GAME PROGRAMMING

Zombie Attack

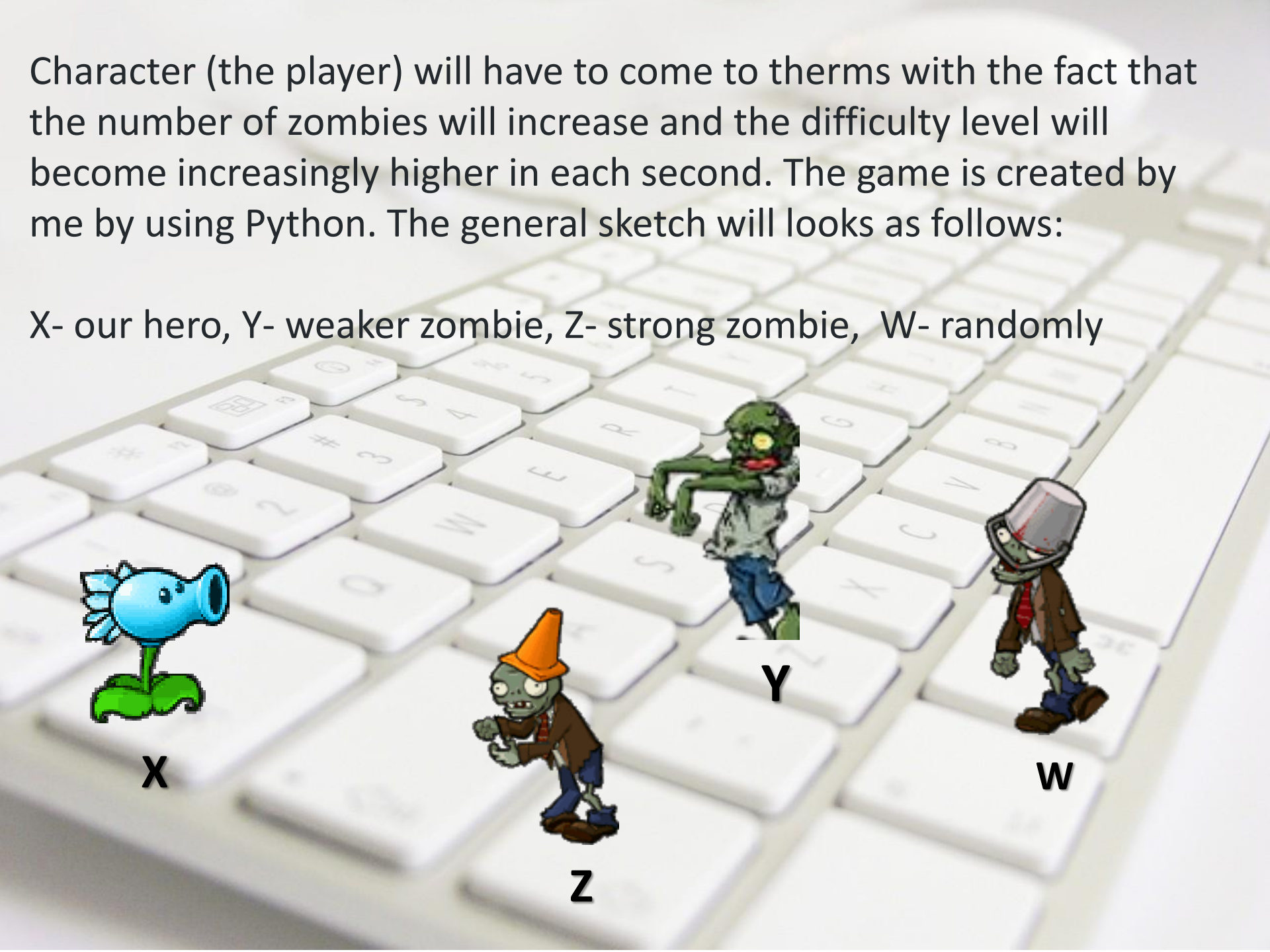
Author: Paczkowski Kamil

In this presentation I would like show in general, what is my game Zombie Attack about, what will be its assumptions and what are the main points and what is the purpose of this game. First of all Zombie Attack is dedicated for people with good manual skills with nerves of steel who are hardworking and with mobilization to keep beating new records. The main thread will be Apocalypse of zombies, monsters and other creatures as well as the fight against us, as a single hero trying to defend the world from destruction. We should strive for killing as many monsters as possible, and try to reach the highest number of points by transforming into brave soldier.

ZOMBIE ATTACK

Character (the player) will have to come to therms with the fact that the number of zombies will increase and the difficulty level will become increasingly higher in each second. The game is created by me by using Python. The general sketch will looks as follows:

X- our hero, Y- weaker zombie, Z- strong zombie, W- randomly



Analyzing this sketch we can see that the hero is attacked by two kinds of creatures: Y- weaker and slower zombies and Z- faster and more powerful zombies. For killing each of them we will receive points which during the game will be added together and sum up. The main objective is to achive as many of them as possible. It's worth to add, that with every second, the game will be more difficult by a larger crowd of zombies who will multiply much faster and also move faster. In addiction, the shooting mutant plants will be introduced to the game, as in the image below. Note: if your game line exceeds more than 10 creatures, we die.

ZOMBIE ATTACK

The atmosphere of the game will be created by moody, gloomy music and proper landscape (probably forest). Artwork will be made probably in Gimp or Photoshop, based on the designs of the other well known and popular games.

The final screen

TECHNOLOGY

Zombie Attack is created by me in high level of programming language, namely by Python. Main reason for choosing this one, not the other, was primarily the fact that this is my favourite language in which I find myself the best. In addiction, I decided to use the popular, but at the same time very effective pyGame library, which is the collection of Phytons modules to create online games using SDL libraries; pyGame works on multiple operating systems and multiple architectures. Phyton uses many brands such as: Google, Yahoo, Nokia, IBM or NASA to their worth many millions dollars applications and projects. Microsoft and Apple offer full support for Phyton in their operating systems and development platforms. Many websites such as YouTube is written in Python. Games and applications that use 3D technology you can create using Python. A common solution is to share in Phyton API game engine written in C/C++. In Phyton, we have access to several engines to support 3D graphics, 2D and other components needed for his type of application- PyGame, PyCrystal (API na CrystalSpace), Python-Ogre (API na Ogre 3D), pyopengl (API na OpenGL). I will base strictly on Phyton, without combining into this other related programming languages.


CODE

```
import pygame, random, sys, time
from pygame.locals import *

#set up some variables
WINDOWWIDTH = 1024
WINDOWHEIGHT = 600
FPS = 60

MAXGOTTENPASS = 10
ZOMBIESIZE = 70 #includes newKindZombies
ADDNEWZOMBIERATE = 30
ADDNEWKINDZOMBIE = ADDNEWZOMBIERATE

NORMALZOMBIESPEED = 2
NEWKINDZOMBIESPEED = NORMALZOMBIESPEED / 2

PLAYERMOVERATE = 15
BULLETSPEED = 10
ADDNEWBULLETRATE = 15

TEXTCOLOR = (255, 255, 255)
RED = (255, 0, 0)

def terminate():
    pygame.quit()
    sys.exit()

def waitForPlayerToPressKey():
    while True:
        for event in pygame.event.get():
            if event.type == QUIT:
                terminate()
            if event.type == KEYDOWN:
                if event.key == K_ESCAPE: # pressing escape qu
                    terminate()
                if event.key == K_RETURN:
                    return

def playerHasHitZombie(playerRect, zombies):
    for z in zombies:
        if playerRect.colliderect(z['rect']):
            return True
    return False

def bulletHasHitZombie(bullets, zombies):
    for b in bullets:
        if b['rect'].colliderect(z['rect']):
            bullets.remove(b)
            return True
    return False

def bulletHasHitCrawler(bullets, newKindZombies):
    for b in bullets:
        if b['rect'].colliderect(c['rect']):
            bullets.remove(b)
            return True
    return False

def drawText(text, font, surface, x, y):
    textobj = font.render(text, 1, TEXTCOLOR)
    textrect = textobj.get_rect()
    textrect.topleft = (x, y)
```

```
88          textrect.topleft = (x, y)
89          surface.blit(textobj, textrect)
90
91      # set up pygame, the window, and the mouse cursor
92      pygame.init()
93      mainClock = pygame.time.Clock()
94      windowSurface = pygame.display.set_mode((WINDOWWIDTH, WINDOWHEIGHT))#, pygame.FULLSCREEN)
95      pygame.display.set_caption('Zombie Defence')
96      pygame.mouse.set_visible(False)
97
98      # set up fonts
99      font = pygame.font.SysFont(None, 48)
100
101     # set up sounds
102     gameOverSound = pygame.mixer.Sound('gameover.wav')
103     pygame.mixer.music.load('grasswalk.mp3')
104
105     # set up images
106     playerImage = pygame.image.load('SnowPea.gif')
107     playerRect = playerImage.get_rect()
108
109     bulletImage = pygame.image.load('SnowPeashooterBullet.gif')
110     bulletRect = bulletImage.get_rect()
111
112     zombieImage = pygame.image.load('tree.png')
113     newKindZombieImage = pygame.image.load('ConeheadZombieAttack.gif')
114
115     backgroundImage = pygame.image.load('background.png')
116     rescaledBackground = pygame.transform.scale(backgroundImage, (WINDOWWIDTH, WINDOWHEIGHT))
```

```
        # Move the player around.
        if moveUp and playerRect.top > 30:
            playerRect.move_ip(0,-1 * PLAYERMOVERATE)
        if moveDown and playerRect.bottom < WINDOWHEIGHT-10:
            playerRect.move_ip(0,PLAYERMOVERATE)

        # Move the zombies down.
        for z in zombies:
            z['rect'].move_ip(-1*NORMALZOMBIESPEED, 0)

        # Move the newKindZombies down.
        for c in newKindZombies:
            c['rect'].move_ip(-1*NEWKINDZOMBIESPEED,0)

        # move the bullet
        for b in bullets:
            b['rect'].move_ip(1 * BULLETSPEED, 0)

        # Delete zombies that have fallen past the bottom.
        for z in zombies[:]:
            if z['rect'].left < 0:
                zombies.remove(z)
                zombiesGottenPast += 1

        # Delete newKindZombies that have fallen past the bottom.
        for c in newKindZombies[:]:
            if c['rect'].left <0:
                newKindZombies.remove(c)
                zombiesGottenPast += 1

                for b in bullets[:]:
                        if b['rect'].right>WINDOWWIDTH:
```

```
                        if b['rect'].right>WINDOWWIDTH:
                                bullets.remove(b)

        # check if the bullet has hit the zombie
        for z in zombies:
            if bulletHasHitZombie(bullets, zombies):
                score += 1
                zombies.remove(z)

        for c in newKindZombies:
            if bulletHasHitCrawler(bullets, newKindZombies):
                score += 1
                newKindZombies.remove(c)

        # Draw the game world on the window.
        windowSurface.blit(rescaledBackground, (0, 0))

        # Draw the player's rectangle, rails
        windowSurface.blit(playerImage, playerRect)

        # Draw each baddie
        for z in zombies:
            windowSurface.blit(z['surface'], z['rect'])

        for c in newKindZombies:
            windowSurface.blit(c['surface'], c['rect'])

        # draw each bullet
        for b in bullets:
            windowSurface.blit(b['surface'], b['rect'])
```

```
    # Stop the game and show the "Game Over" screen.
    pygame.mixer.music.stop()
    gameOverSound.play()
    time.sleep(1)
    if zombiesGottenPast >= MAXGOTTENPASS:
        windowSurface.blit(rescaledBackground, (0, 0))
        windowSurface.blit(playerImage, (WINDOWWIDTH / 2, WINDOWHEIGHT - 70))
        drawText('score: %s' % (score), font, windowSurface, 10, 30)
        drawText('GAME OVER', font, windowSurface, (WINDOWWIDTH / 3), (WINDOWHEIGHT / 3))
        drawText('YOUR COUNTRY HAS BEEN DESTROIED', font, windowSurface, (WINDOWWIDTH / 4)- 80, (WINDOWHEIGHT / 3) + 100)
        drawText('Press enter to play again or escape to exit', font, windowSurface, (WINDOWWIDTH / 4) - 80, (WINDOWHEIGHT / 3) + 150)
        pygame.display.update()
        waitForPlayerToPressKey()
    if playerHasHitZombie(playerRect, zombies):
        windowSurface.blit(rescaledBackground, (0, 0))
        windowSurface.blit(playerImage, (WINDOWWIDTH / 2, WINDOWHEIGHT - 70))
        drawText('score: %s' % (score), font, windowSurface, 10, 30)
        drawText('GAME OVER', font, windowSurface, (WINDOWWIDTH / 3), (WINDOWHEIGHT / 3))
        drawText('YOU HAVE BEEN KISSED BY THE ZOMMBIE', font, windowSurface, (WINDOWWIDTH / 4) - 80, (WINDOWHEIGHT / 3) +100)
        drawText('Press enter to play again or escape to exit', font, windowSurface, (WINDOWWIDTH / 4) - 80, (WINDOWHEIGHT / 3) + 150)
        pygame.display.update()
        waitForPlayerToPressKey()
    gameOverSound.stop()
```

Finally I would like to say that the game is almost ready. To start a project, I have to do a few things: show the "Start" screen, set up the start of the game, move the zombies down, move the zombies down, move the bullet, delete zombies that have fallen past the bottom. I think that the project will be finished next week.

ZOMBIE ATTACK

THE END

Bibliography:

•Make Games with Pytohn -Sean. M. Tracey

•Beginning Game Development with Python and Pygame Will McGugan

•python.rk.edu.pl

•youtube.com

Paczkowski Kamil 8076