**CS Homework #26: Numpy**

*Deadline: 5/9/2020, 9:00 pm.*
*Save your code as lastname_homework26.py and submit on* ***GOOGLE CLASSROOM***

**Task 1**
Using np.array(), manually create a numpy array that looks like this:

| | |
|---|---|
| 5 | 0 |
| 7 | 1 |

**Task 2**
Using either np.zeros() or np.ones(), create a two-dimensional numpy array that has 3 rows and 4 columns. The data type of the array should be *np.int32*.

**Task 3**
Using the array from Task 2, change all values in the array to 9.

**Task 4**
Complete the same task (create a 3 by 4 array with all values equal to 9) using np.repeat(x, y) command, where *x* is the value(s) that you repeat and *y* is how many types you repeat them.

**Task 5**
Create a 5 by 5 array, with integer values randomly drawn between 4 and 8 (included). Print the value of the cell located in the second row and third column of the array.

**Task 6**
Create a 5 by 5 array, with continuous values randomly drawn between 0.1 and 0.2. Print all values located in the fourth row of the array. Can you print these values one at a time (that is, one per line)? Can you print these values all at once (the whole row)? Hint: Try the slicing techniques that we used when working with lists.

**Task 7**
Using np.arange(), create a 4 by 4 array that looks like this (notice that we start with 1, not 0!):

| | | |
|---|---|---|
| 1 | 2 | 3 |
| 4 | 5 | 6 |
| 7 | 8 | 9 |

Let's call it X. What is np.sum(X)? What is np.sum(X, 0)? What is np.sum(X, 1)?

**Task 8**
What is np.mean(X)? What is np.mean(X, 0)? What is np.mean(X, 1)?

**Task 9**
Using for loops, create a 4 by 4 identity matrix (consisting of integer values):

| | | | |
|---|---|---|---|
| 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 1 |

(Do not do this manually. Do not use np.identity() since we did not study this command. Use for loops to the assign values above).

**Task 10**
Create a deep copy of the identity matrix from Task 9 and transform the matrix to the following:

| | | | |
|---|---|---|---|
| 0 | 2 | 2 | 2 |
| 2 | 0 | 2 | 2 |
| 2 | 2 | 0 | 2 |
| 2 | 2 | 2 | 0 |

(There are many different ways to complete this task. Try to do this using element-wise operations such as addition and multiplication).

Verify that the original identity matrix is intact (since you made a deep copy).

**Task 11**
Create a Rock-Paper-Scissors game between a human player and the computer (similar to what we did in class previously). This time, however, use a numpy array to calculate the points that the players get (assume W = 3, D = 1, L = 0). Do not use if statements!

***Task 12**
Assume that the game is played 5 times. Create a numpy array to store all choices AND points of the human player. Do the same for the AI. These arrays represent the players' "memories".

A "memory" array should look something like this (for example):

| | |
|---|---|
| 0 | 3 |
| 1 | 1 |
| 1 | 3 |
| 2 | 0 |
| 0 | 0 |

Where each row represent the round of the game (five rounds played). The first column represents the choices of the player (0-Rock, 1-Paper, 2-Scissors). The second column represent the number of points that the player got in the round (3-Won, 1-Draw, 0-Lost).