Лабараториска вежба 6

Squirrel

1. 960x480

```
#WINWIDTH = 640 # width of the program's window, in pixels
WINWIDTH = 960 # width of the program's window, in pixels
```

2. HCAMERASLACK – horizontal, VCAMERASLACK - vertical

```
#CAMERASLACK = 90     # how far from the center the squirrel moves before moving the camera
HCAMERASLACK = 70
VCAMERASLACK = 40
MOVERATE = 9         # how fast the player moves
```

```
        if (camerax + HALF_WINWIDTH) - playerCenterx > HCAMERASLACK:
            camerax = playerCenterx + HCAMERASLACK - HALF_WINWIDTH
        elif playerCenterx - (camerax + HALF_WINWIDTH) > HCAMERASLACK:
            camerax = playerCenterx - HCAMERASLACK - HALF_WINWIDTH
        if (cameray + HALF_WINHEIGHT) - playerCentery > VCAMERASLACK:
            cameray = playerCentery + VCAMERASLACK - HALF_WINHEIGHT
        elif playerCentery - (cameray + HALF_WINHEIGHT) > VCAMERASLACK:
            cameray = playerCentery - VCAMERASLACK - HALF_WINHEIGHT

```

3.

```
# move all the squirrels
for sObj in squirrelObjs:
    # move the squirrel, and adjust for their bounce
    if sObj['pcrandom'] == 2:
        sObj['y'] += sObj['movey']
    else:
        sObj['x'] += sObj['movex']
        sObj['y'] += sObj['movey']
        sObj['bounce'] += 1
    if sObj['bounce'] > sObj['bouncerate']:
        sObj['bounce'] = 0 # reset bounce amount
```

```
366 def makeNewSquirrel(camerax, cameray):
367     sq = {}
368     generalSize = random.randint(5, 25)
369     multiplier = random.randint(1, 3)
370     sq['width'] = (generalSize + random.randint(0, 10)) * multiplier
371     sq['height'] = (generalSize + random.randint(0, 10)) * multiplier
372     sq['x'], sq['y'] = getRandomOffCameraPos(camerax, cameray, sq['width'], sq['height'])
373     sq['movex'] = getRandomVelocity()
374     sq['movey'] = getRandomVelocity()
375     sq['pcrandom'] = random.randint(1,5)
376     if sq['movex'] < 0: # squirrel is facing left
377         sq['surface'] = pygame.transform.scale(L_SQUIR_IMG, (sq['width'], sq['height']))
378     else: # squirrel is facing right
379         sq['surface'] = pygame.transform.scale(R_SQUIR_IMG, (sq['width'], sq['height']))
380 
381     sq['bounce'] = 0
382     sq['bouncerate'] = random.randint(10, 18)
383     sq['bounceheight'] = random.randint(10, 50)
384     return sq
385 
```

- маркирање на squirrels од 1 до 5 random, кога squirrel со број 2 ќе се генерира движењето е само вертикално.

3.

```
27  WINSIZE = 300        # how big the player needs to be to win
28  LOSESIZE = 5
```

```
            if playerObj['size'] > WINSIZE:
                winMode = True # turn on "win mode"
                gameOverMode = True  # turn on "game over mode"
                gameOverStartTime = time.time()


        elif not invulnerableMode:
            # player is smaller and takes damage
            invulnerableMode = True
            invulnerableStartTime = time.time()
            playerObj['size'] -= 1
            if playerObj['size'] <= LOSESIZE:
                gameOverMode = True # turn on "game over mode"
                gameOverStartTime = time.time()
```

- Користејќи ја логиката за губење на живот сега вашиот squirrel при удар со поголем не губи живот туку истиот се намалува на големина. Кога големината ќе достигне 5 (LOSESIZE) играта завршува(gameover mode = on).