Programozói dokumentáció: Flappy Bird

Bogyó Sándor Lóránt – TG2CM4

A Flappy Bird játék elindításához szükséges a Pygame grafikus környezet. Ammenyiben ez telepítve van, a játék rögtön elkezdhető.

A program egyik legfontosabb eleme, vagyis adatszerkezete az Objektum class.

class Objektum:  
 def \_\_init\_\_(self, szelesseg, magassag, kep):  
 self.szelesseg = szelesseg  
 self.magassag = magassag  
 self.kep = kep

Ennek segítségével tároljuk el egy elemhez, vagyis Objektumhoz a pontos pozíciót az ablakban (szélesség, és magasság), illetve az elem képét is. A madár, a csövek, a talaj, és minden ehhez tartozik.

A másik fontos eleme a programnak a Valasztas class (main fv.):

class Valasztas:  
 fomenu = 1  
 jatek = 2  
 gamover = 3  
 eredmeny = 4

A main függvény ennek segítségével hívja meg a megfelelő menüpontot a programban. Például ha a felhasználó a főmenüben a játék gombra kattint, a fomenu függvény visszatéríti a jatek függvény számát a main függvénynek, a main függvény pedig meghívja a jatek-ot miután a valasztas átváltozott.

valasztas = Valasztas.fomenu  
  
while True:

…  
 elif valasztas == Valasztas.jatek:  
 valasztas = jatek()

# Menüpontok, avagy a legfontosabb függvények

## Főmenü: fomenu()

## Játék: jatek()

## Játék vége: def gamover(talaj, cso, madar, hatter, eredmenyszamlalo): Bekéri a játék elemeit: a talajt, a madarat, a hátteret, és az eredményt.

## Eredmények: eredmeny()

# Modulok

A program 3 modulra van felosztva: main, eredmenyek, és betoltesek

## Az eredmenyek modul

Ebben a modulban tároljuk az eredmények kiszámításához, a név és eredmény kiírásához, megjelenítéséhez tartozó összes függvényt.

**Fontosabb függvények, adatszerkezetek:**

**class NevPont:** Ennek segítségével lesz egy neve és pontszáma egy elemnek.

**def szamlalas(cso):** A pontszámhoz hozzáad mindig egyet, amikor a madár a cső végéhez ér. Ehhez bekéri a csövet, hogy tudja a cső szélességét, és ennek a segítségével határozza meg.

**def listaban():** Beolvassa az eddigi eredményeket, és azokat visszaadja egy listában rendezve.

**def kiir(nev, eredmeny):** Kiírja a pontszámot és nevet az eredmenyek.txt fájlba. (pl. Béla 5) Ehhez bekéri a nevet, és az eredményt.

**def szamokmegjelenitese(felulet, eredmenyszamlalo, szamok):** Ennek segítségével jeleníti meg számokat a képernyőn a (szám)képekből. Ehhez bekéri a felületet, hogy hová rakja, illetve a mostani eredményt, és a számképeket is amit meg kell jelenítenie.

**def nagytabla(felulet):** A nagy eredménytábla betöltője, ehhez bekéri a felületet, amin meg kell jelenítenie.

**def nevbeker(felulet):** Bekéri a játékos nevét, ehhez bekéri a felületet, amin meg kell jelenítenie.

**def maxpontszame(eredmenyszamlalo):** Megnézi, hogy a játékos rekordot ért-e el, ehhez bekéri a jelenlegi pontszámot.

**def legjobbnev(eredmenyszamlalo, nev):** Megnézi, hogy a játékos a rekorder-e, ha igen akkor az ő nevét adja vissza, ellenkező esetben az eddigi rekorderét, ehhez pedig bekéri a jelenlegi pontszámot és a nevet.

**def maxpontnevkiiras(felulet, eredmenyszamlalo, kiiras):** Kiírja a maxpontszámot, illetve az eddigi rekorder nevét is amennyibe az nem a mostani játékos, ehhez bekéri a felületet, amin meg kell jelenítenie, a jelenlegi eredményt illetve a kiirást, ami megmondja, hogy ki lett-e már írva.

**def jatekosnevmegjelenites(felulet, nev, legjobbnev):** Kiírja a rekorder nevét, akár a játékosét is amennyiben ő az, ehhez bekéri a felületet, amin meg kell jelenítenie, illetve a neve(ket)t is.

**def pontszam(felulet, eredmenyszamlalo):** Jelenlegi pontszám kiírása az eredménytáblára, ehhez bekéri a felületet, amin meg kell jelenítenie, illetve a jelenlegi eredményt.

## A betoltesek modul

Ebben a modulban tároljuk azokat a függvényeket, amelyek a betöltést, megjelenítést (például képek) illetve a beolvasást (név bekérése) segítik.

**Függvények, adatszerkezetek:**

**def szovegbeolvasas(x, y, width, height, bg\_color, fg\_color, font, felulet):** Bekéri a szöveget, esetünkben a nevet a felhasználótól, ehhez a függvény bekéri azt, hogy hová jelenítsük meg a szövegdobozt, és milyen hosszú és széles legyen, illetve milyen színnel, és betűtípussal írja ki ezeket.

**def ablak():** Betölti a pygame-t, az ablak címét, faviconját

**def kep(kepnev):** Beolvassa a képeket, majd konvertálja is, ehhez pedig csak a képünk nevét kéri be

**def kepmegjelenites(ablak, nev):** Ennek segítségével jeleníti meg az adott objektumot a képernyőn, ehhez bekéri a felületet, amin meg kell jelenítenie, illetve a kép nevét amit meg kell jelenítenie.

**def szamok():** Ezzel tölti be az összes számot ami kép formátumú

## A main modul

A program szíve, a legfontosabb modul. Ide kerül minden ami a játék működéséhez szükséges és a másik 2 modulba nem tartozik bele.

**Függvények a fomenu, jatek, gamover, és eredmeny mellett:**

**def talajgeneralas(felulet, talaj, folyamatosvagynem):** Folyamatosan mozgó talajt generál, tölt be (a mozgás ki is kapcsolható), ehhez bekéri a felületet, amin meg kell jelenítenie, a talajt amit majd meg kell jelenítenie, és bekéri azt is, hogy folyamatosan legyen a talaj vagy nem.

**def elforgatas(kep, fok):** Ennek segítségével elforgathatok egy képet így például a madarat miután meghal, ehhez bekéri a képet amit el kell forgatnia, illetve a fokot, amivel el kell forgatnia.

**def madarbetoltes(felulet, madar, jatekvege):** Ezzel jeleníti és tölti be a madarat: ha él akkor az eredeti képet, ha nem, akkor pedig a halott elforgatott képet, ehhez bekéri a felületet, amin meg kell jelenítenie, a madarat amit meg kell jelenítenie, és azt is, hogy vége-e a játéknak vagy nem.

**def jatekinditouzenet(start, felulet, uzenet):** A játék indítása előtti üzenetet tölti be, bekéri, hogy a játékos elindította-e a játékot vagy nem, emellett bekéri a felületet, amin meg kell jelenítenie, és bekéri az üzenet képét.

**def belementazoszlopba(madar, cso):** Ennek segítségével ellenőrzi, hogy a madár beleütközött-e a csövekbe vagy nem, ehhez bekéri a madarat és a csövet, hogy össze tudja őket hasonlítani.

**def halalutanfold(madar, talaj, jatekvege):** Amennyiben meghalt a madár és a levegőben volt, lefelé esik amíg nem ér a pálya aljára, ehhez bekéri a madarat, a talajt, és ezeknek a magasságát folyamatosan ellenőrzi, illetve azt is, hogy vége-e a játéknak vagy nem.

**def madarlefele(madar, talaj, madarklikk):** Ennek segítségével tart folyamatosan lefelé a madár. Ehhez bekéri a madarat, a talajt, hogy tudjuk a magasságukat, és azt is, hogy klikkelt-e közben a játékos vagy nem.

**def madartalajhozert(madar, talaj, jatekvege):** Amennyiben a madár a talajhoz ér, vége a játéknak, ehhez bekéri a madarat, a talajt, hogy tudjuk a magasságukat, és azt, hogy vége-e a játéknak vagy nem.

**def madarfelklikkutan(madar, madarklikk, jatekvege):** Amennyiben a játékos klikkelt, a madár elkezd felfelé menni, ehhez bekéri a madarat (így a magasságát), emellett, hogy klikkelt-e a játékos, illetve azt, hogy vége-e a játéknak vagy nem.

**def madarelerteaklikkmagassagot(madarklikk, jatekvege, madar, ujmadarmagassag):** Amennyiben a játékos klikkelt, és a madár felért a klikkmagassághoz, a klikket false tesszük, ehhez bekéri, hogy a játékos klikkelt-e, hogy vége-e a játéknak, bekéri a madarat, és azt is, hogy milyen magasra kell elérnie a madárnak a klikk után.

**def kimentapalyarol(madarmagassag):** Amennyiben a játékos megpróbál felül kimenni a pályáról, visszaállítja a legfelső lehetséges pozícióba, ehhez bekéri a madár jelenlegi magasságát.

**def csoszelesseg(cso, jatekvege):** Ez felel a folyamatosan mozgó csövekért, hogy balra mozogjanak amíg el nem tűnik a képernyőről. Miután eltűnt újra elindulnak a pálya jobb oldaláról. Ehhez bekéri a csövet (így a szélességét), illetve azt, hogy vége-e a játéknak.

**def csomagassag(cso, jatekvege):** Véletlenszerűen generálja az alsőcső magasságát, ehhez bekéri a csövet (így a szélességét), illetve azt, hogy vége-e a játéknak.

**def csogeneralas(felulet, cso, start, jatekvege):** Betölti a képernyőre az alsó cső-t, illetve magasságának függvényében a felső csövet is, ehhez bekéri a felületet, amin meg kell jelenítenie, a csövet, így a magasságot és a szélességet, illetve, hogy elindult-e a játék vagy nem és azt, hogy vége-e a játéknak.

**def klikkeltobjektum(objektum, event, szel, mag):** Ennek segítségével tudom ellenőrizni, hogy rákattintottak-e egy objektumra. Ehhez bekéri az objektum nevét (így megtudja a pozíciót), a Python eventjét, és az Objektum pontos szélességét és magasságát.