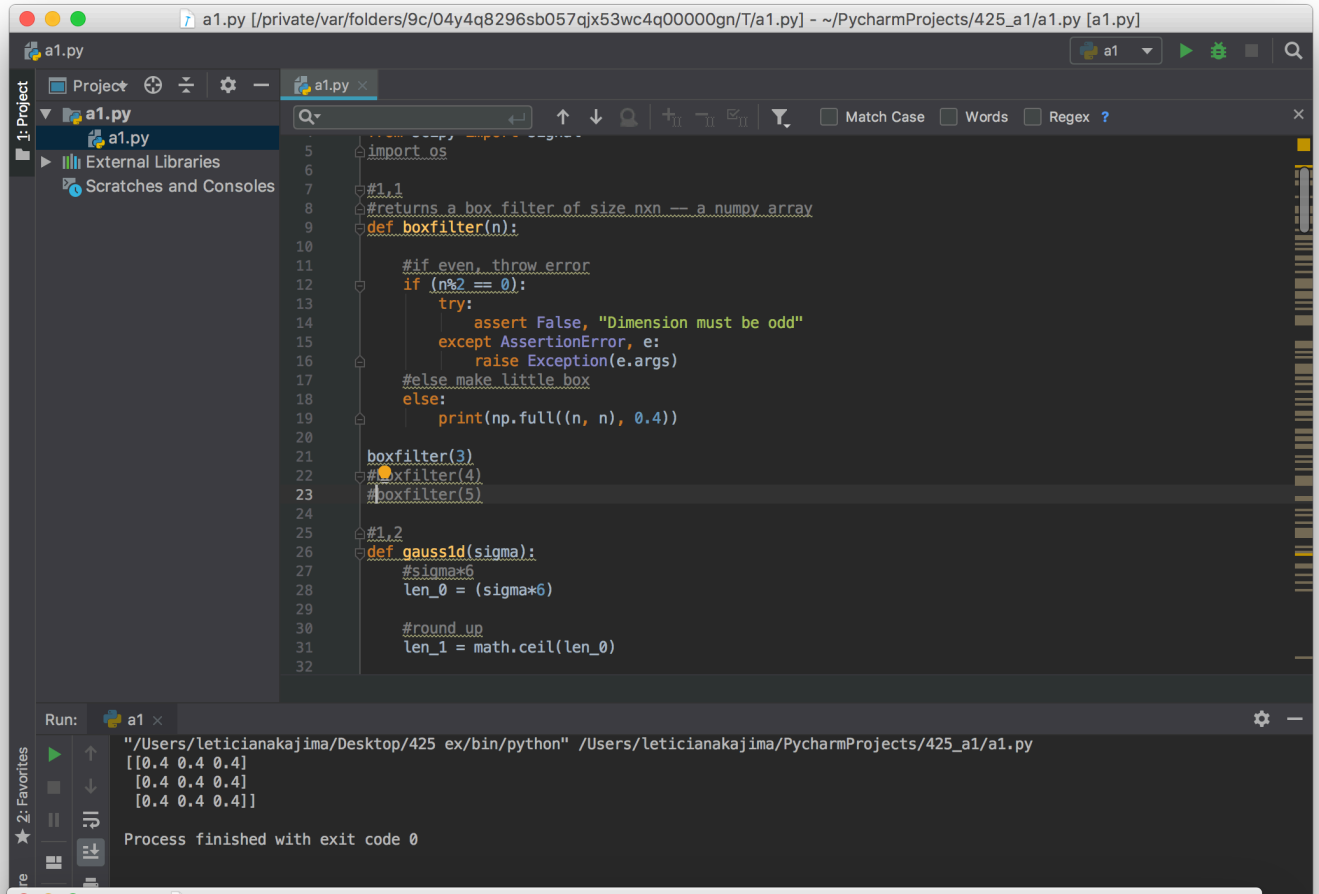
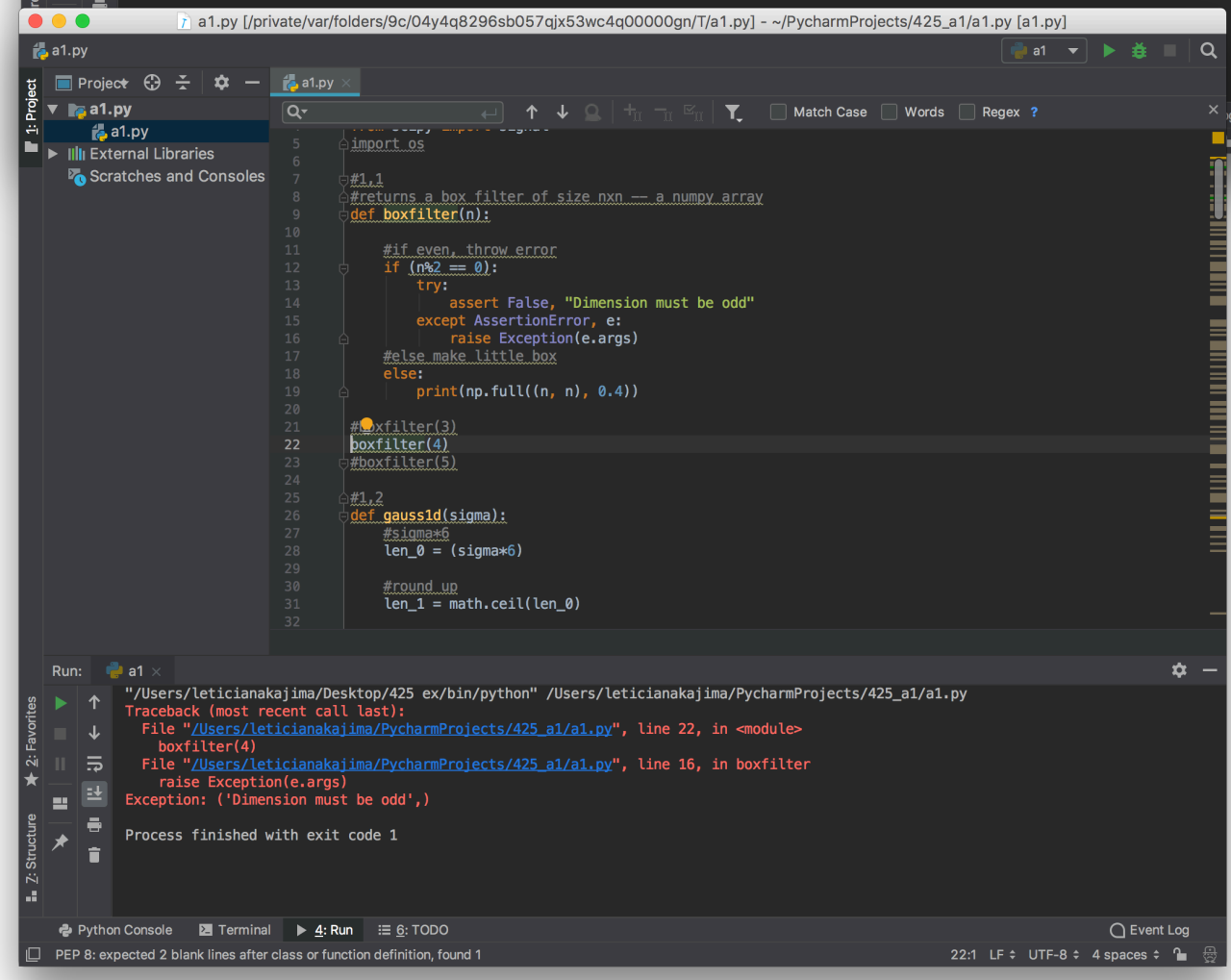
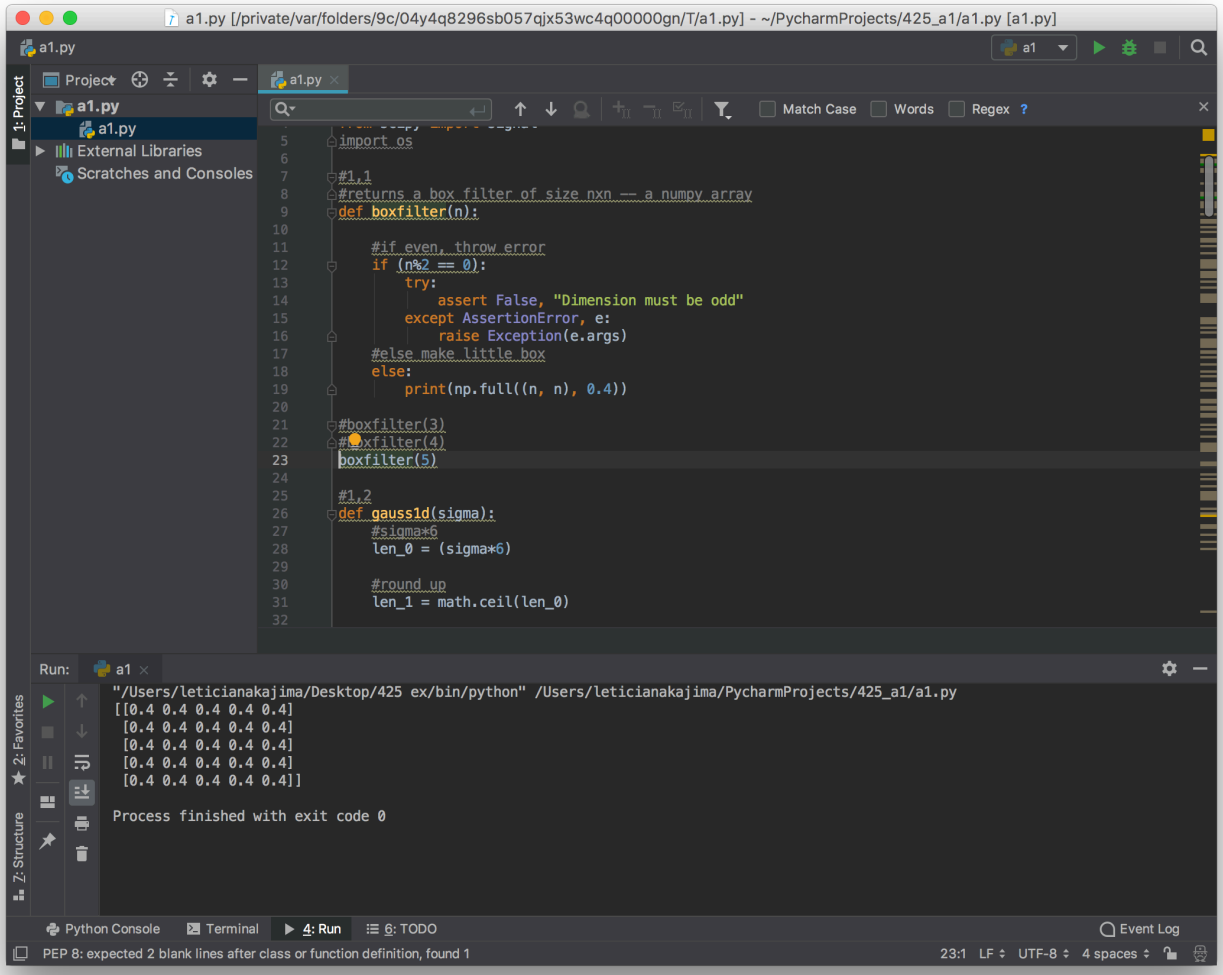
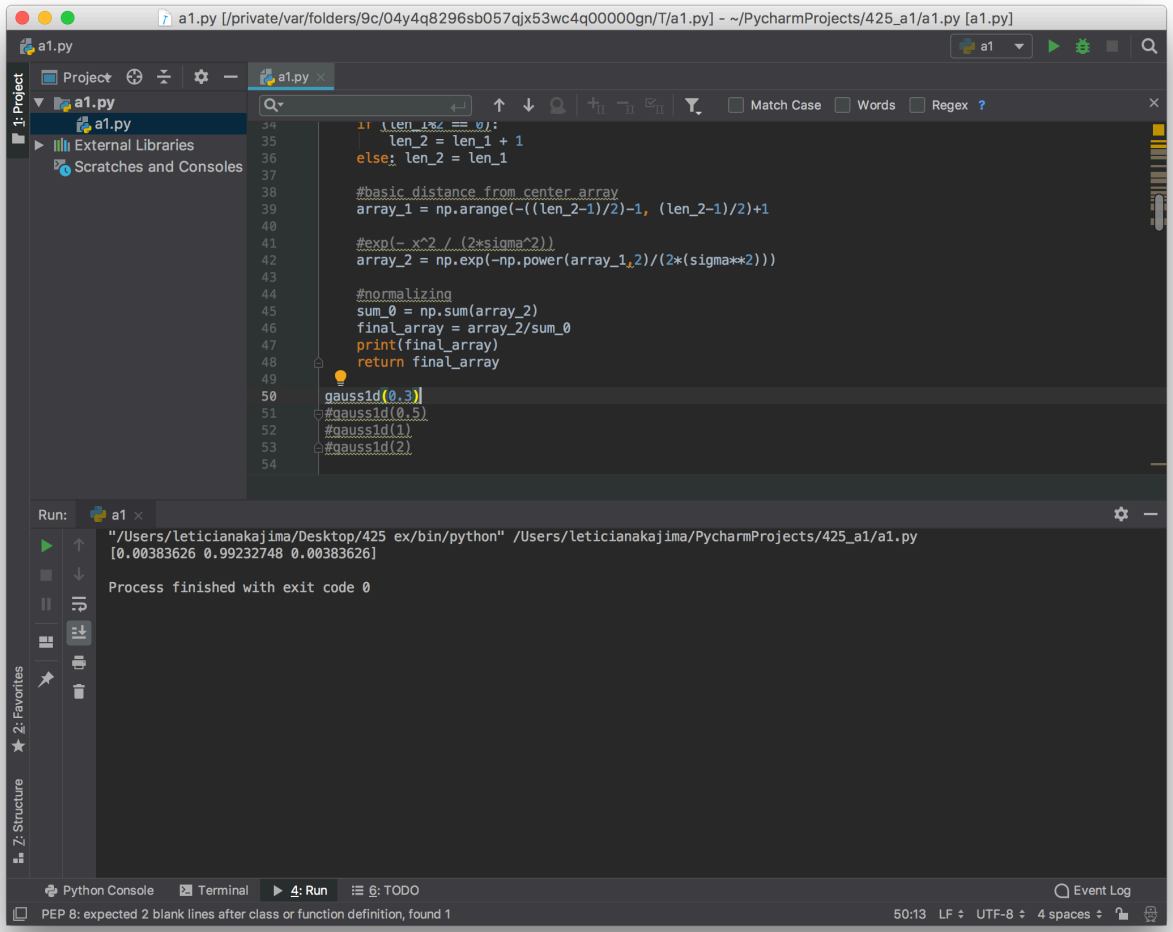
PART1:
1.tests

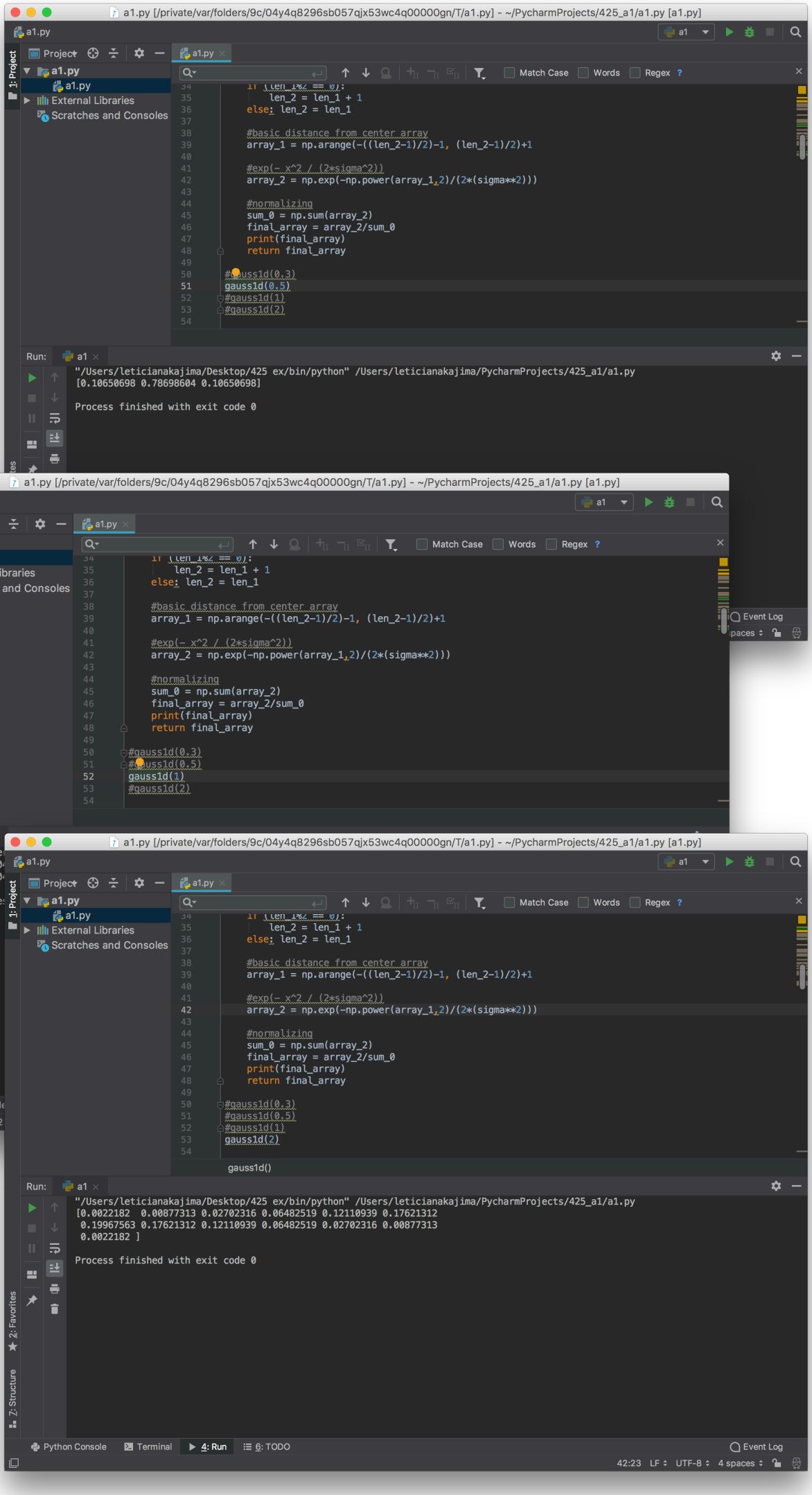




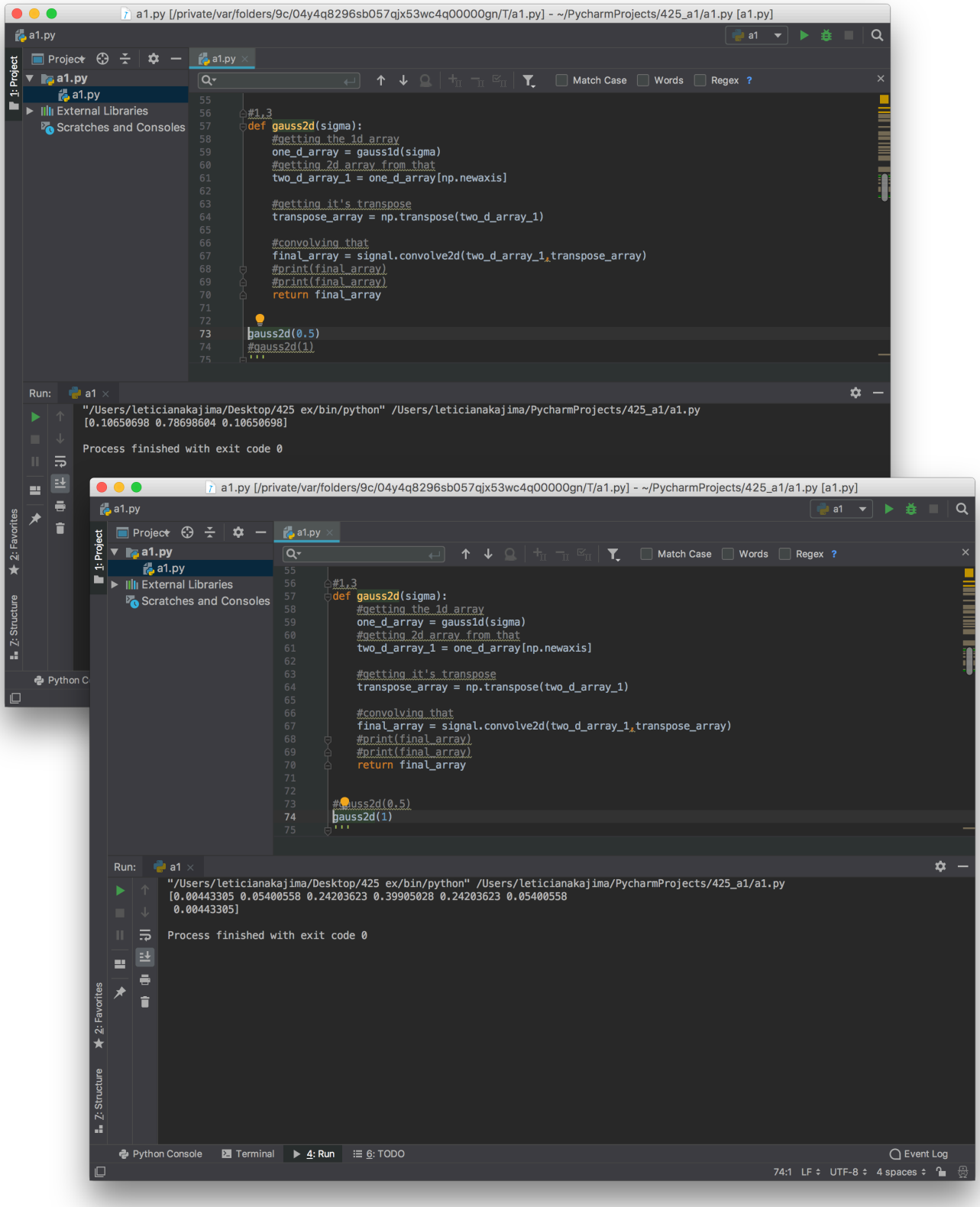

a1.py [/private/var/folders/9c/04y4q8296sb057qjx53wc4q00000gn/T/a1.py] - ~/PycharmProjects/425_a1/a1.py [a1.py]
a1.py
Project
a1.py
a1.py
External Libraries
Scratches and Consoles
Match Case
Words
Regex
import os
#1.1
#returns a box filter of size nxn == a numpy array
def boxfilter(n):
#if even, throw error
if (n%2 == 0):
try:
assert False, "Dimension must be odd"
except AssertionError, e:
raise Exception(e.args)
#else make little box
else:
print(np.full((n, n), 0.4))
#boxfilter(3)
#boxfilter(4)
boxfilter(5)
#1.2
def gauss1d(sigma):
#sigma*6
len_0 = (sigma*6)
#round up
len_1 = math.ceil(len_0)
Run:
a1
"/Users/leticianakajima/Desktop/425 ex/bin/python" /Users/leticianakajima/PycharmProjects/425_a1/a1.py
[[0.4 0.4 0.4 0.4 0.4]
[0.4 0.4 0.4 0.4 0.4]
[0.4 0.4 0.4 0.4 0.4]
[0.4 0.4 0.4 0.4 0.4]
[0.4 0.4 0.4 0.4 0.4]]
Process finished with exit code 0
Python Console
Terminal
4: Run
6: TODO
Event Log
PEP 8: expected 2 blank lines after class or function definition, found 1
23:1
LF
UTF-8
4 spaces
1: Project
2: Favorites
Z: Structure

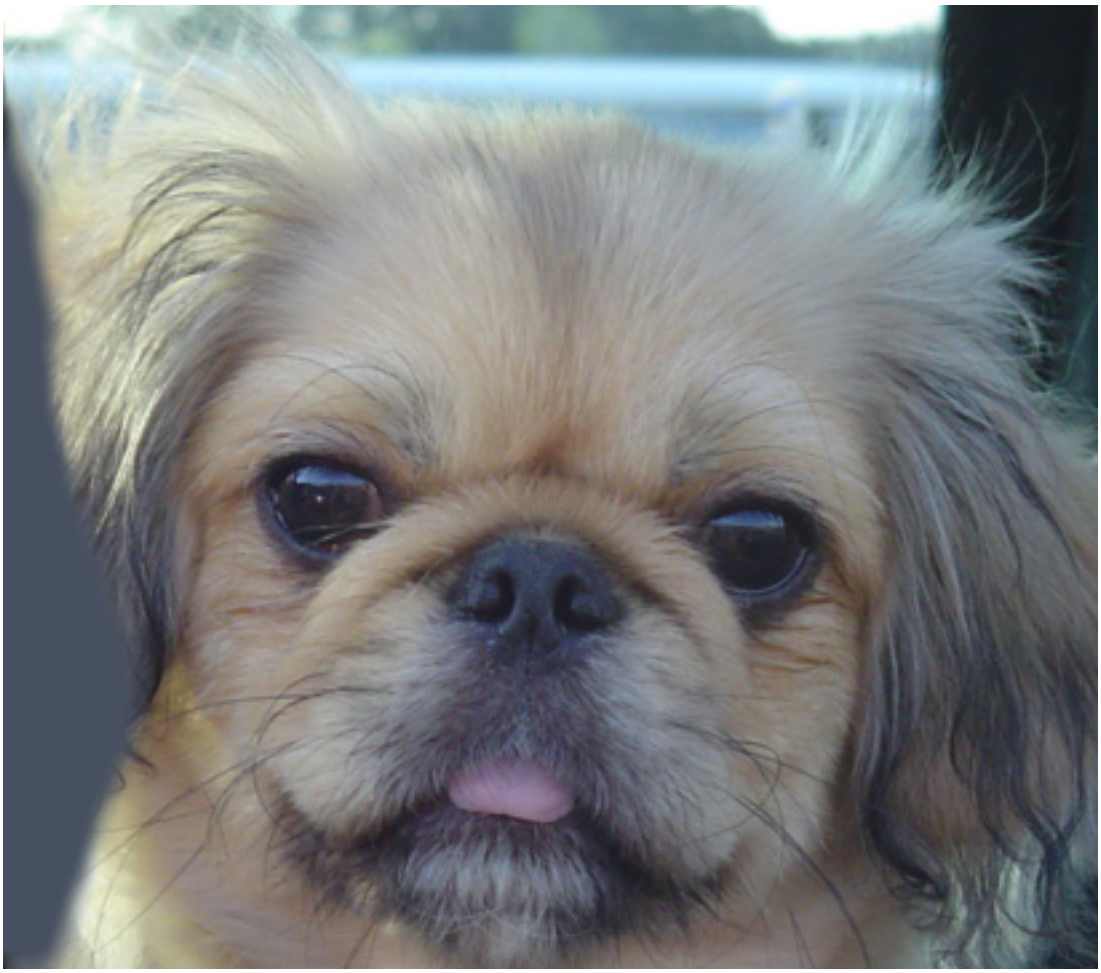
2.tests


a1.py [/private/var/folders/9c/04y4q8296sb057qjx53wc4q00000gn/T/a1.py] - ~/PycharmProjects/425_a1/a1.py [a1.py]
a1.py
Project
a1.py
a1.py
External Libraries
Scratches and Consoles
Match Case
Words
Regex
len_2 = len_1 + 1
else: len_2 = len_1
#basic distance from center array
array_1 = np.arange(-((len_2-1)/2)-1, (len_2-1)/2)+1
#exp(- x^2 / (2*sigma^2))
array_2 = np.exp(-np.power(array_1,2)/(2*(sigma**2)))
#normalizing
sum_0 = np.sum(array_2)
final_array = array_2/sum_0
print(final_array)
return final_array
gauss1d(0.3)
#gauss1d(0.5)
#gauss1d(1)
#gauss1d(2)
Run:
a1
"/Users/leticianakajima/Desktop/425 ex/bin/python" /Users/leticianakajima/PycharmProjects/425_a1/a1.py
[0.00383626 0.99232748 0.00383626]
Process finished with exit code 0
Python Console
Terminal
4: Run
6: TODO
Event Log
PEP 8: expected 2 blank lines after class or function definition, found 1
50:13
LF
UTF-8
4 spaces
1: Project
2: Favorites
Z: Structure


a1.py [/private/var/folders/9c/04y4q8296sb057qjx53wc4q00000gn/T/a1.py] - ~/PycharmProjects/425_a1/a1.py [a1.py]
a1.py
a1
Project
a1.py
a1.py
External Libraries
Scratches and Consoles
Match Case
Words
Regex
if (len_1%2 == 0):
len_2 = len_1 + 1
else: len_2 = len_1
#basic distance from center array
array_1 = np.arange(-((len_2-1)/2)-1, (len_2-1)/2)+1
#exp(- x^2 / (2*sigma^2))
array_2 = np.exp(-np.power(array_1,2)/(2*(sigma**2)))
#normalizing
sum_0 = np.sum(array_2)
final_array = array_2/sum_0
print(final_array)
return final_array
#gauss1d(0.3)
gauss1d(0.5)
#gauss1d(1)
#gauss1d(2)
Run:
a1
"/Users/leticianakajima/Desktop/425 ex/bin/python" /Users/leticianakajima/PycharmProjects/425_a1/a1.py
[0.10650698 0.78698604 0.10650698]
Process finished with exit code 0
a1.py [/private/var/folders/9c/04y4q8296sb057qjx53wc4q00000gn/T/a1.py] - ~/PycharmProjects/425_a1/a1.py [a1.py]
a1.py
a1
Project
a1.py
a1.py
External Libraries
Scratches and Consoles
Match Case
Words
Regex
if (len_1%2 == 0):
len_2 = len_1 + 1
else: len_2 = len_1
#basic distance from center array
array_1 = np.arange(-((len_2-1)/2)-1, (len_2-1)/2)+1
#exp(- x^2 / (2*sigma^2))
array_2 = np.exp(-np.power(array_1,2)/(2*(sigma**2)))
#normalizing
sum_0 = np.sum(array_2)
final_array = array_2/sum_0
print(final_array)
return final_array
#gauss1d(0.3)
#gauss1d(0.5)
gauss1d(1)
#gauss1d(2)
Event Log
Run:
a1
1: Project
2: Favorites
Z: Structure
Python Console
PEP 8: expected 2
a1.py [/private/var/folders/9c/04y4q8296sb057qjx53wc4q00000gn/T/a1.py] - ~/PycharmProjects/425_a1/a1.py [a1.py]
a1.py
a1
Project
a1.py
a1.py
External Libraries
Scratches and Consoles
Match Case
Words
Regex
if (len_1%2 == 0):
len_2 = len_1 + 1
else: len_2 = len_1
#basic distance from center array
array_1 = np.arange(-((len_2-1)/2)-1, (len_2-1)/2)+1
#exp(- x^2 / (2*sigma^2))
array_2 = np.exp(-np.power(array_1,2)/(2*(sigma**2)))
#normalizing
sum_0 = np.sum(array_2)
final_array = array_2/sum_0
print(final_array)
return final_array
#gauss1d(0.3)
#gauss1d(0.5)
#gauss1d(1)
gauss1d(2)
gauss1d()
Run:
a1
"/Users/leticianakajima/Desktop/425 ex/bin/python" /Users/leticianakajima/PycharmProjects/425_a1/a1.py
[0.0022182 0.00877313 0.02702316 0.06482519 0.12110939 0.17621312
0.19967563 0.17621312 0.12110939 0.06482519 0.02702316 0.00877313
0.0022182]
Process finished with exit code 0
1: Project
2: Favorites
Z: Structure
Python Console
Terminal
4: Run
6: TODO
Event Log
42:23
LF
UTF-8
4 spaces

3.


a1.py [/private/var/folders/9c/04y4q8296sb057qjx53wc4q00000gn/T/a1.py] - ~/PycharmProjects/425_a1/a1.py [a1.py]
a1.py
Project
a1.py
a1.py
External Libraries
Scratches and Consoles
Match Case
Words
Regex
#1.3
def gauss2d(sigma):
#getting the 1d array
one_d_array = gauss1d(sigma)
#getting 2d array from that
two_d_array_1 = one_d_array[np.newaxis]
#getting it's transpose
transpose_array = np.transpose(two_d_array_1)
#convolving that
final_array = signal.convolve2d(two_d_array_1,transpose_array)
#print(final_array)
#print(final_array)
return final_array
gauss2d(0.5)
#gauss2d(1)
'''
Run:
a1
"/Users/leticianakajima/Desktop/425 ex/bin/python" /Users/leticianakajima/PycharmProjects/425_a1/a1.py
[0.10650698 0.78698604 0.10650698]
Process finished with exit code 0
a1.py [/private/var/folders/9c/04y4q8296sb057qjx53wc4q00000gn/T/a1.py] - ~/PycharmProjects/425_a1/a1.py [a1.py]
gauss2d(1)
"/Users/leticianakajima/Desktop/425 ex/bin/python" /Users/leticianakajima/PycharmProjects/425_a1/a1.py
[0.00443305 0.05400558 0.24203623 0.39905028 0.24203623 0.05400558
0.00443305]
Process finished with exit code 0
Python Console
Terminal
4: Run
6: TODO
Event Log
74:1
LF
UTF-8
4 spaces
1: Project
2: Favorites
7: Structure



4.

4a. Correlation represents the similarity between 2 images while convolution represents the effect of one image ‘onto’ another.

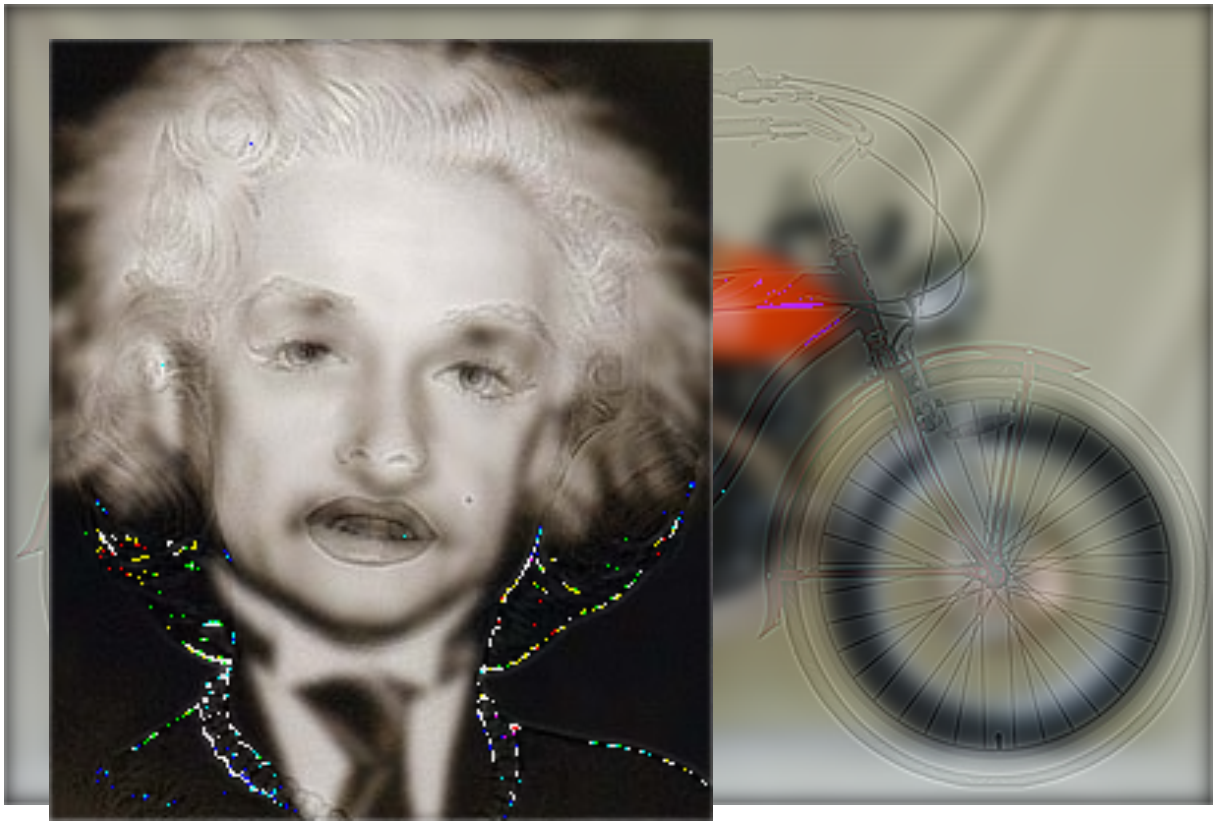
5. This could be more efficiently done by taking advantage of separability - which reduces the number of multiplications required. Doing it this way we would reduce multiplications from (m^2*n^2) to 2m*n^2.

PART2:
1.hybrid image #1

here both the cat and god image are sigma = 3

2.hybrid image #2

here the motorcycle is made using sigma = 4 and the bicycle using sigma = 0.55



3.hybrid image - here the Einstein photo has a sigma = 2 and the Marilyn photo has sigma = 1