

SlidePuzzle работа на час

Слика од почетниот екран на играта



1. *Decrease the amount of time the player has to push the next button in the sequence every 7th time the pattern is repeated successfully (start from 5 and decrease by 1).:*

Променливата TIMEOUT што имаше улога на константа ја поставуваме како локална променлива во main функцијата и ја иницијализираме на вредност 5.

```
TIMEOUT = 5
```

Во main loop во if делот кој што се извршува кога играчот ја погодил секвенцата додаваме код кој што ја намалува променливата TIMEOUT за 1 доколку суквенцата е делива со 7 и TIMEOUT е поголем од 1.

```
if len(pattern) % 7 == 0 and TIMEOUT > 1:
```

```
    TIMEOUT -= 1
```

Дополнително во делот што се извршува кога играчот ќе ја промаши секвенцата односно во делот кога играта треба да се припреми за нов почеток вредноста на TIMEOUT повторно се посравува на 5.

```
    TIMEOUT = 5 # starts from 5sec
```

2. *Change the pattern list so that the new random color will be added to the beginning of the list, rather than at the end of the list.:*

Во main loop во делот кој што се извршува кога „Компјутерот е на потег“ односно во (*if not waitingForInput:*) кога се додава нова боја во секвенцата наместо да ја додаваме

на крај со функцијата `append()` се користи функцијата `insert` со со параметар `index=0` и со тоа се постигнува додавање на нова боја на почеток на секвенцата.

```
pattern.insert(0, (random.choice((YELLOW, BLUE, RED, GREEN))))  
# Insert at the start of the pattern
```

Дополнителни промени во кодот не се потребни.

3. *Change the number of flashes of the game over animation.:*

Во функцијата за анимација при крај на игра `gameOverAnimation` for циклусот наместо да се повторува 3 пати ќе смениме да се повторува 5 пати и со тоа бројот на светкања (flashes) на екранот ќе го промениме од 3 на 5

```
def gameOverAnimation(color=WHITE, animationSpeed=50):  
...  
    for i in range(5): # Now we will do the flash 5 times  
...  
...
```

4. *Change the shades of color used for the buttons.:*

За да ги промениме нијансите на боите само ќе ги промениме вредностите на променливите односно константите кои ги означуваат боите

```
BRIGHTRED = (255, 20, 30)  
RED = (170, 10, 15)  
BRIGHTGREEN = (20, 255, 40)  
GREEN = (10, 160, 20)  
BRIGHTBLUE = (20, 20, 255)  
BLUE = (10, 10, 135)  
BRIGHTYELLOW = (255, 255, 10)  
YELLOW = (120, 120, 5)
```

5. *Increase the number of buttons and update the game properly. Regarding the music files, use the available four files.:*

Бројот на копчиња ќе го зголемиме од 4 на 6. Најпрвин ќе додадеме нови константи кој ќе ги репрезентираат боите

```
BRIGHTPINK = (255, 0, 255)  
PINK = (150, 0, 150)  
BRIGHTCYAN = (0, 255, 255)  
CYAN = (0, 140, 140)
```

Потоа ќе ја прилагодиме X маргината на игра со 3 копчиња во еден ред

```
XMARGIN = int((WINDOWWIDTH - (3 * BUTTONSIZE) - 2*BUTTONGAPSIZE) / 2)
```

Потоа ќе ги креираме ограничувачките правоаголници на копчињата

```
PINKRECT = pygame.Rect(XMARGIN + 2 * BUTTONSIZE + 2 * BUTTONGAPSIZE,  
YMARGIN, BUTTONSIZE, BUTTONSIZE)
```

```
CYANRECT = pygame.Rect(XMARGIN + 2 * BUTTONSIZE + 2 * BUTTONGAPSIZE,  
YMARGIN + BUTTONSIZE + BUTTONGAPSIZE, BUTTONSIZE, BUTTONSIZE)
```

Следно во main loop во делот каде се регистрираа кликување на тастер од тастатурата додаваме код кој клик на копчето Е ќе означува клик на копчето со розева боја и клик на копчето Д ќе означува клик на копчето со цијан боја

```
elif event.key == K_e:  
    clickedButton = PINK  
  
elif event.key == K_d:  
    clickedButton = CYAN
```

Следно во main loop во делот каде компјутерот е на потег и кога се додава случајна нова боја во секвенцата ќе го смениме кодот со цел да се може да се додадат и ново креираните бои розева и цијан

```
pattern.insert(0, (random.choice((YELLOW, BLUE, RED, GREEN, PINK, CYAN))))
```

Сега во функцијата *flashButtonAnimation* која прави анимација на трепкање на копчињата додаваме код кој ќе овозможи и новите копчиња да трепкаат

```
elif color == PINK:  
    sound = BEEP2  
    flashColor = BRIGHTPINK  
    rectangle = PINKRECT  
elif color == CYAN:  
    sound = BEEP1  
    flashColor = BRIGHTCYAN  
    rectangle = CYANRECT
```

Следно во функцијата *drawButtons* која ги исцртува копчињата додаваме код за да ги исцрта и новите копчиња

```
pygame.draw.rect(DISPLAYSURF, PINK, PINKRECT)
pygame.draw.rect(DISPLAYSURF, CYAN, CYANRECT)
```

И последно во функцијата *getButtonClicked* која проверува дали некое копче е кликнато додаваме код кој ги вклучува и новите копчиња

```
elif PINKRECT.collidepoint((x, y)):
    return PINK
elif CYANRECT.collidepoint((x, y)):
    return CYAN
```

6. *Change the size of the four buttons.*

За да го постигнеме зголемувањето на копчињата треба само да ја зголемиме вредноста на променливата `BUTTONSIZE`

```
BUTTONSIZE = 260
BUTTONGAPSIZE = 30
```

Со цел правилно функционирање на играта и соодветен преиказ на екран ќе ги зголемиме ширината и висината на прозорецот какоми просторот помеѓу копчињата `BUTTONGAPSIZE`.

```
WINDOWWIDTH = 960
WINDOWHEIGHT = 720
```

7. *Прикачете документ со кратко објаснување за секое од прашањата*

8. *Change the amount of time the player has to push the next button in the sequence before the game times out.*

Временската рамка во која корисникот треба да кликне на следно копче може да се промени со менувње на вредноста на променливата `TIMEOUT`. Ова и плус други дополнителни промени во однос на оваа временска рамка направивме во барањето 1.

9. *Make a change of your choice:*

Сакаме освен информации освен за моменталниот *score* да прикажуваме информации и за најдобриот резултат односно *highscore*. За оваа цел воведуваме нова променлива `highscore` и ја иницијализираме на 0

```
highscore = 0
```

Потоа во `main loop` додаваме код кој ќе го прикажува најдобриот резултат на екран

```
highscoreSurf = BASICFONT.render('HighScore: ' + str(highscore), 1, WHITE)
highscoreRect = highscoreSurf.get_rect()
```

```
highscoreRect.topleft = (WINDOWWIDTH - 135, 30)  
DISPLAYSURF.blit(highscoreSurf, highscoreRect)
```

Во делот каде се проверува дали играчот ја погодил секвенцата додаваме код кој што ќе ја ажурира вредноста на *highscore*

```
if score >= highscore:  
    highscore = score
```