

Лабораториска 4 – ПНВИ

Фросина Цветковска 191216

1. 20 seconds after starting the game add a second worm. The original worm should avoid the second one. If the original worm touches the second worm with the head, its body grows for one segments. If the second worm touches the original, than its body grows for one segment. The movement of the second worm is random.

Доаваме виолетова боја за вториот црв кој треба да биде креиран. Додаваме функција која што ќе ја повикаме при креирање на црвот

```
302
303 #барање 1 креирање на нов црв
304 def drawSecondWorm(wormCoords):
305     for coord in wormCoords:
306         x = coord['x'] * CELLSIZE
307         y = coord['y'] * CELLSIZE
308         wormSegmentRect = pygame.Rect(x, y, CELLSIZE, CELLSIZE)
309         pygame.draw.rect(DISPLAYSURF, DARKPURPLE, wormSegmentRect)
310         wormInnerSegmentRect = pygame.Rect(x + 4, y + 4, CELLSIZE - 8, CELLSIZE - 8)
311         pygame.draw.rect(DISPLAYSURF, PURPLE, wormInnerSegmentRect)
312
313 #барање 2: креирање на розево и сино јаболко
314 def drawPinkApple(coord):
```

```
19 #барање 1 креирање на нов црв
20 WHITE = (255, 255, 255)
21 BLACK = ( 0, 0, 0)
22 RED = (255, 0, 0)
23 GREEN = ( 0, 255, 0)
24 DARKGREEN = ( 0, 155, 0)
25 DARKGRAY = ( 40, 40, 40)
26 #барање 1 и 2 додавање на нови бои
27 PURPLE = (128, 0, 128)
28 DARKPURPLE = ( 48, 25, 52)
29 PINK = ( 48, 25, 52)
30 BLUE = ( 0, 0, 255)
31 BGCOLOR = BLACK
32
```

Потоа во runGame() додаваме нови координати за вториот црв, рандом позиција на која што ќе се движи, неговото појавување на 20 секунди од играта и потоа на крајот го исцртуваме црвот и неговите координати се додаваат во променливата.

```

64     direction = RIGHT
65     #барање 1 додавање на координати
66     startx = random.randint(5, CELLWIDTH - 6)
67     starty = random.randint(5, CELLHEIGHT - 6)
68     secondWormCoords = [{'x': startx, 'y': starty},
69                          {'x': startx - 1, 'y': starty},
70                          {'x': startx - 2, 'y': starty}]
71     secondDirection = LEFT
72     # Start the apple in a random place.
73     # Start the apple in a random place.

```

```

119     wormCoords.insert(0, newHead)
120
121     #барање 1 random direction
122     if secondDirection == LEFT:
123         secondDirection = random.choice([UP, DOWN, LEFT])
124     elif secondDirection == RIGHT:
125         secondDirection = random.choice([UP, DOWN, RIGHT])
126     elif secondDirection == UP:
127         secondDirection = random.choice([RIGHT, LEFT, UP])
128     elif secondDirection == DOWN:
129         secondDirection = random.choice([RIGHT, DOWN, LEFT])
130
131     if((time.time() - start) > 20):
132         if secondWormCoords[HEAD]['x'] == wormCoords[HEAD]['x'] and secondDirection[HEAD]['y'] == wormCoords[HEAD]['x']:
133             pass
134         else:
135             del secondWormCoords[-1]
136
137     #барање 1 движење на вториот црв
138     if secondDirection == UP:
139         newHead1 = {'x': secondWormCoords[HEAD]['x'], 'y': secondWormCoords[HEAD]['y'] - 1}
140     elif secondDirection == DOWN:
141         newHead1 = {'x': secondWormCoords[HEAD]['x'], 'y': secondWormCoords[HEAD]['y'] + 1}
142     elif secondDirection == LEFT:
143         newHead1 = {'x': secondWormCoords[HEAD]['x'] - 1, 'y': secondWormCoords[HEAD]['y']}
144     elif secondDirection == RIGHT:
145         newHead1 = {'x': secondWormCoords[HEAD]['x'] + 1, 'y': secondWormCoords[HEAD]['y']}
146     secondWormCoords.insert(0, newHead1)
147     DISPLAYSURF.fill(BG_COLOR)
148     drawGrid()

```

```

146     secondWormCoords.insert(0, newHead1)
147     DISPLAYSURF.fill(BG_COLOR)
148     drawGrid()
149     drawWorm(wormCoords)
150     if((time.time() - start) > 20):
151         secondWormCoords.insert(0, newHead1)
152         drawSecondWorm(secondWormCoords)
153     #барање 2 се појавува сино јаболко секои 7 секунди
154     if((time.time() - start) < 7):
155         drawBlueApple(blueApple)
156     #барање 2 се појавува розево јаболко 5 секунди секои 5 секунди
157     if((time.time() - start) % 10 > 5):
158         drawPinkApple(pinkApple)
159     drawApple(apple)
160     #барање 2 нова формула за пресметување на резултатот
161     score = (len(wormCoords) - 3) + extra
162     drawScore(score)
163     pygame.display.update()
164     FPSLOCK.tick(FPS)
165
166     def drawPressKeyMsg():

```

2. Add 2 elements that blink (3 each time) at randomly selected positions with dimensions of 1 cell. The first one appears every 5 seconds, and it lasts 5 seconds. The second one appears only once, lasting 7 seconds. If the original worm eats any of these elements, the player gets additional points (3 for each eaten element). These points should be included in the end result in a way that you will choose. You

need to provide an explanation for the formula that you will use for calculating the result. The result should be shown on the screen after the game ends.

Додавање две нови јаболка розево и сино кое што розевото ќе се прикажува на секои 7 секунди а розевото на секои 5 секунди ќе се покажува 5 секунди во играта.

Креираме две нови функции за исцртување на самите две јаболка и две нови бои сина и розева и променлива за екстра поени:

```
313 #барање 2: креирање на розево и сино јаболко
314 def drawPinkApple(coord):
315     x = coord['x'] * CELL_SIZE
316     y = coord['y'] * CELL_SIZE
317     for i in range(3):
318         appleRect = pygame.Rect(x, y, CELL_SIZE, CELL_SIZE)
319         pygame.draw.rect(DISPLAYSURF, BLUE, appleRect)
320         pygame.display.update()
321
322 def drawBlueApple(coord):
323     x = coord['x'] * CELL_SIZE
324     y = coord['y'] * CELL_SIZE
325     for i in range(3):
326         appleRect = pygame.Rect(x, y, CELL_SIZE, CELL_SIZE)
327         pygame.draw.rect(DISPLAYSURF, PINK, appleRect)
328         pygame.display.update()
329
```

```
25 DARKGRAY = (40, 40, 40)
26 #барање 1 и 2 додавање на нови бои
27 PURPLE = (128, 0, 128)
28 DARKPURPLE = (48, 25, 52)
29 PINK = (48, 25, 52)
30 BLUE = (0, 0, 255)
31 BG_COLOR = BLACK
32
33 UP = 'up'
34 DOWN = 'down'
35 LEFT = 'left'
36 RIGHT = 'right'
37
38 HEAD = 0 # syntactic sugar: index of the worm's head
39 score = 0
40
```

Во runGame() функцијата прво додаваме глобална променлива score која што ќе се користи за чување на самиот резултат и променливи за јаболката и екстра поени

```

54
55
56 def runGame():
57     global secondDirection, score
58     # Set a random start point.
59     startx = random.randint(5, CELLWIDTH - 6)
60     starty = random.randint(5, CELLHEIGHT - 6)
61     wormCoords = [{'x': startx, 'y': starty},
62                   {'x': startx - 1, 'y': starty},
63                   {'x': startx - 2, 'y': starty}]

```

```

70     {'x': startx - 2, 'y': starty}]
71     secondDirection = LEFT
72     # Start the apple in a random place.
73     apple = getRandomLocation()
74     #барање 2 додавање на вредности за розево, сино јаболко и екстра поени
75     pinkApple = getRandomLocation()
76     blueApple = getRandomLocation()
77     extra = 0
78     start = time.time()
79
80

```

Потоа во согласност со времето на нивното покажување ги исцртуваме објектите и додаваме екстра поени на самата формула за пресметување на поени

```

drawWorm(wormCoords)
if((time.time() - start) > 20):
    secondWormCoords.insert(0, newHead1)
    drawSecondWorm(secondWormCoords)
#барање 2 се појавува сино јаболко секои 7 секунди
if((time.time() - start) < 7):
    drawBlueApple(blueApple)
#барање 2 се појавува розево јаболко 5 секунди секои 5 секунди
if((time.time() - start) % 10 > 5):
    drawPinkApple(pinkApple)
drawApple(apple)
#барање 2 нова формула за пресметување на резултатот
score = (len(wormCoords) - 3) + extra
drawScore(score)
pygame.display.update()
FPSLOCK.tick(FPS)

```

На крајот го печатиме резултатот на крајот на играта во функцијата showGameOverScreen()

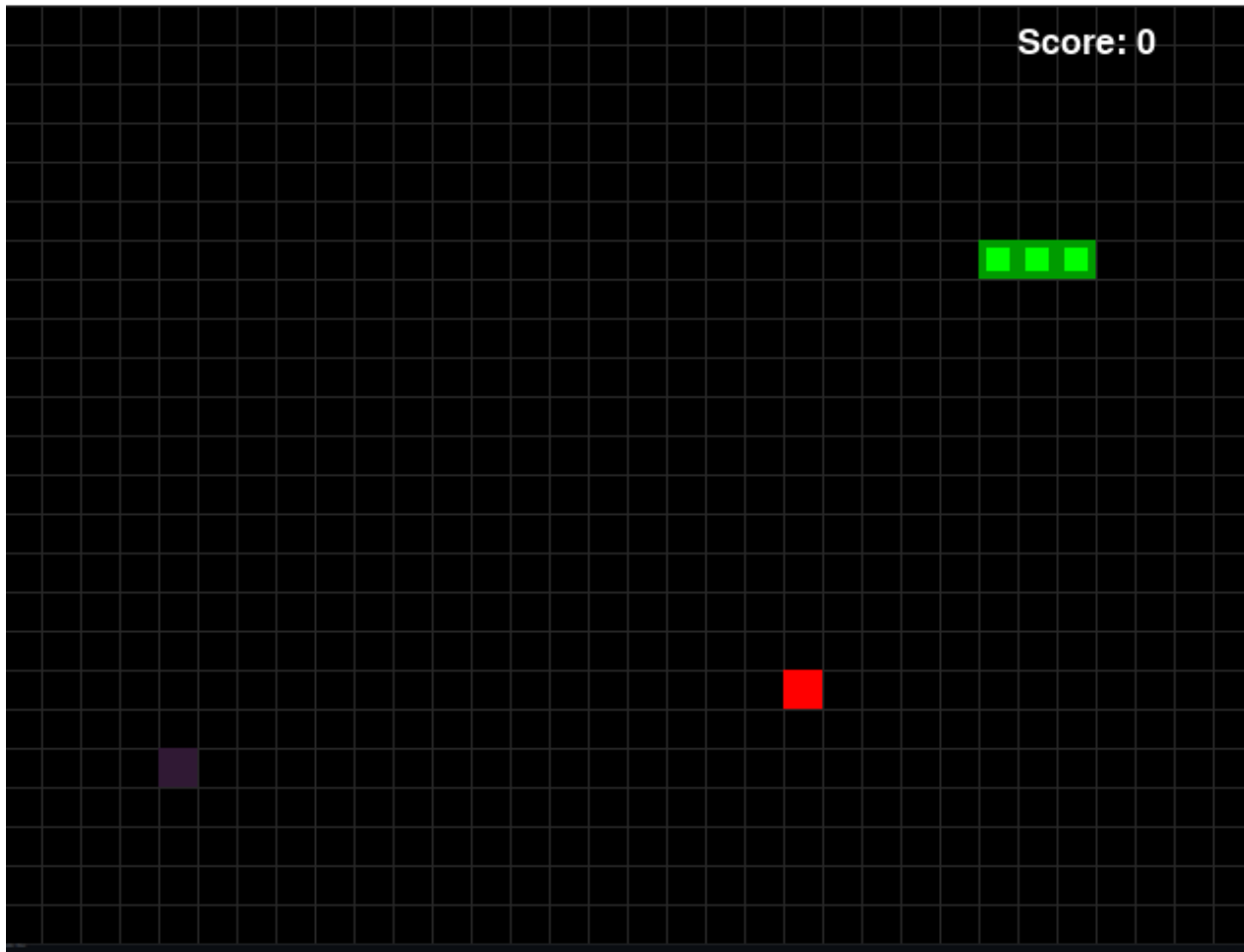
```

223
224 ✓ def showGameOverScreen():
225     global score
226     gameOverFont = pygame.font.Font('freesansbold.ttf', 150)
227     gameSurf = gameOverFont.render('Game', True, WHITE)
228     overSurf = gameOverFont.render('Over', True, WHITE)
229     gameRect = gameSurf.get_rect()
230     overRect = overSurf.get_rect()
231     gameRect.midtop = (WINDOWWIDTH / 2, 10)
232     overRect.midtop = (WINDOWWIDTH / 2, gameRect.height + 10 + 25)
233
234     DISPLAYSURF.blit(gameSurf, gameRect)
235     DISPLAYSURF.blit(overSurf, overRect)
236
237     #барање 2: резултат на gameOver екран
238     scoreFont = pygame.font.Font('freesansbold.ttf', 16)
239     scoreSurf = scoreFont.render('Score: ' + str(score), True, WHITE)
240     scoreRect = scoreSurf.get_rect()
241     scoreRect.midtop = (WINDOWWIDTH / 2, 450)
242     DISPLAYSURF.blit(scoreSurf, scoreRect)
243

```

Wormy

Score: 0

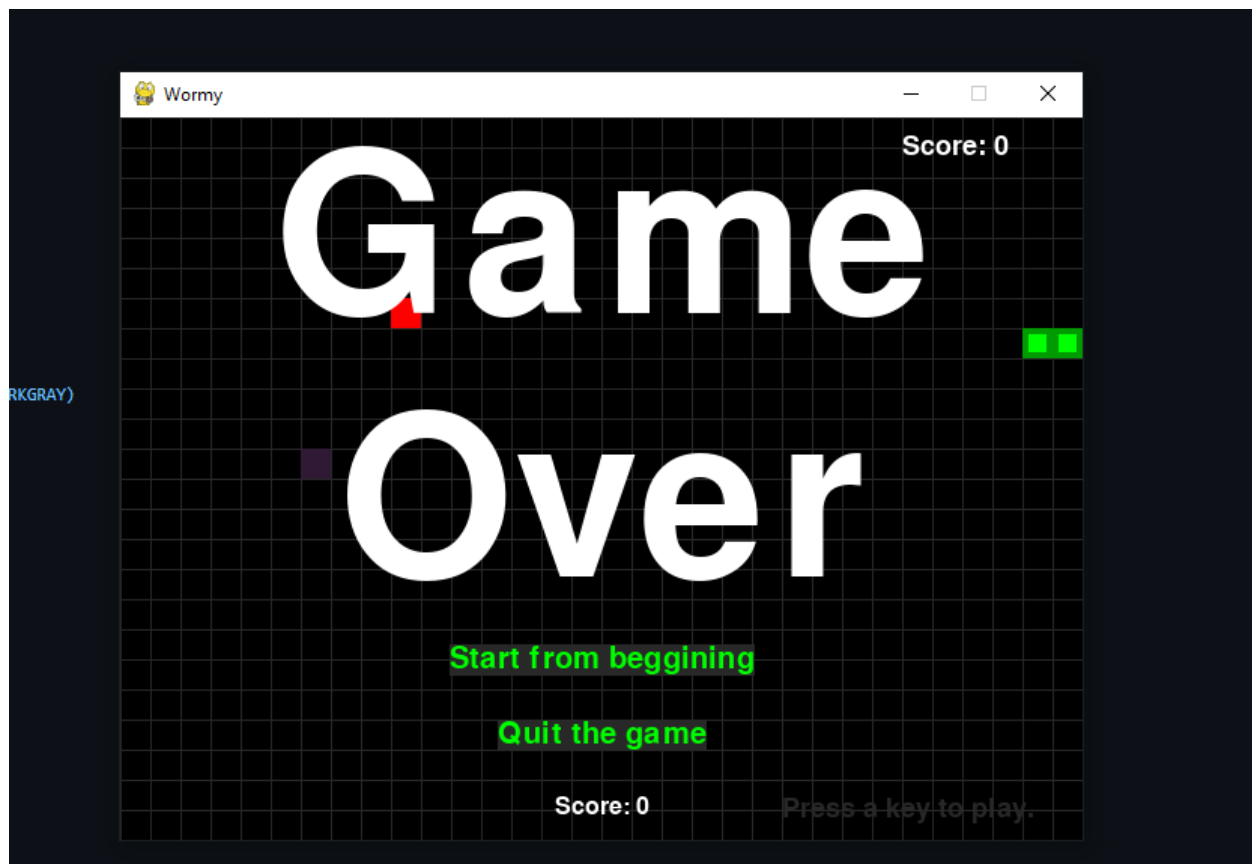




3. Add two buttons "Start from the beginning" and "Quit" to the game over screen. When the user clicks the first button, the game should start over (without showing the starting screen). When the player clicks the second button, the game should terminate.

На почетокот во функцијата `showGameOverScreen` додаваме две нови копчиња за старт од почеток и за крај на играта со зелена боја и темно сива позадина (246-254)

Потоа со цел истите да се прикажуваат се генерира функција која што при клик на start копчето почнува нова игра а quit ја терминира играта (266-271)



```
244
245 #барање 3 додавање на копчиња за старт и крај
246 buttonFont = pygame.font.Font('freesansbold.ttf', 20)
247 startSurf = buttonFont.render('Start from beggining', True, GREEN, DARKGRAY)
248 quitSurf = buttonFont.render('Quit the game', True, GREEN, DARKGRAY)
249 startRect = startSurf.get_rect()
250 quitRect = quitSurf.get_rect()
251 startRect.midtop = (WINDOWWIDTH / 2, 350)
252 quitRect.midtop = (WINDOWWIDTH / 2, 400)
253 DISPLAYSURF.blit(startSurf, startRect)
254 DISPLAYSURF.blit(quitSurf, quitRect)
255
256 drawPressKeyMsg()
257 pygame.display.update()
258 pygame.time.wait(500)
259 checkForKeyPress() # clear out any key presses in the event queue
260
261 while True:
262     if checkForKeyPress():
263         pygame.event.get() # clear event queue
264         return
265     #барање 3 функции за копчињата
266     for event in pygame.event.get():
267         if event.type == MOUSEBUTTONDOWN:
268             if startRect.collidepoint(pygame.mouse.get_pos()):
269                 return
270             if quitRect.collidepoint(pygame.mouse.get_pos()):
271                 terminate()
272
```