VINUNIVERSITY

Lab Practice № 13: Numpy in Python

COMP1010 Introduction to Programming **Week 14**

Lab Practice Submission Instructions:

- This is an individual lab practice and will typically be assigned in the laboratory (computer lab).
- Your program should work correctly on all inputs. If there are any specifications about how the program should be written (or how the output should appear), those specifications should be followed.
- Your code and functions/modules should be appropriately commented. However, try to avoid making your code overly busy (e.g., include a comment on every line).
- Variables and functions should have meaningful names, and code should be organized into functions/methods where appropriate.
- Academic honesty is required in all work you submit to be graded. You should **NOT** copy or share your code with other students to avoid plagiarism issues.
- Use the template provided to prepare your solutions.
- You should upload your .py file(s) to the Canvas **before the end of the laboratory session** unless the instructor gave a specified deadline.
- Submit separate .py file for each Lab problem with the following naming format, for example: **V202000999_Lab13_Q1.py**. **Note:** If you are working on Jupiter Notebook, you need to download/convert it to Python .py file for submission.
- Late submission of lab practice without an approved extension will incur the following penalties:

  (a) No submission by the deadline will incur 0.25 point deduction for each problem (most of the problems are due at the end of the lab session).

  (b) The instructor will deduct an additional 0.25 point per problem for each day past the deadline.

  (c) The penalty will be deducted until the maximum possible score for the lab practice reaches zero (0%) unless otherwise specified by the instructor.

Problem 1 – Rank, Shape and Values of a NumPy array

[CMS is not required] Given an NumPy array `a = np.array([1, 2, 3])`. Write a program to determine the rank, shape of `a`, and the current values of `a[0], a[1], a[2]`?

Problem 2 – Computations on arrays

[CMS is not required] Given the matrix below:

$$A = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}$$

Using NumPy to create the matrix A, then:

(a) Compute sum of all elements of the matrix A.

(b) Compute sum of each column.

(c) Compute sum of each row.

Problem 3 – Python list vs Numpy

[CMS is not required] Given the script code below:

```
x = np.array([[1,2,3], [4,5,6], [7,8,9], [10, 11, 12]])
v = np.array([1,0,1])
y = np.zeros(x.shape)

for i in range(x.shape[0]):
    for j in range(x.shape[1]):
       y[i,j] = # complete
```

(a) Complete the code above in order to add the vector `v` to each row of a matrix `x`.

(b) The previous code works well; however when the matrix `x` is very large, computing an explicit loop in Python could be slow. An alternative is to use NumPy. In this question, you are required to used NumPy to compute `y` from `x` and `v` without explicit loops.

Problem 4 – Calculating the square root

[CMS is not required] Given two NumPy arrays as following:

```
array_1 = numpy.array([[5, 6, 9], [21 ,18, 27]])
array_2 = numpy.array([[15 ,33, 24], [4 ,7, 1]])
```

Create a result array by adding the two NumPy arrays above. Next, modify the result array by calculating the square of each element.