

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ
ФГАОУ ВО «СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»
Институт информационных технологий и телекоммуникаций

Кафедра инфокоммуникаций.

Дисциплина: Кроссплатформенное программирование

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №2

Основы работы с библиотекой NumPy

Выполнила:
студентка 2 курса
ИВТ-б-о-19-1
Хубиева Аида

Проверил:
Воронкин
Роман Александрович

Работа защищена с оценкой:

Ставрополь, 2021

Цель работы: исследовать базовые возможности библиотеки NumPy языка программирования Python.

Ход работы:

Ссылка на репозиторий:

<https://github.com/hubieva-a/lab2.1>

Задание №1 (1,5 балла)

Создайте два массива: в первом должны быть четные числа от 2 до 12 включительно, а в другом числа 7, 11, 15, 18, 23, 29.

1. Сложите массивы и возведите элементы получившегося массива в квадрат:

```
In [12]: import numpy as np
a = np.array([2, 4, 6, 8, 10, 12])
b = np.array([7, 11, 15, 18, 23, 29])
c = a + b
c = c * c
print(c)

[ 81  225  441  676 1089 1681]
```

2. Выведите все элементы из первого массива, которые стоят на тех местах, где элементы второго массива больше 12 и дают остаток 3 при делении на 5.

```
In [15]: ind_b = np.logical_and(b>12, b%5==3)
print(ind_b)
print(a[ind_b])
print(a[4])

[False False False  True  True False]
8
10
```

3. Для первого массива найдите остатки от деления на 2, а для второго — на 3. Для каждого получившегося массива выведите его уникальные значения (см. семинар).

```
In [17]: m = a%2
n = b%3
print(m)
print(n)

[0 0 0 0 0 0]
[1 2 0 0 2 2]
```

Рисунок 1. Выполнение Homework_1

Выделите строки, у которых ответов "нет" больше, чем ответов "да":

```
In [64]: # student's code here
import numpy as np

arr = np.array([[0, 1, 1, 0], [0, 0, 0, 1], [1, 1, 0, 1], [1, 1, 1, 1]])
ar = np.array(arr, dtype=bool)

a = np.sum(ar[0, :])
if a > 2:
    print(ar[0, :])
b = np.sum(ar[1, :])
if b > 2:
    print(ar[1, :])
c = np.sum(ar[2, :])
if c > 2:
    print(ar[2, :])
d = np.sum(ar[3, :])
if d > 2:
    print(ar[3, :])

[ True  True False  True]
[ True  True  True  True]
```

Вывести квадраты первых десяти натуральных чисел:

```
In [58]: # student's code here
c = np.array([1, 2, 3, 4, 5, 6, 7, 8, 9, 10])
c2 = c*c
print(c2)

[ 1  4  9 16 25 36 49 64 81 100]
```

Перемешать числа натуральные числа от 1 до 10 (воспользуйтесь np.random.permutation):

```
In [ ]: # помощь по функции
?np.random.permutation
```

```
In [ ]: # student's code here
d = np.random.permutation(10)
print(d)
```

Составить таблицу умножения от 1 до 10:

```
In [57]: # student's code here
r = np.array([1, 2, 3, 4, 5, 6, 7, 8, 9, 10])
col = np.copy(r)
col = col.reshape(10, 1)
table = r * col
print(table)

[[ 1  2  3  4  5  6  7  8  9 10]
 [ 2  4  6  8 10 12 14 16 18 20]
 [ 3  6  9 12 15 18 21 24 27 30]
 [ 4  8 12 16 20 24 28 32 36 40]
 [ 5 10 15 20 25 30 35 40 45 50]
 [ 6 12 18 24 30 36 42 48 54 60]
 [ 7 14 21 28 35 42 49 56 63 70]
 [ 8 16 24 32 40 48 56 64 72 80]
 [ 9 18 27 36 45 54 63 72 81 90]
 [10 20 30 40 50 60 70 80 90 100]]
```

Рисунок 2. Выполнение Homework_2

Для вычисления накопленной надо посчитать средние оценки за проверочные для каждого студента и средние за домашние работы (может пригодиться `np.mean`). Накопленная оценка есть $0.3 \cdot \text{Проверочные} + 0.4 \cdot \text{Домашние} + 0.3 \cdot \text{Коллоквиум}$. Вычислите накопленные оценки и добавьте их в последний столбец матрицы (нужно добавить еще один новый столбец).

```
In [6]: # Веса следует брать из переменной weights, а не вписывать вручную
import numpy as np
weights = [0.3, 0.4, 0.3]

import numpy as np
marks = np.loadtxt("https://raw.githubusercontent.com/nadiinchi/iad2019/master/materials/data_hw1.txt")
marks = marks[:3]

pr = marks[:, 0:3] * weights[0]
hw = marks[:, 3:6] * weights[1]
q = marks[:, 6:7] * weights[2]

pr_student = np.mean(pr, axis = 1, keepdims=True)
hw_student = np.mean(hw, axis = 1, keepdims=True)

the_mark = np.round(pr_student + hw_student + q, 2)
marks = np.hstack((marks, the_mark))
print(marks)

[[ 7.   8.   10.   0.   0.   2.1   1.5   3.23]
 [ 0.   6.   4.4   4.   0.   5.   4.3   3.53]
 [ 3.   7.   8.7   7.6   7.   9.7  10.   8.11]]
```

Рисунок 3. Выполнение Homework_3

Вопросы для защиты работы:

1. Каково назначение библиотеки NumPy?

numpy – это библиотека для языка программирования Python, которая предоставляет в распоряжение разработчика инструменты для эффективной работы с многомерными массивами и высокопроизводительные вычислительные алгоритмы.

2. Что такое массивы Narray?

Многомерный (n-dimensions) массив numpy

3. Как осуществляется доступ к частям многомерного массива?

`given_array[0, 1]` где первое число – номер строки, второе число – номер столбца

4. Как осуществляется расчет статистик по данным?

Argmax -- Индексы элементов с максимальным значением (по осям)

Argmin-- Индексы элементов с минимальным значением (по осям)

max -- Максимальные значения элементов (по осям)

min -- Минимальные значения элементов (по осям)

mean -- Средние значения элементов (по осям)

prod -- Произведение всех элементов (по осям)

std -- Стандартное отклонение (по осям)

sum -- Сумма всех элементов (по осям)

var -- Дисперсия (по осям)

```
>>> m = np.array(m)
```

```
>>> type(m)
```

```
<class 'numpy.ndarray'>
```

5. Как выполняется выборка данных из массивов ndarray?

given_array[0, :] где первое число – номер строки, второе число – номер столбца, : -- выделить всю строку или столбец