**Assignment 2**

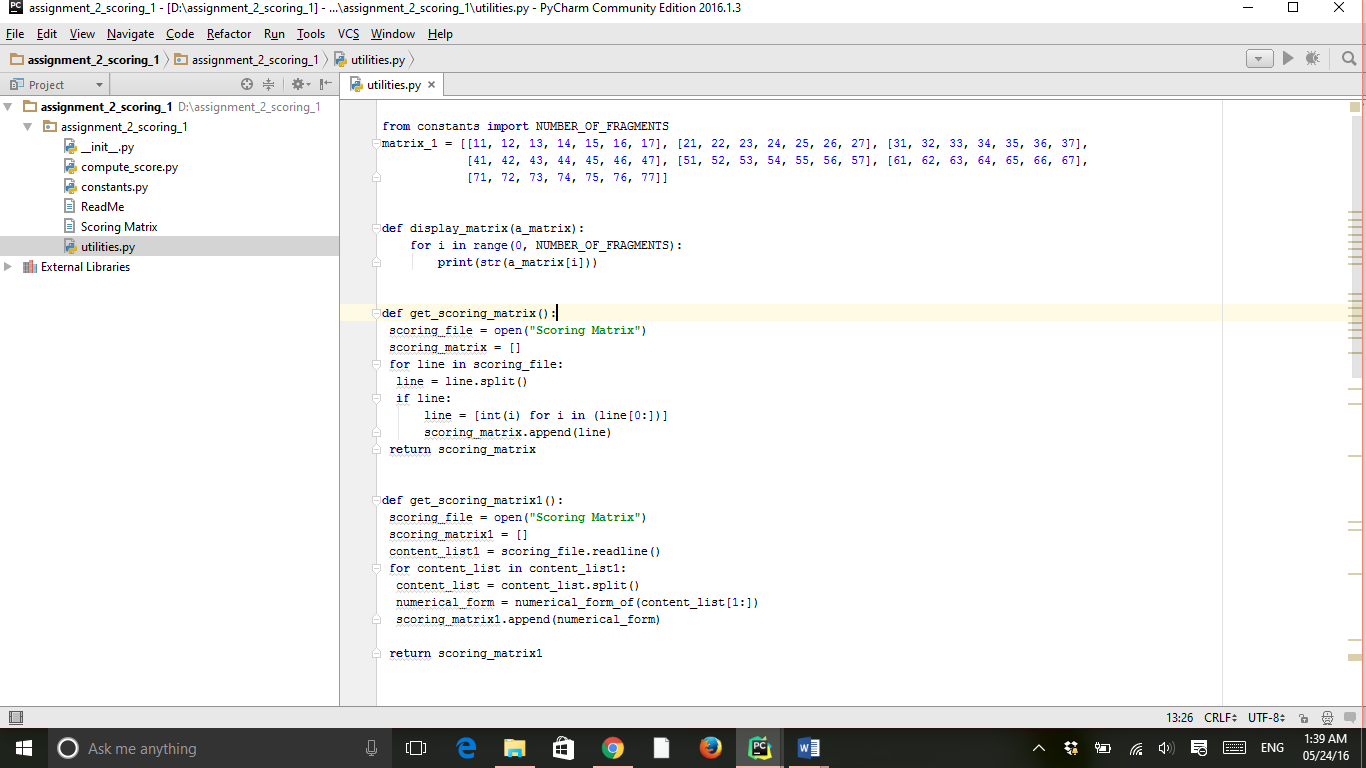
**CS 521**

**Information structures with python**

**Purpose :**

To deliver the code for getting the percentage of which the student response was correct

**Structure of assignment**



**Code for utilities.py and compute\_score is attached as .zip file**

**utilities.py**

**from** constants **import** NUMBER\_OF\_FRAGMENTS *###importing the variable from other file  
  
##### Code for creating function display\_matrix ##########***def** display\_matrix(a\_matrix):  
 **for** i **in** range(0, NUMBER\_OF\_FRAGMENTS): *#### for 0 to number of fragments* **print**(str(a\_matrix[i])) *#### display the corresponding matrix*

***I did not used the code given corresponding to get\_scoring \_matrix ,I used different variable and function***  
*##### Code for creating function get\_scoring\_matrix ##########***def** get\_scoring\_matrix():  
 scoring\_matrix = []  
 scoring\_file = open(**"Scoring Matrix"**)*###opening the socinr matrix the read file line by line* i=0  
 **while**( i<=NUMBER\_OF\_FRAGMENTS):  
 content\_of\_line = scoring\_file.readline()  
 content\_list = content\_of\_line.split(**' '**)*##for blankspaces* numerical\_form = numerical\_form\_of(content\_list[1:])  
 scoring\_matrix.append(numerical\_form)  
 i += 1  
 **del** scoring\_matrix[0]  
 **return** scoring\_matrix

*##### Code for creating function numerical\_form\_of ##########***def** numerical\_form\_of(a\_list):  
 **return** [int(a\_list[i]) **for** i **in** range(len(a\_list))] *#### return the list with the corresponding integers*

*##### Code for creating function pointwise\_product ##########***def** pointwise\_product(a\_matrix, a\_second\_matrix): *### define function* m = [[0 **for** \_ **in** range(7)] **for** \_ **in** range(7)]*## intializing the matrix to null value* **for** i **in** range(0, NUMBER\_OF\_FRAGMENTS): *## for every i(row) starting from 0 to number\_fragments* **for** j **in** range(0, NUMBER\_OF\_FRAGMENTS): *##for every i(column) starting from 0 to number\_fragments* m[i][j] = (a\_matrix[i][j] \* a\_second\_matrix[i][j]) *## m[i][j]= multiplication of two matrix (a\_matrix and a\_second\_matrix)* **return** m

*########### Code for creating function list\_as\_precedence\_matrix (student response is calculated)#########***def** list\_as\_precedence\_matrix(a\_response): *### define function* return\_matrix = [[0 **for** row **in** range(NUMBER\_OF\_FRAGMENTS)] *## intializing the matrix to null value* **for** column **in** range(NUMBER\_OF\_FRAGMENTS)]  
  
 **for** index\_of\_first **in** range(len(a\_response) - 1): *### for loop index\_of\_first till length of (a\_response-1)* **for** index\_of\_second **in** range(index\_of\_first + 1, len(a\_response)):  
 value1, value2 = int(a\_response[index\_of\_first]), int(a\_response[index\_of\_second])  
 **if** value1 < value2: *## if the value1 < value 2 fro eg value1=3 and value2=4* return\_matrix[value1 - 1][value2 - 1] = 1 *##return\_matrix (the index of (value1 -1)(value2 -1)) will be initilaize to 1* **return** return\_matrix *###then return return\_matrix*

*############ Code for creating function sum\_of \_upper\_triangle ##############***def** sum\_of\_upper\_triangle(a\_matrix):  
 sum\_returned = 0 *##### initializing to zero so that it doesnot any garabage value* **for** row **in** range(NUMBER\_OF\_FRAGMENTS): *### for loop for getting into row from 0 to number of fragments* **for** column **in** range(row + 1, NUMBER\_OF\_FRAGMENTS): *### for loop for getting into column starting row+1 to number\_of fragments* sum\_returned += a\_matrix[row][column] *####### sum of a\_matrix is assign to sum\_returned variable* **return** (sum\_returned) *### return the value*

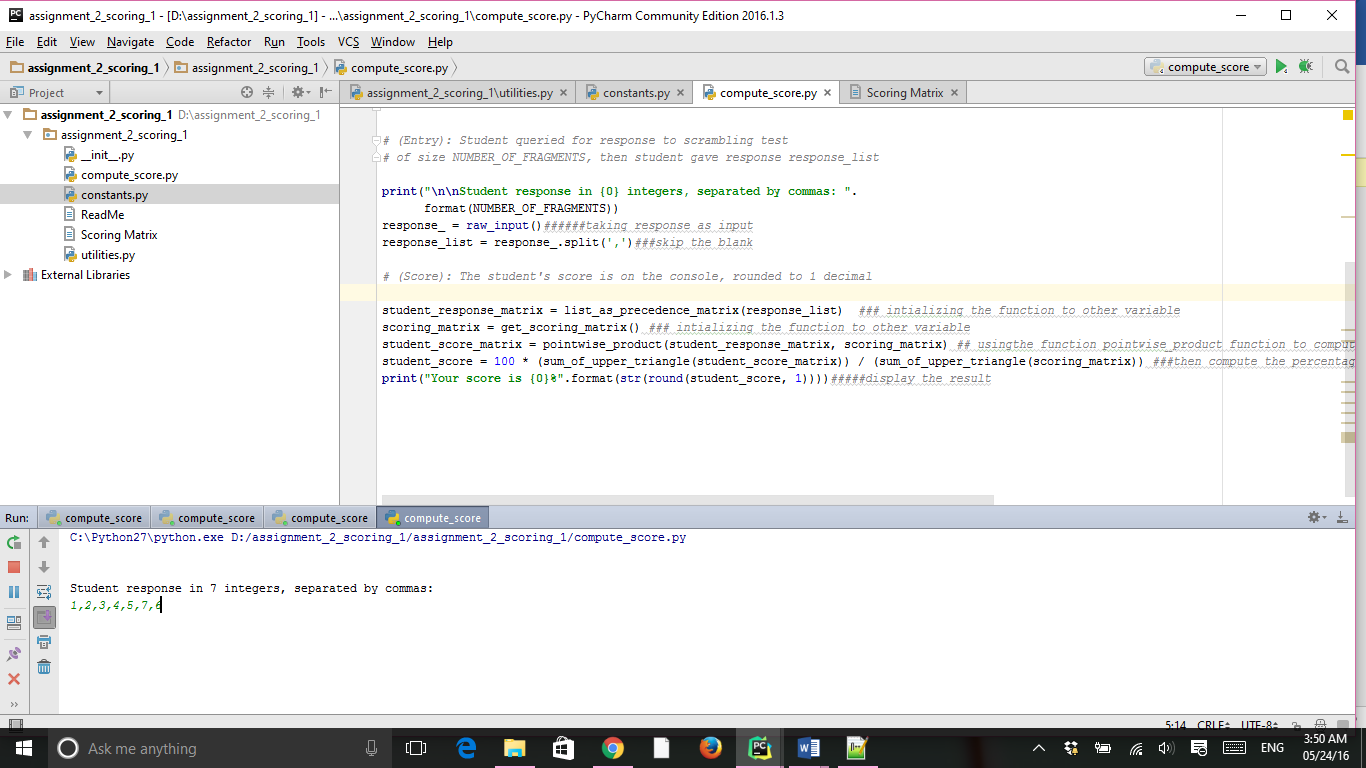
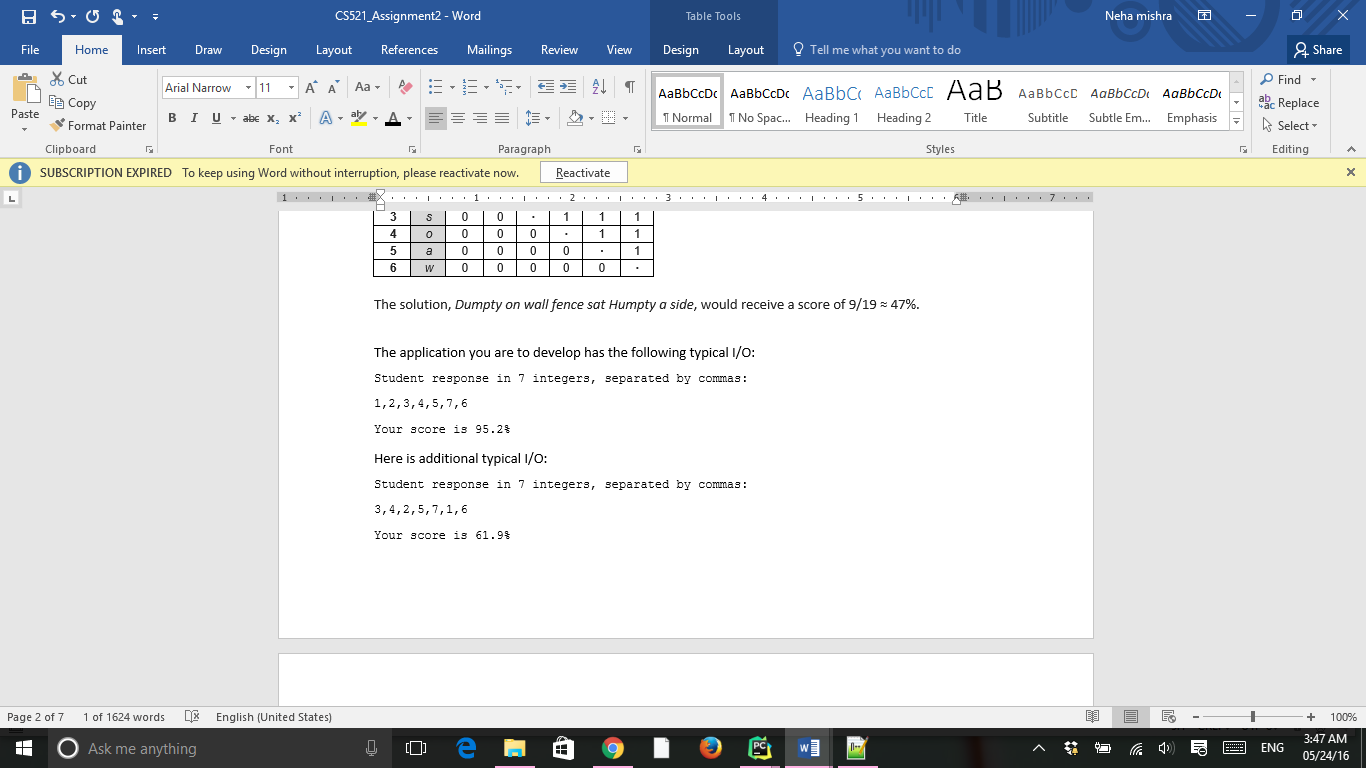
**compute\_score.py**

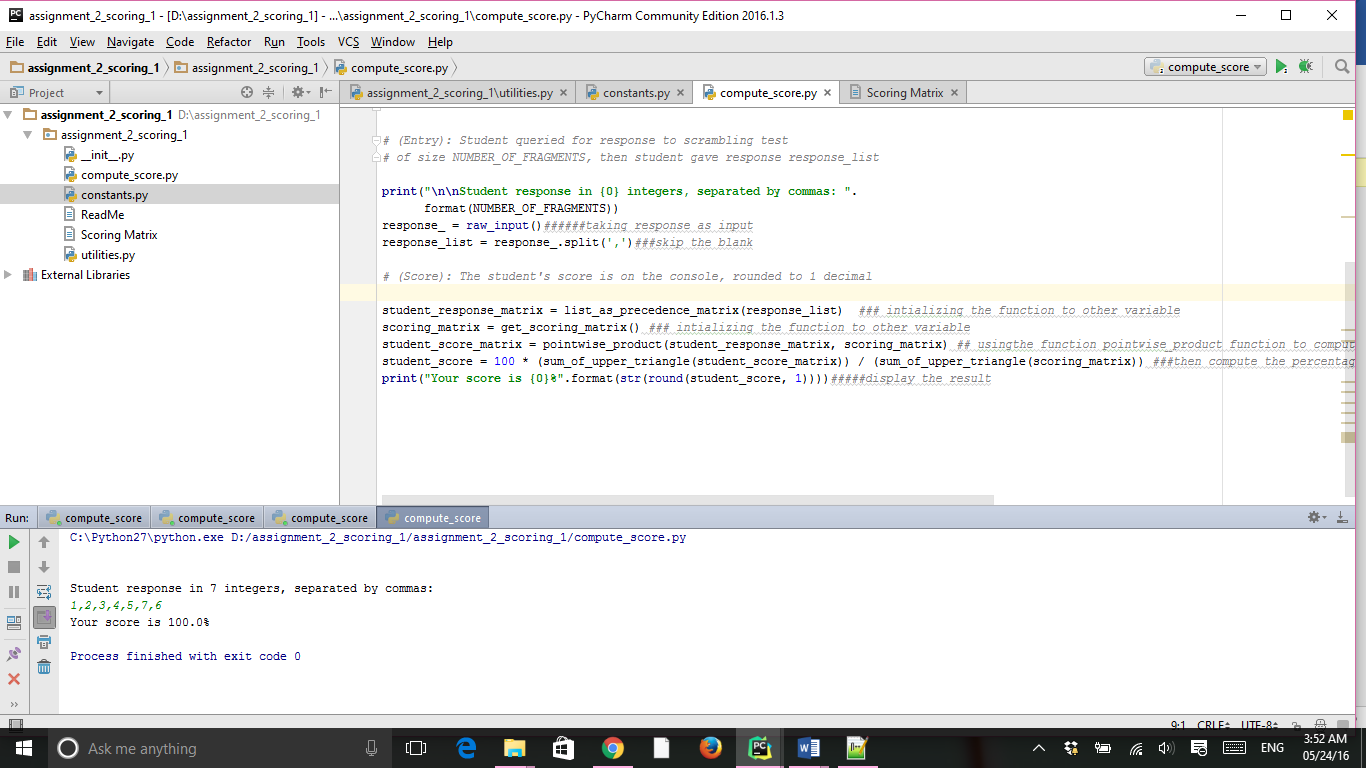
**from** constants **import** NUMBER\_OF\_FRAGMENTS  
**from** utilities **import** display\_matrix,numerical\_form\_of,\  
 get\_scoring\_matrix, pointwise\_product, list\_as\_precedence\_matrix,\  
 sum\_of\_upper\_triangle  
  
  
  
**'''  
User was queried for response to scrambling test of size NUMBER\_OF\_FRAGMENTS,  
then student gave response, then evaluation matrix was shown, then  
student's score was shown as a percentage.  
'''***# (Entry): Student queried for response to scrambling test  
# of size NUMBER\_OF\_FRAGMENTS, then student gave response response\_list***print**(**"\n\nStudent response in {0} integers, separated by commas: "**.  
 format(NUMBER\_OF\_FRAGMENTS))  
response\_ = raw\_input()*######taking response as input*response\_list = response\_.split(**','**)*###skip the blank  
  
# (Score): The student's score is on the console, rounded to 1 decimal*student\_response\_matrix = list\_as\_precedence\_matrix(response\_list) *### intializing the function to other variable*scoring\_matrix = get\_scoring\_matrix() *### intializing the function to other variable*student\_score\_matrix = pointwise\_product(student\_response\_matrix, scoring\_matrix) *## usingthe function pointwise\_product function to compute the matrix multiplication of student\_response\_matrix andscoring\_matrix*student\_score = 100 \* (sum\_of\_upper\_triangle(student\_score\_matrix)) / (sum\_of\_upper\_triangle(scoring\_matrix)) *###then compute the percentage by the calculating the value is matrix***print**(**"Your score is {0}%"**.format(str(round(student\_score, 1))))*#####display the result*

**output:**

the input I give was 1,2,3,4,5,7,6

when we run the file it asked for the student response :





When I give input 3,4,2,5,7,1,6

The output was

