Министерство образования Республики Беларусь

УО «Брестский государственный технический университет»

Кафедра ИИТ

**Лабораторная работа №7**

По дисциплине: “Языки программирования”

Тема: “Изучение NumPy. Сравнение производительности с классическими библиотеками Python”**Вариант №3**

**Выполнил**: студент 2 курса группы ПО-7 Панкратов Роман Сергеевич

**Проверила:** Бойко Д.О.

Брест 2021

### Цель работы

### Изучение NumPy. Сравнение производительности с классическими библиотеками Python

### Общие требования

1. Для написания кода использовать библиотеки классического Python, NumPy и SciPy.
2. Код демонстрируется в Jupyter Notebook
3. По каждому заданию должно быть предоставлено не менее 3-х вариантов решения.
4. Все варианты решения должны быть протестированы на скорость выполнения при помощи %timeit
5. Полученные результаты отразить в отчете и сделать выводы о производительности и комфорте использования NumPy в различных задачах.

### Задания

**Задание №1**

Подсчитать произведение ненулевых элементов на диагонали прямоугольной матрицы.

**Результат**:



**Задание №2**

Дана матрица x и два вектора одинаковой длины i и j. Построить вектор np.array([X[i[0], j[0]], X[i[1], j[1]], . . . , X[i[N-1], j[N-1]]]).  
**Результат**:



**Задание №3**

Даны два вектора x и y. Проверить, задают ли они одно и то же мультимножество.

**Результат**:



**Задание №4**

Найти максимальный элемент в векторе x среди элементов, перед которыми стоит нулевой.

**Результат**:



**Задание №5**

Дан трёхмерный массив, содержащий изображение, размера (height, width, numChannels), а также вектор длины numChannels. Сложить каналы изображения с указанными весами, и вернуть результат в виде матрицы размера (height, width). Считать реальное изображение можно при помощи функции scipy.misc.imread (если изображение не в формате png, установите пакет pillow: conda install pillow).  
Преобразуйте цветное изображение в оттенки серого, использовав коэффициенты np.array([0.299, 0.587, 0.114]).

**Результат**:



**Задание №6**

Реализовать кодирование длин серий (Run-length encoding). Дан вектор x. Необходимо вернуть кортеж из двух векторов одинаковой длины. Первый содержит числа, а второй - сколько раз их нужно повторить.

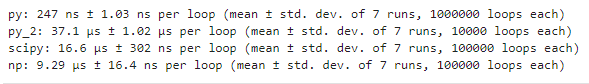
**Результат**:



**Задание №7**

Даны две выборки объектов - X и Y. Вычислить матрицу евклидовых расстояний между объектами. Сравнить с функцией scipy.spatial.distance.euclidean.

**Результат**:



**Задание №8**

Реализовать функцию вычисления логарифма плотности многомерного нормального распределения. Входные параметры: точки X, размер (N, D), мат. ожидание m, вектор длины D, матрица ковариаций C, размер (D, D). Разрешается использовать библиотечные функции для подсчета определителя матрицы, а также обратной матрицы, в том числе в невекторизованном варианте. Сравнить с scipy.stats.multivariate\_normal(m, C).logpdf(X) как по скорости работы, так и по точности вычислений.

**Результат**:



### Вывод:

При небольшом количестве входных данных python превосходит numpy, однако по мере увеличения количества данных numpy все сильнее превосходит python.