatvori ekran
pygame.quit()
```