**Практическая работа № 2**

***Тема: “Разработка программ, реализующих циклические алгоритмы над списками”.***

**Цель:**

1. Продолжить закрепление навыков разработки программ на языке Python с помощью интерактивной среды IDLE, либо с помощью редакторов Geany, PyCharm;

2. научиться:

* создавать списки;
* работать с нумерованным объектом - списком;
* обращаться к элементу списка по индексу;
* выполнять операции над списками;
* применять при разработке программ специальные функции и методы, работающие со списками.

**Задание на выполнение:**

**ЗАДАНИЕ 1.** (общая) Составьте блок-схему поиска максимального элемента

в одномерном массиве и напишите соответствующую программу, не используя

встроенных методов.

Задача поиска максимального элемента в массиве решается следующим

образом. Пусть maxA — требуемое значение максимального элемента. Сначала

присваиваем переменной maxA значение первого элемента массива, потом

сравниваем первый элемент со следующим. Если следующий элемент (второй) больше

первого, присваиваем его значение переменной maxA, а если нет — переходим к

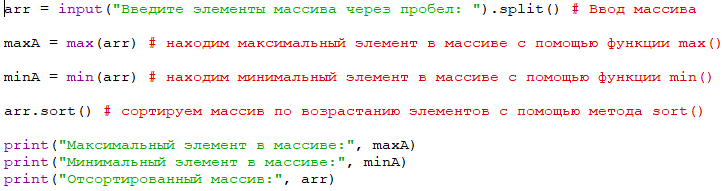
следующему (третьему элементу) и т. д.

Аналогично решается задача поиска минимального элемента в массиве. В

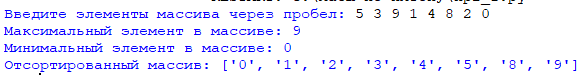
Python эти алгоритмы уже реализованы в функциях max(), min() и в методе sort ()

(метод sort () сортирует список по возрастанию значений элементов).

***Текст программы:***



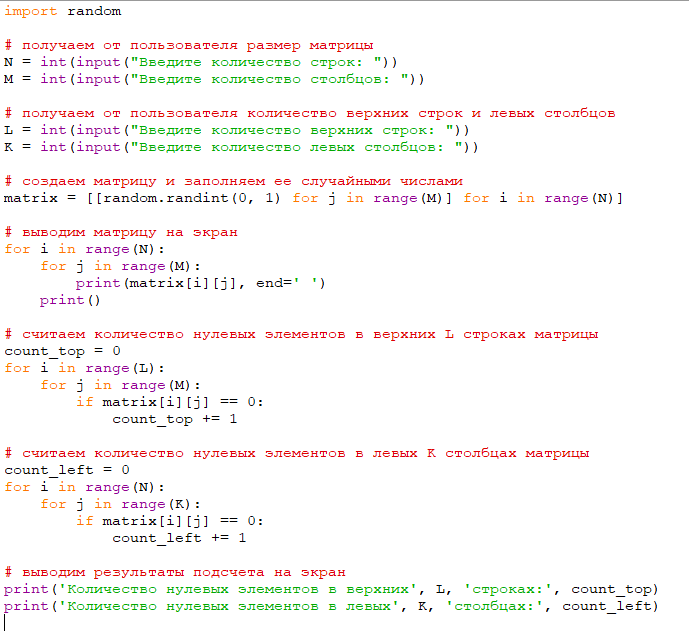
***Работа программы:***



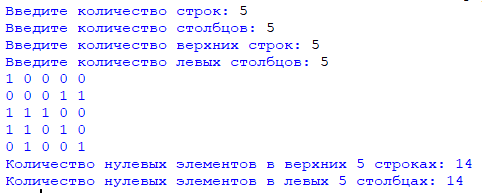
**Задание 2**

Определить, сколько нулевых элементов содержится в верхних L строках матрицы и в левых К столбцах матрицы.

***Текст программы:***

**

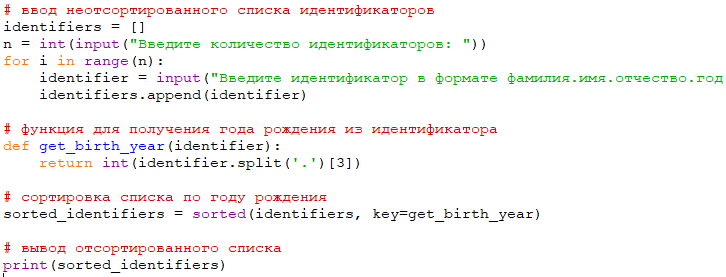
***Работа программы***

******

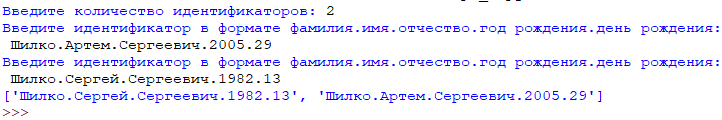
**Задание 3**

Дан неотсортированный список строк некоторых идентификаторов в формате фамилия.имя.отчество.год рождения.день рождения. Необходимо выполнить сортировку массива по значению года рождения, по возрастанию.

***Код программы:***

******

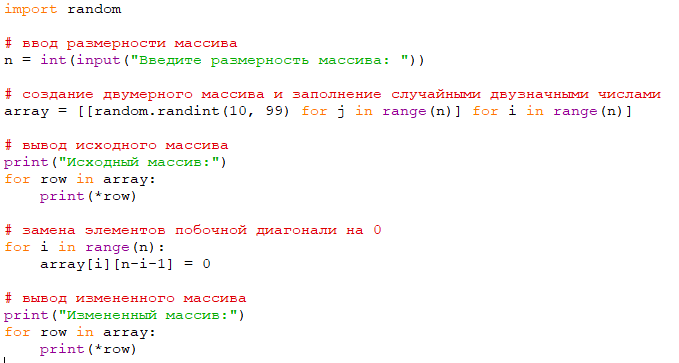
***Работа программы:***

******

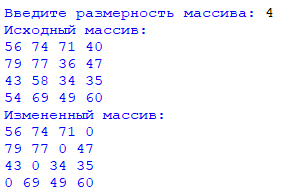
**Задание 4**

Создайте двумерный массив размером n×n и заполните его случайными двузначными числами. Замените все элементы побочной диагонали на 0. Исходный и полученный массивы выведите на экран. Числа в строке разделяйте одним пробелом.

***Код программы:***

******

***Работа программы:***

******

**КОНТРОЛЬНЫЕ ВОПРОСЫ**

1. Что такое список?

список в Python - это упорядоченный набор объектов. В список могут

одновременно входить объекты разных типов (числа, строки и другие структуры,

в частности, списки и кортежи). Объекты, входящие в список, называют элементами

списка. Подчеркнем, любой элемент списка может содержать объект произвольного

типа.

1. Назовите способы задания списков?

В Python есть несколько способов задания списков. Рассмотрим основные из них. С использованием квадратных скобок [].

С использованием функции list().

С использованием генератора списков.

С использованием метода split() для разделения строки на элементы списка.

С использованием конструктора list(), который принимает другой итерируемый объект в качестве аргумента.

Также, списки могут быть созданы при выполнении операций со списками, например при копировании или объединении списков. Такие списки называются производными.

1. Является ли список объектом?

Да, список в Python является объектом.

1. Какую операцию следует использовать для объединения двух списков?

Метод extend()

1. Можно ли изменять списки?

Да можно

1. Каково назначение метода split?

Split используется для разбиения строки с разделителями на подстроки.

1. Каково назначение метода join?

Метод join в Python отвечает за объединение списка строк с помощью определенного указателя.

1. Каково назначение функции range()?

Он генерирует серию целых чисел, от значения start до stop

1. Приведите примеры возможных срезов списка.

my\_list[1:5] - вырезает элементы от второго включительно до шестого не включительно.

my\_list[:3] - вырезает элементы от начала списка до третьего не включительно.

my\_list[3:] - вырезает элементы от четвертого элемента включительно до конца списка.

my\_list[::2] - вырезает каждый второй элемент списка.

1. Что такое многомерный список?

это список, который содержит другие списки в качестве своих элементов. Каждый элемент может быть списком, который сам может содержать другие списки и т.д. Это позволяет создавать структуры данных с более сложной структурой и хранить многомерные данные.

1. Назовите способы задания многомерных списков?

Создание пустого двумерного списка с помощью цикла фор:

my\_list = []

for i in range(rows):

row = []

for j in range(cols):

row.append(0)

my\_list.append(row)

Создание многомерного списка с помощью генератора списков:

my\_list = [[0 for j in range(cols)] for i in range(rows)]

Создание многомерного списка с помощью метода ляст() и функции ранге():

my\_list = list(list(range(cols)) for i in range(rows))

Создание многомерного списка из других списков с помощью оператора \*:

list1 = [1, 2, 3]

list2 = [4, 5, 6]

my\_list = [list1, list2]

Создание многомерного списка с помощью библиотеки нтымпу:

import numpy as np

my\_list = np.zeros((rows, cols))

1. Как обратиться к элементу двумерного списка? Трехмерного?

Чтобы обратиться к элементу двумерного списка в Python, вам нужно указать  индексы элемента, который вы хотите получить.

1. Каково назначение функции zip()?

Функция zip() в Python позволяет объединить два или более итерируемых объекта  (списки, кортежи и т.д.) и создать объект zip, который содержит кортежи из  элементоввзятых из исходных объектов по порядку. Каждый кортеж содержит  элементы на  соответствующих позициях из исходных объектов.

1. Каково назначение функции map()?

Функция map() в путчасон позволяет применить заданную функцию к каждому  элементу итерируемого объекта (списку, кортежу и т.д.) и вернуть новый итератор срезультатами.

1. Где используются lambda-функции?

используются там, где требуется определить простую функцию без необходимости  создания отдельной функции при помощи ключевого слова геф. Они могут  использоваться в качестве аргументов функций высшего порядка, таких как map(),  filter(),  reduce(), и т.д.

1. Как обратиться к элементу главной диагонали матрицы?

 вы можете использовать следующий код:

matrix = [[1, 2, 3], [4, 5, 6], [7, 8, 9]]

diagonal = [matrix[i][i] for i in range(len(matrix))]

print(diagonal) # [1, 5, 9]

1. Каково назначение функций filter() и reduce()?

Функция filter() в путчасон используется для фильтрации элементов итерируемого  объекта (списка, кортежа, и т.д.) на основе заданного условия. Она принимает  функцию в качестве аргумента, которая определяет условие фильтрации. Функция filter() возвращает.

1. Можно ли таким способом создать список размерностью n x m?

A = [[0] \* m] \* n

Yes

1. Если для заполнения массива (n = 5 строк, m = 6 столбцов) использовать генератор [[i \* j for j in range(m)] for i in range(n)], то чему будет равен последний элемент массива?

[[0, 0, 0, 0, 0], [6, 7, 8, 9, 10], [12, 14, 16, 18, 20], [18, 21, 24, 27, 30], [24, 28, 32, 36, 40]]

1. Запишите комментарии к следующему коду:

a = [1, 2, 3]

b = a # значение b? [1,2,3]

a[1] = 10 # значение b? [1,10,3]

b[0] = 20 # значение a? [20,10,3]

a = [5, 6] # значение b? [20,10,3]

1. Какой список будет создан следующим генератором

[[i\*m+j for j in range(m)] for i in range (n)]? (при n=3, m=6)

Список будет выглядеть так: [[0, 1, 2, 3, 4, 5], [6, 7, 8, 9, 10, 11], [12, 13,14, 15, 16, 17]]