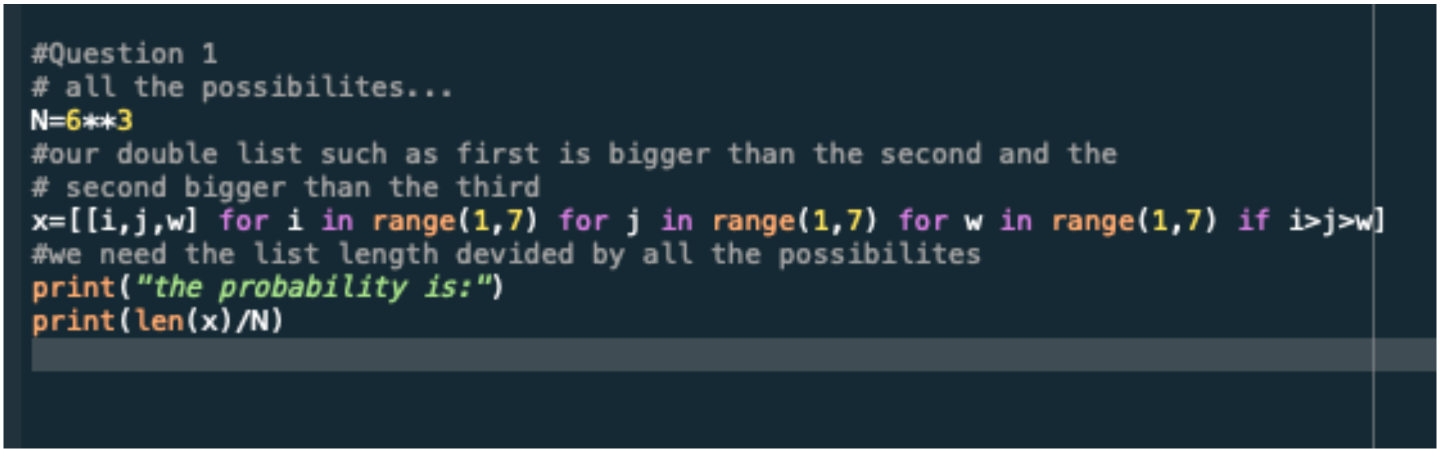
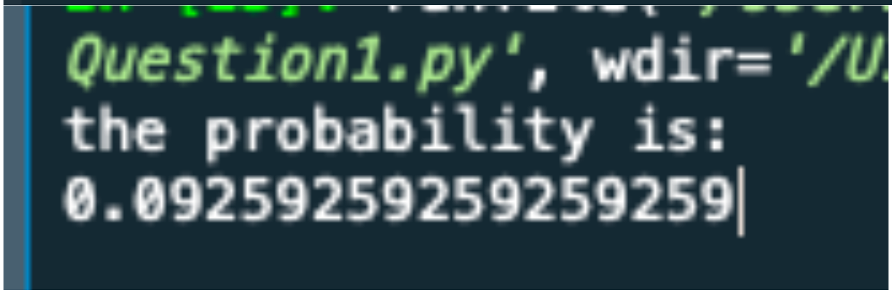
מבוא לתכנות מדעי פיתון
תרגיל בית 8
שותפים: נימר נסייר 322626896
פאטמה נעמה 212100580

שאלה 1:

Link: https://py3.codeskulptor.org/#user306_PrU0tOM9n4_1.py



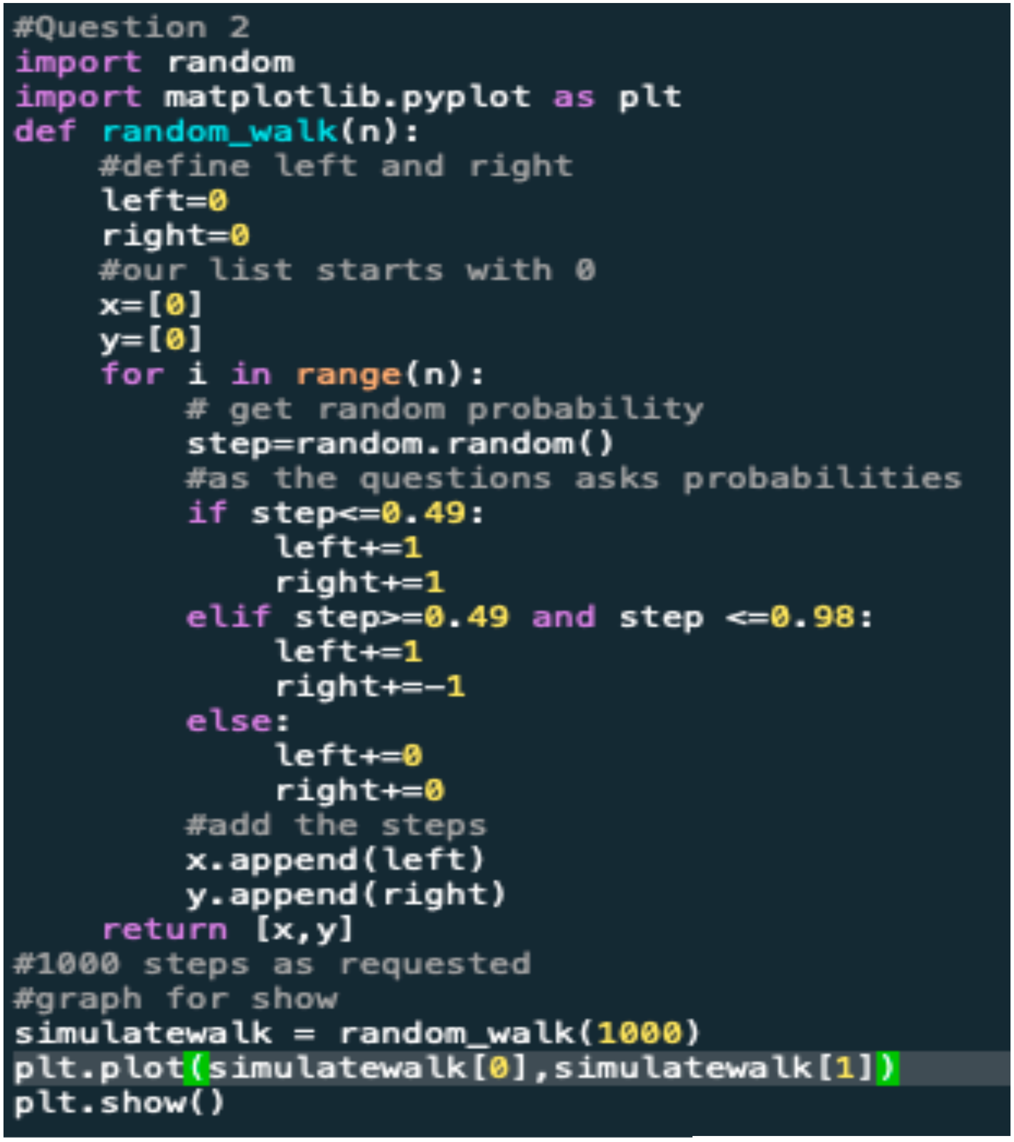
```
#Question 1
# all the possibilites...
N=6**3
#our double list such as first is bigger than the second and the
# second bigger than the third
x=[[i,j,w] for i in range(1,7) for j in range(1,7) for w in range(1,7) if i>j>w]
#we need the list length devided by all the possibilites
print("the probability is:")
print(len(x)/N)
```



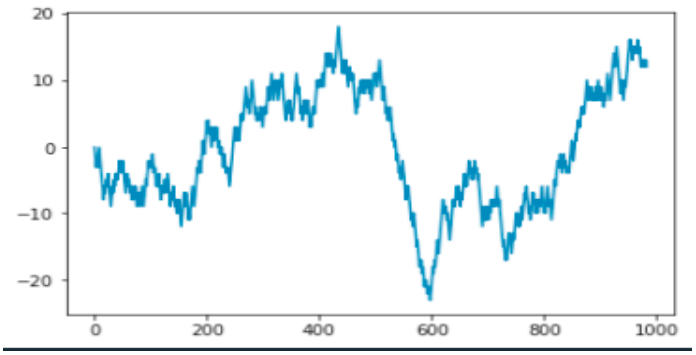
```
Question1.py', wdir='/U
the probability is:
0.09259259259259259
```

שאלה 2:

Link: https://py3.codeskulptor.org/#user306_PrU0tOM9n4_2.py



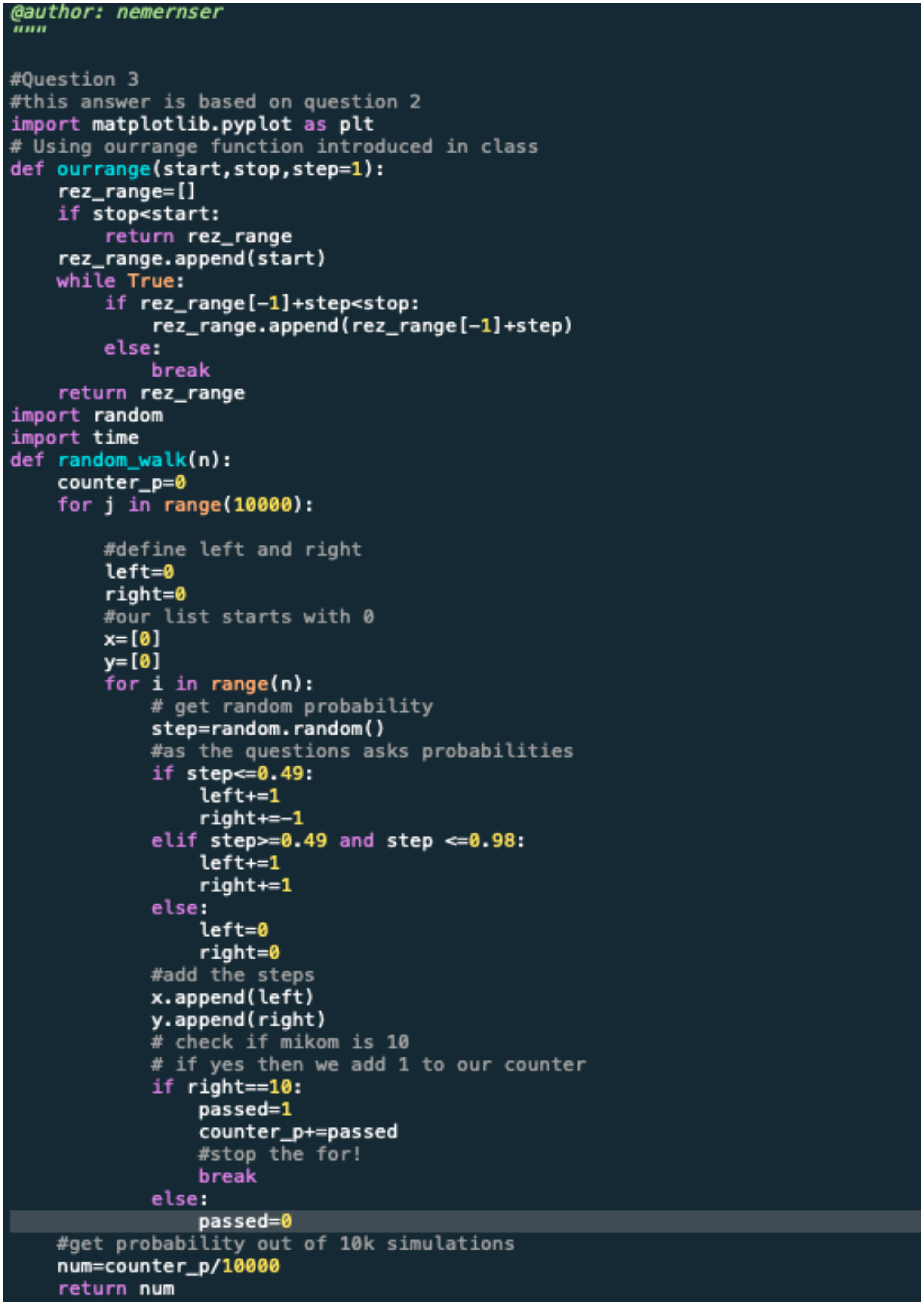
```
#Question 2
import random
import matplotlib.pyplot as plt
def random_walk(n):
    #define left and right
    left=0
    right=0
    #our list starts with 0
    x=[0]
    y=[0]
    for i in range(n):
        # get random probability
        step=random.random()
        #as the questions asks probabilities
        if step<=0.49:
            left+=1
            right+=1
        elif step>=0.49 and step <=0.98:
            left+=1
            right+=-1
        else:
            left+=0
            right+=0
        #add the steps
        x.append(left)
        y.append(right)
    return [x,y]
#1000 steps as requested
#graph for show
simulatewalk = random_walk(1000)
plt.plot(simulatewalk[0],simulatewalk[1])
plt.show()
```



שאלה 3:

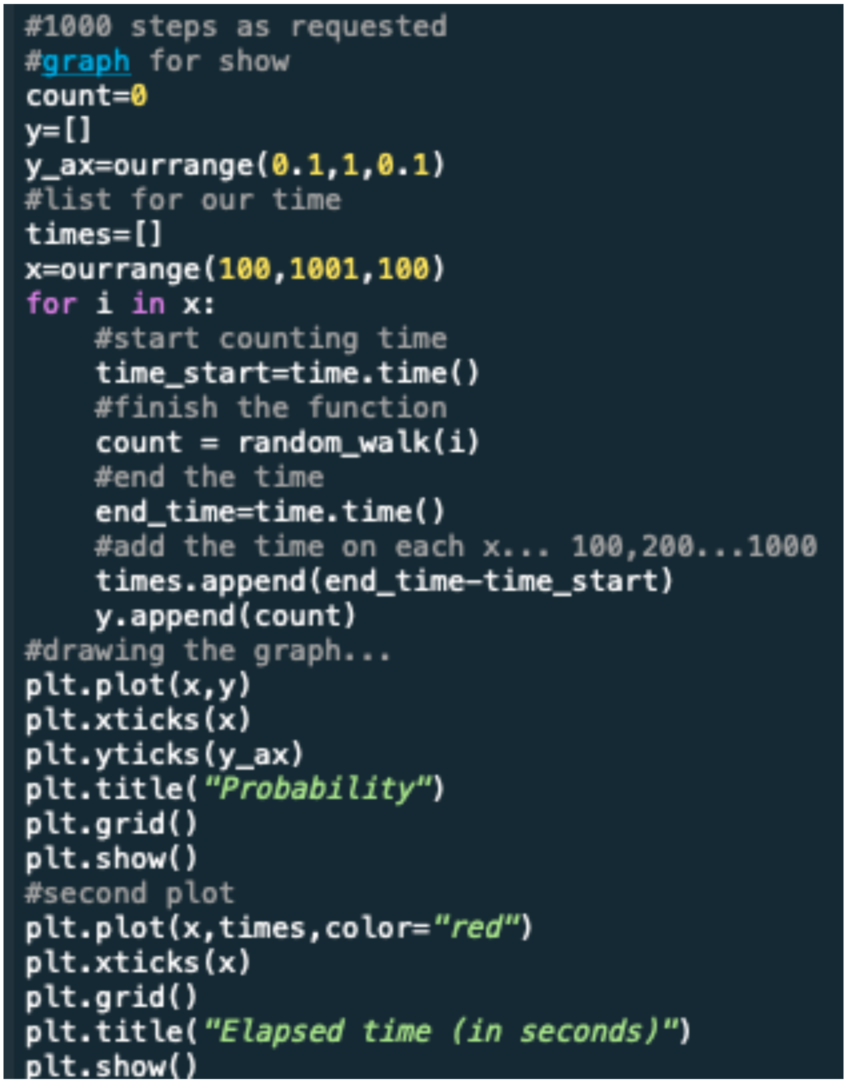
Link: https://py3.codeskulptor.org/#user306_PrU0tOM9n4_3.py

Part 1:



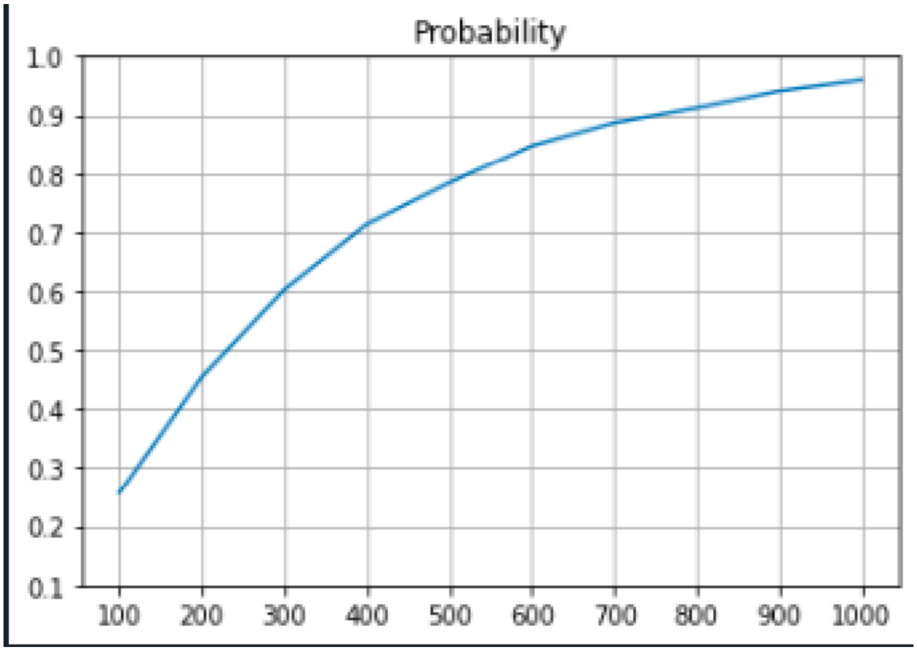
```
@author: nemernser
"""

#Question 3
#this answer is based on question 2
import matplotlib.pyplot as plt
# Using ourrange function introduced in class
def ourrange(start,stop,step=1):
    rez_range=[]
    if stop<start:
        return rez_range
    rez_range.append(start)
    while True:
        if rez_range[-1]+step<stop:
            rez_range.append(rez_range[-1]+step)
        else:
            break
    return rez_range
import random
import time
def random_walk(n):
    counter_p=0
    for j in range(10000):

        #define left and right
        left=0
        right=0
        #our list starts with 0
        x=[0]
        y=[0]
        for i in range(n):
            # get random probability
            step=random.random()
            #as the questions asks probabilities
            if step<=0.49:
                left+=1
                right+=-1
            elif step>=0.49 and step <=0.98:
                left+=1
                right+=1
            else:
                left=0
                right=0
            #add the steps
            x.append(left)
            y.append(right)
            # check if mikom is 10
            # if yes then we add 1 to our counter
            if right==10:
                passed=1
                counter_p+=passed
                #stop the for!
                break
            else:
                passed=0
    #get probability out of 10k simulations
    num=counter_p/10000
    return num
```

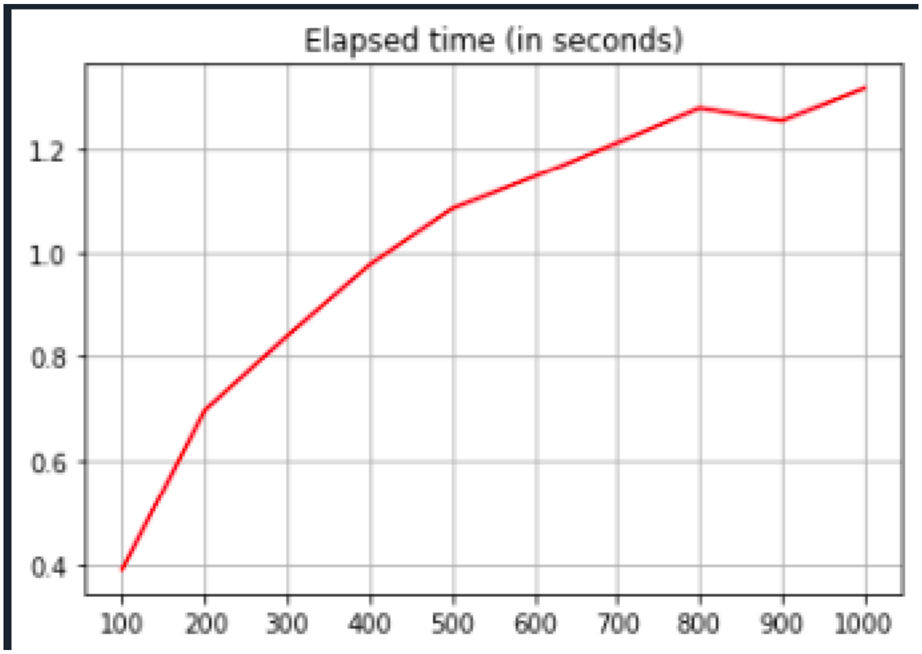
Part 2:



```
#1000 steps as requested
#graph for show
count=0
y=[]
y_ax=ourrange(0.1,1,0.1)
#list for our time
times=[]
x=ourrange(100,1001,100)
for i in x:
    #start counting time
    time_start=time.time()
    #finish the function
    count = random_walk(i)
    #end the time
    end_time=time.time()
    #add the time on each x... 100,200...1000
    times.append(end_time-time_start)
    y.append(count)
#drawing the graph...
plt.plot(x,y)
plt.xticks(x)
plt.yticks(y_ax)
plt.title("Probability")
plt.grid()
plt.show()
#second plot
plt.plot(x,times,color="red")
plt.xticks(x)
plt.grid()
plt.title("Elapsed time (in seconds)")
plt.show()
```

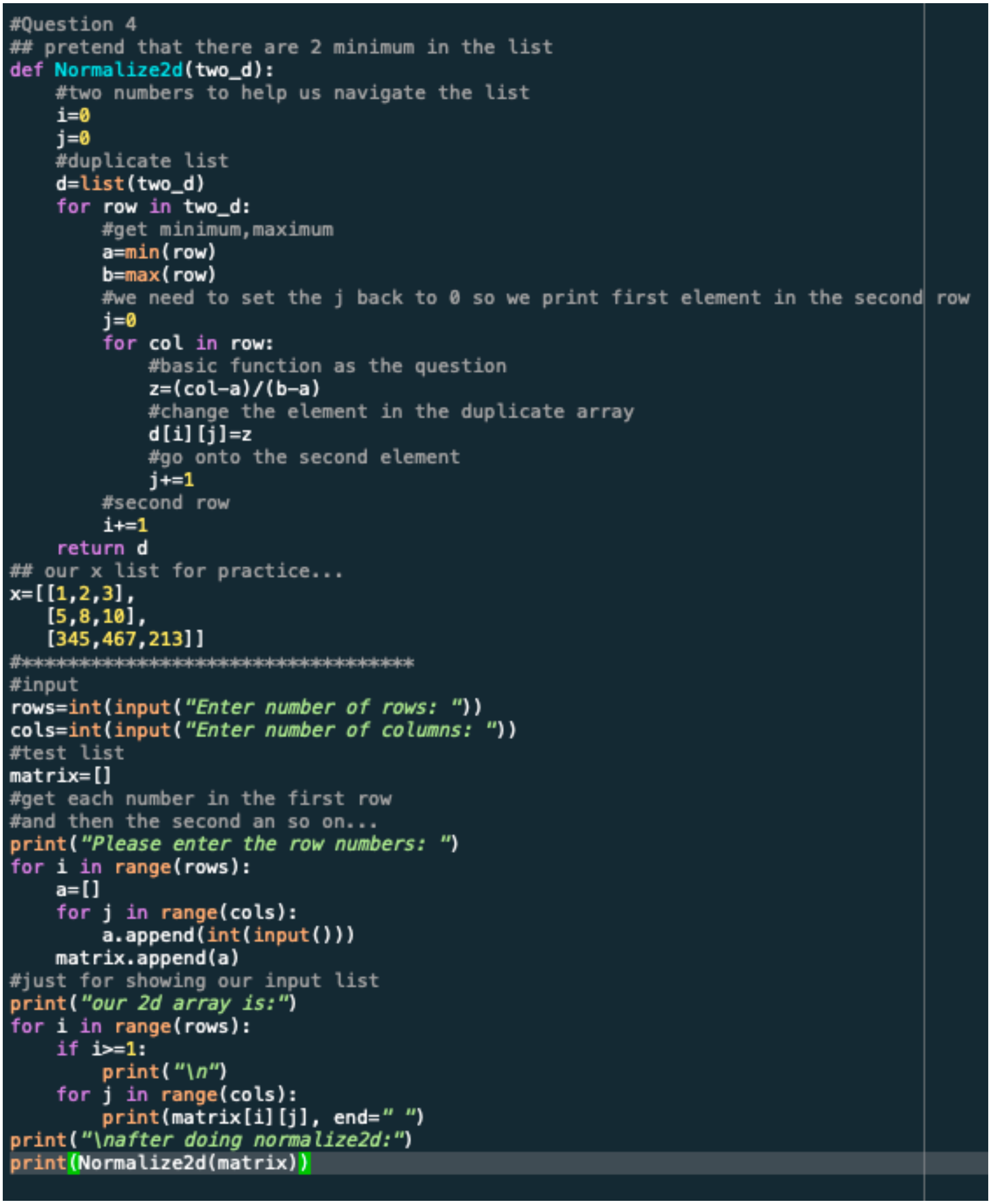
Part 3:


Probability
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
100
200
300
400
500
600
700
800
900
1000

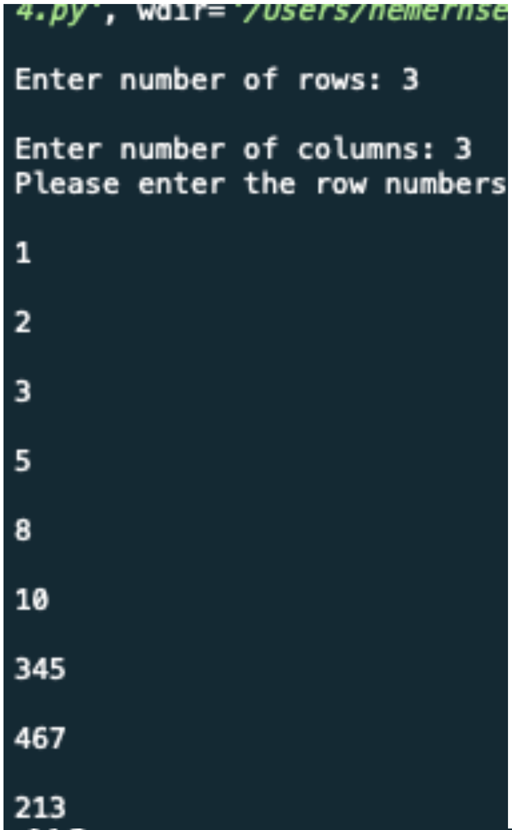

Elapsed time (in seconds)
1.2
1.0
0.8
0.6
0.4
100
200
300
400
500
600
700
800
900
1000

שאלה 4:

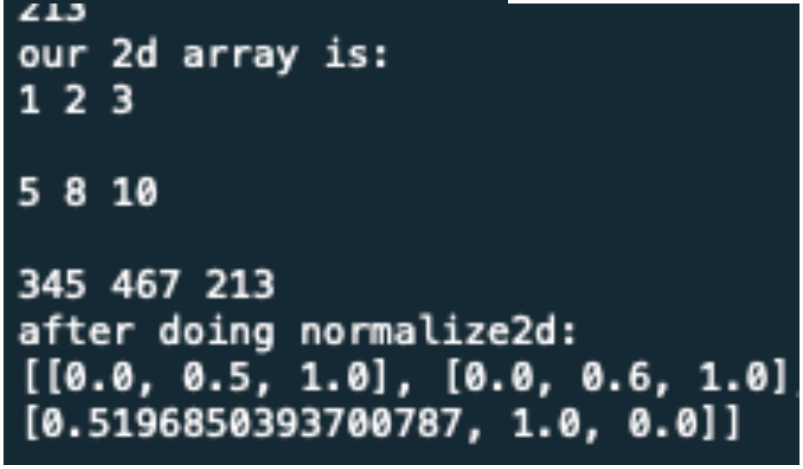
Link: https://py3.codeskulptor.org/#user306_PrU0tOM9n4_4.py



```
#Question 4
## pretend that there are 2 minimum in the list
def Normalize2d(two_d):
    #two numbers to help us navigate the list
    i=0
    j=0
    #duplicate list
    d=list(two_d)
    for row in two_d:
        #get minimum,maximum
        a=min(row)
        b=max(row)
        #we need to set the j back to 0 so we print first element in the second row
        j=0
        for col in row:
            #basic function as the question
            z=(col-a)/(b-a)
            #change the element in the duplicate array
            d[i][j]=z
            #go onto the second element
            j+=1
        #second row
        i+=1
    return d
## our x list for practice...
x=[[1,2,3],
   [5,8,10],
   [345,467,213]]
#**********************************
#input
rows=int(input("Enter number of rows: "))
cols=int(input("Enter number of columns: "))
#test list
matrix=[]
#get each number in the first row
#and then the second an so on...
print("Please enter the row numbers: ")
for i in range(rows):
    a=[]
    for j in range(cols):
        a.append(int(input()))
    matrix.append(a)
#just for showing our input list
print("our 2d array is:")
for i in range(rows):
    if i>=1:
        print("\n")
    for j in range(cols):
        print(matrix[i][j], end=" ")
print("\nafter doing normalize2d:")
print(Normalize2d(matrix))
```



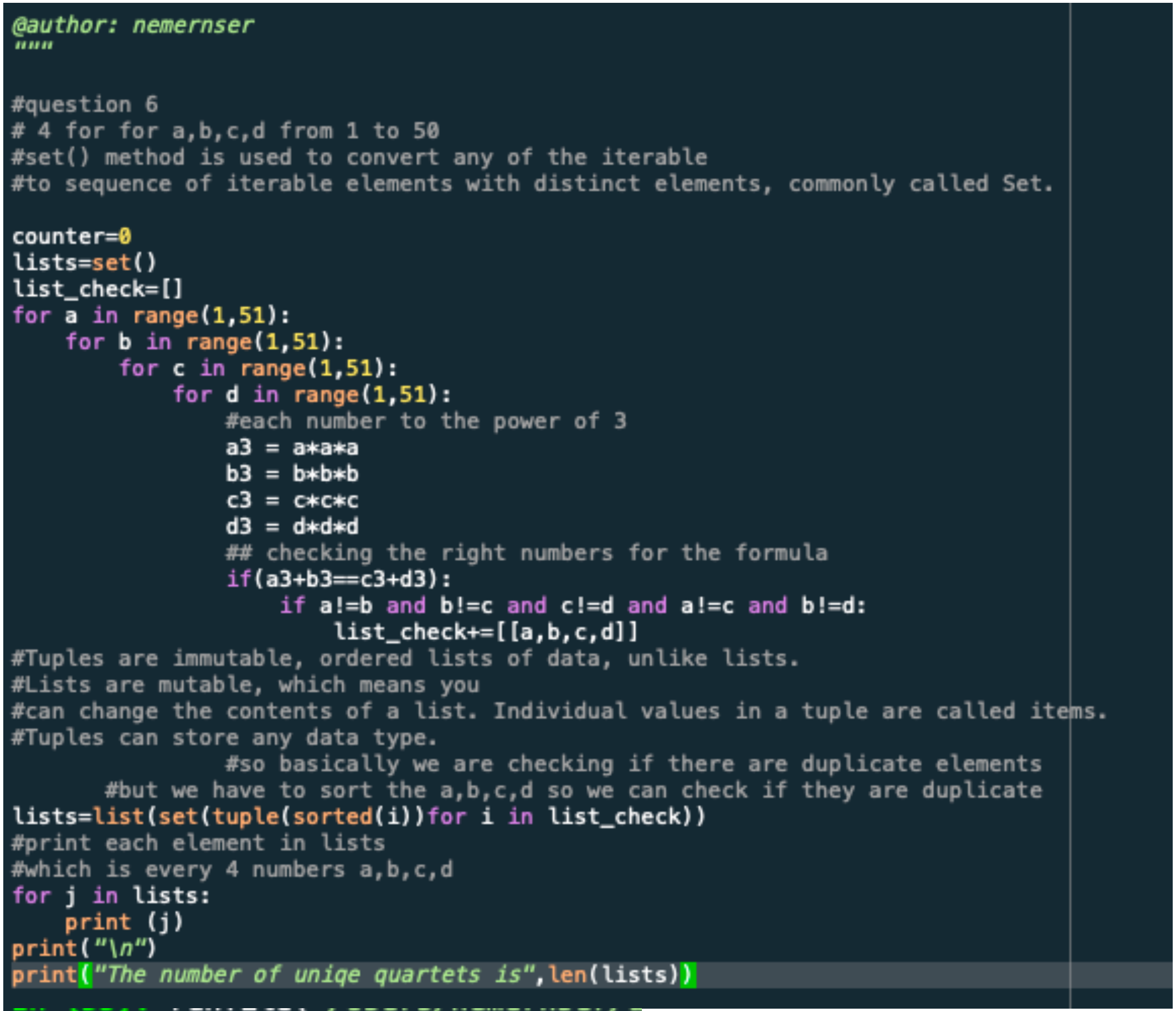
```
4.py', wdir='/Users/nemernse

Enter number of rows: 3

Enter number of columns: 3
Please enter the row numbers

1

2

3

5

8

10

345

467

213
```



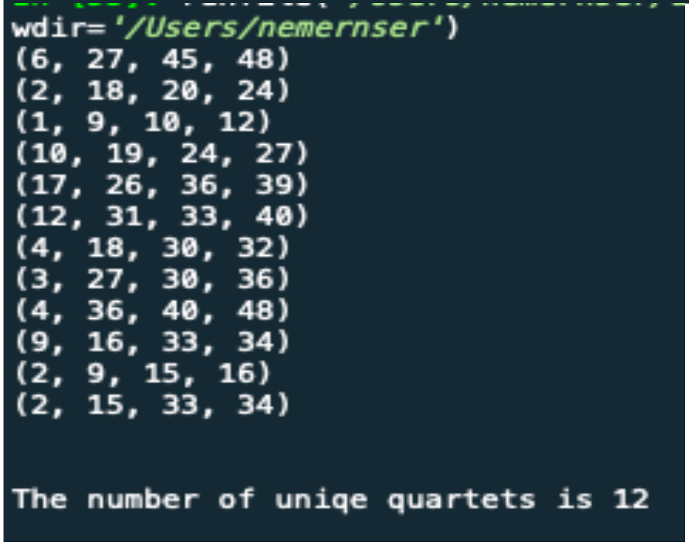
```
213
our 2d array is:
1 2 3

5 8 10

345 467 213
after doing normalize2d:
[[0.0, 0.5, 1.0], [0.0, 0.6, 1.0],
[0.5196850393700787, 1.0, 0.0]]
```

שאלה 5:

Link: https://py3.codeskulptor.org/#user306_PrU0tOM9n4_5.py



```
@author: nemernser
"""

#question 6
# 4 for for a,b,c,d from 1 to 50
#set() method is used to convert any of the iterable
#to sequence of iterable elements with distinct elements, commonly called Set.

counter=0
lists=set()
list_check=[]
for a in range(1,51):
    for b in range(1,51):
        for c in range(1,51):
            for d in range(1,51):
                #each number to the power of 3
                a3 = a*a*a
                b3 = b*b*b
                c3 = c*c*c
                d3 = d*d*d
                ## checking the right numbers for the formula
                if(a3+b3==c3+d3):
                    if a!=b and b!=c and c!=d and a!=c and b!=d:
                        list_check+=[[a,b,c,d]]
#Tuples are immutable, ordered lists of data, unlike lists.
#Lists are mutable, which means you
#can change the contents of a list. Individual values in a tuple are called items.
#Tuples can store any data type.
                #so basically we are checking if there are duplicate elements
       #but we have to sort the a,b,c,d so we can check if they are duplicate
lists=list(set(tuple(sorted(i))for i in list_check))
#print each element in lists
#which is every 4 numbers a,b,c,d
for j in lists:
    print (j)
print("\n")
print("The number of uniqe quartets is",len(lists))
```



```
wdir='/Users/nemernser')
(6, 27, 45, 48)
(2, 18, 20, 24)
(1, 9, 10, 12)
(10, 19, 24, 27)
(17, 26, 36, 39)
(12, 31, 33, 40)
(4, 18, 30, 32)
(3, 27, 30, 36)
(4, 36, 40, 48)
(9, 16, 33, 34)
(2, 9, 15, 16)
(2, 15, 33, 34)


The number of uniqe quartets is 12
```