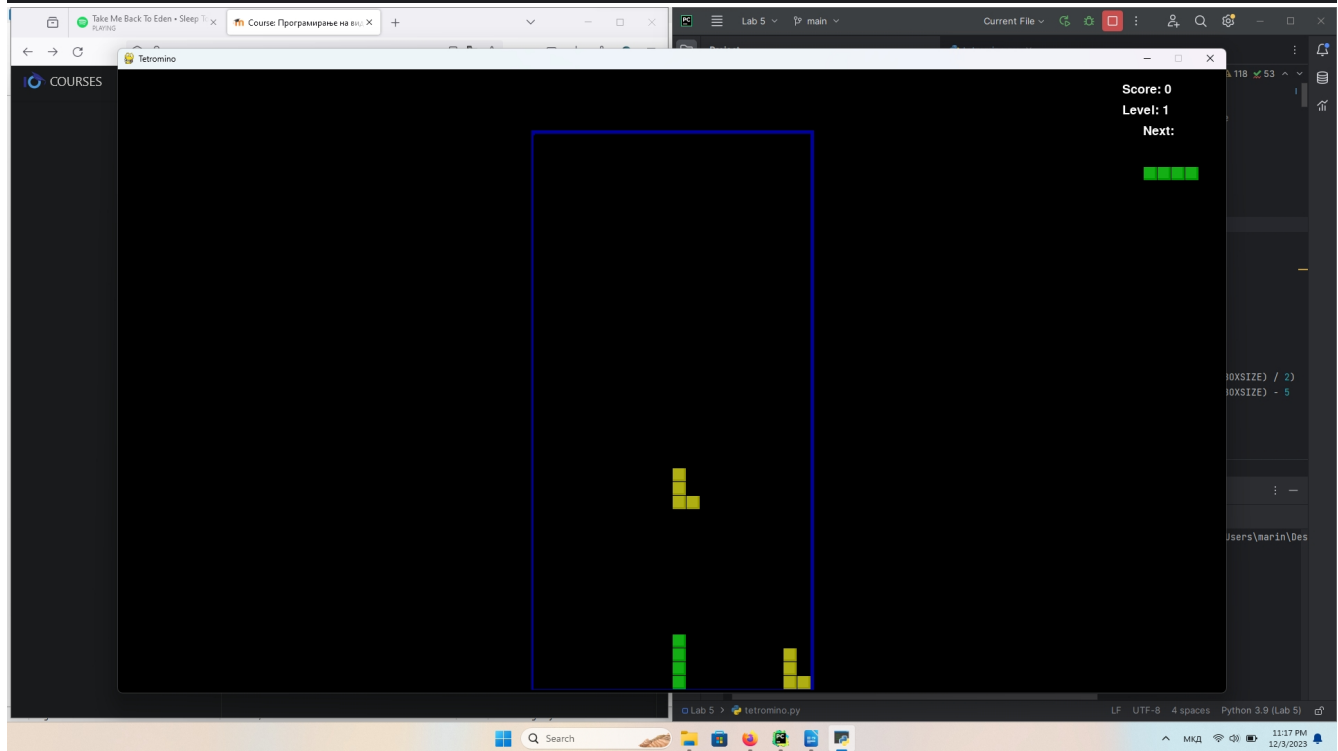


Прво барање:

Ги променив димензиите на прозорецот за да се акомодираат зголемените димензии на board-от.

```
WINDOWWIDTH = 1600
WINDOWHEIGHT = 900
BOARDWIDTH = 20
BOARDHEIGHT = 40
```



Второ барање:

Додадов три нови листи од листи за формите L, J, и I. Само тие три можат да се нацртаат со 3 коцки. Додадов три нови записи во речникот во кој се чуваат сите можни форми.

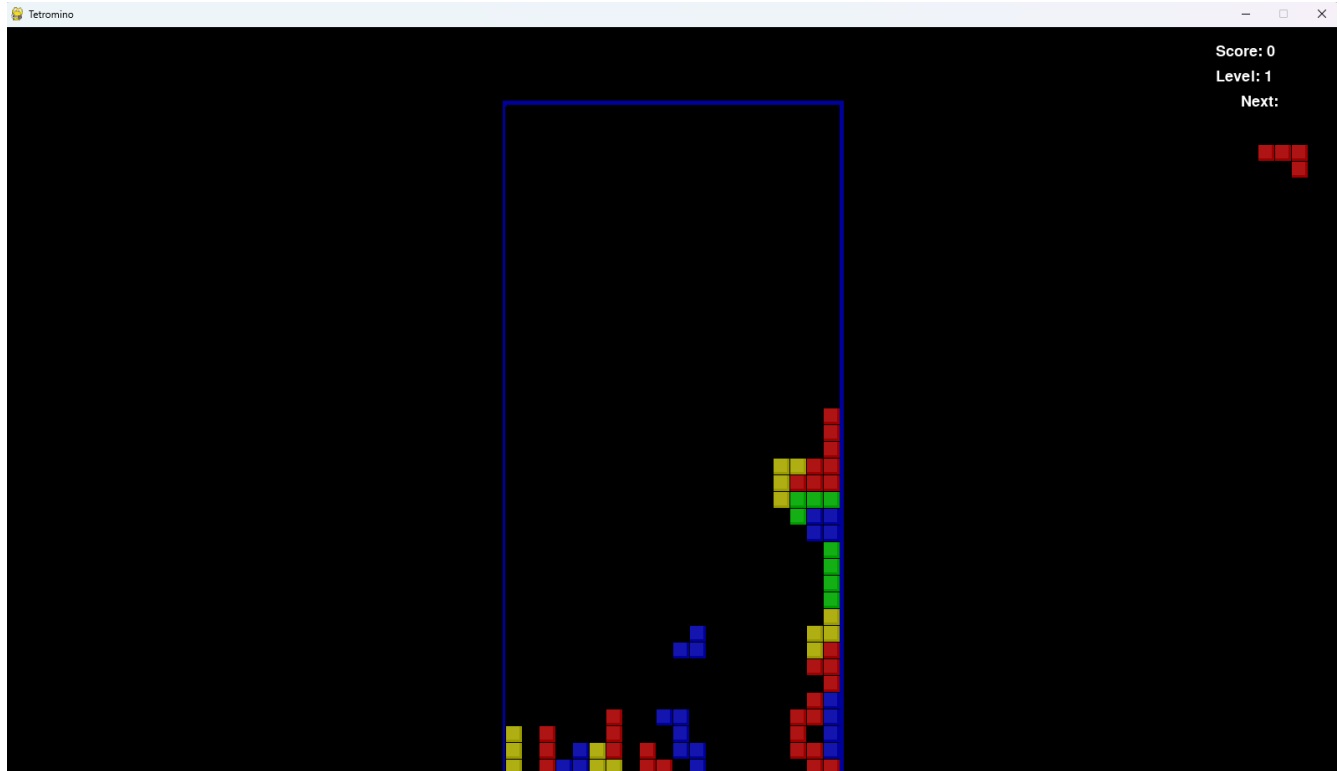
```
THREE_L_SHAPE_TEMPLATE = [['.....',  
    '..O.',  
    '..OO.',  
    '.....',  
    '.....'],  
    ['.....',  
    '.....',  
    '..O.',  
    '..OO.',  
    '.....'],  
    ['.....',  
    '.....',  
    'OO..',  
    'O..',  
    '.....'],  
    ['.....',  
    'OO..',  
    '..O.',  
    '.....',  
    '.....']]
```

```
THREE_J_SHAPE_TEMPLATE = [['.....',  
    'O..',  
    'OO..',  
    '.....',  
    '.....'],  
    ['.....',  
    'OO.',  
    '..O.',  
    '.....',  
    '.....'],  
    ['.....',  
    '.....',  
    'OO.',  
    'O..',  
    '.....'],  
    ['.....',  
    '.....',  
    'O..',  
    'OO..',  
    '.....']]
```

```
THREE_I_SHAPE_TEMPLATE = [['..O.',  
    '..O.',  
    '..O.',  
    '.....'],  
    ['.....',  
    '.....',  
    'OOO..',  
    '.....',  
    '.....']]
```

```
PIECES = {'S': S_SHAPE_TEMPLATE,  
    'Z': Z_SHAPE_TEMPLATE,  
    'J': J_SHAPE_TEMPLATE,
```

```
'L': L_SHAPE_TEMPLATE,  
'I': I_SHAPE_TEMPLATE,  
'O': O_SHAPE_TEMPLATE,  
'T': T_SHAPE_TEMPLATE,  
'THREE J': THREE_J_SHAPE_TEMPLATE,  
'THREE L': THREE_L_SHAPE_TEMPLATE,  
'THREE I': THREE_I_SHAPE_TEMPLATE}
```



Трето барање:

За жал не можам за овој дел да направам слика бидејќи ќе треба да стигнам до 23 поени, а не сум доволно добар во тетрис.

Воведов три нови глобални променливи кои служат за чување на моментално ниво, чување на поени потребни за следно ниво, и чување за колку поени да се зголеμουва next level requirement. Променета е и логиката во функцијата за пресметка на поени и брзина на паѓање.

```
current_level = 1
next_level_requirement = 10
level_increment = 3

def calculateLevelAndFallFreq(score):
    global current_level, next_level_requirement, level_increment

    if score >= next_level_requirement:
        current_level += 1
        next_level_requirement += level_increment

    fallFreq = 0.27 - (current_level * 0.02)
    return current_level, fallFreq
```

Четврто барање:

За ова барање воведов променлива `completed_lines` која чува колку линии биле комплетирани. Вредноста во оваа променлива се добива од повикот на функцијата `removeCompletedLines(board)`, и притоа оваа функција не се повикува кај `score += ...`, туку сега тука стои новата променлива.

Потоа по добивање на `fallFreq` од соодветната функција која го прави тоа, се проверува дали воопшто биле комплетирани линии. Доколку да, се проверува парноста на тој број, и соодветно се зголемува/намалува `fallFreq` за коефициентот `level*0.02`.

```
completed_lines = removeCompleteLines(board)
score += completed_lines
level, fallFreq = calculateLevelAndFallFreq(score)
if completed_lines > 0:
    if completed_lines % 2 == 0:
        fallFreq = fallFreq + (level * 0.02)
    else:
        fallFreq = fallFreq - (level * 0.02)
```