**Національний технічний університет України**

**«Київський політехнічний інститут імені Ігоря Сікорського»**

**Факультет інформатики та обчислювальної техніки**

**Кафедра обчислювальної техніки**

**Програмування**

**Лабораторна робота №4**

**«Списки, кортежі, множини і діапазони в Python»**

Виконав:

студент групи ІО-63

Братун А.Ю.

Залікова книжка № 6305

Перевірив Новотарський М. А.

Київ 2016 р.

**Мета:** вивчити способи створення списків, кортежів, множин та задавання діапазонів. Операції над списками, кортежами та діапазонами. Функції для перетворень списків, кортежів та множин.

**Завдання:**

1. Вивчити матеріал лекцій 11, 12, 13 та 14.

2. Виконати індивідуальне завдання лабораторної роботи, вибране відповідно до варіанту.

**Теоретичні основи:**

Списки є змінюваними типами даних.

Створити список можна такими способами: 1. За допомогою функції list([]). 2. Перелічивши всі елементи списку всередині квадратних дужок. 3. Застосувавши метод append() для заповнення списку поелементно. 4. За допомогою генератора списків.

1.Присвоювання значень елементам списку по індексу: x[0 1 ] = (Ця операція не застосовна до рядка!)

2. Присвоювання з кінця за від’ємним індексом: y[ 1] 2 − =

3. Присвоювання за допомогою зрізу

4. Оператори + та +=

5. Оператори in та not in

Додавання й видалення елементів списку

append () - додає один об'єкт у кінець списку.

ехtеnd() - додає елементи послідовності в кінець списку.

Оператор += - додає елементи за допомогою операції конкатенації.

insert (, ) - додає один об'єкт у зазначену позицію.

рор([]) – видаляє елемент, розташований по зазначеному індексу, і повертає його.

remove () - видаляє перший елемент, який містить зазначене значення.

clear()- видаляє всі елементи списку, очищаючи його.

**Методи**

index()- повертає індекс елемента всередині списку.

count()- повертає загальну кількість елементів

**Перевертання й перемішування списку**

reverse() змінює порядок проходження елементів списку на протилежний.

rеvеrsеd()- змінює порядок проходження та створює новий список.

shuffle( [, ] ) - з модуля random «перемішує» список випадковим чином.

**Вибір елементів випадковим чином**

Одержати елементи зі списку випадковим чином дозволяють наступні функції з модуля random:

choice () - повертає випадковий елемент із будь- якої послідовності.

sample(, ) – повертає список із зазначеної кількості елементів.

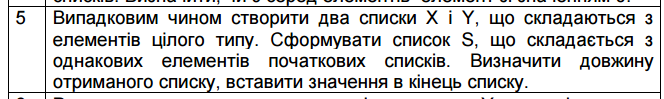
**Сортування списку**

Відсортувати список дозволяє метод sort():

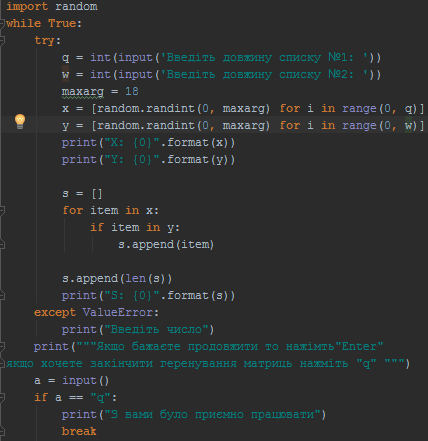
sort([key=None][, reverse= False])

Функція sоrtеd([, key=None] [, reverse= False]) – формує новий відсортований список, а початковий залишає без змін.

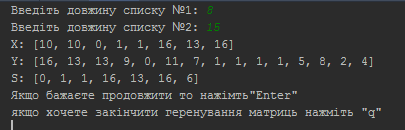
**Завдання 1**

****

**Роздруківка того фрагменту тексту програми, який написаний індивідуально**

****

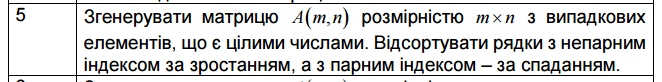
**Роздруківка результатів виконання програми з контрольним прикладом.**

****

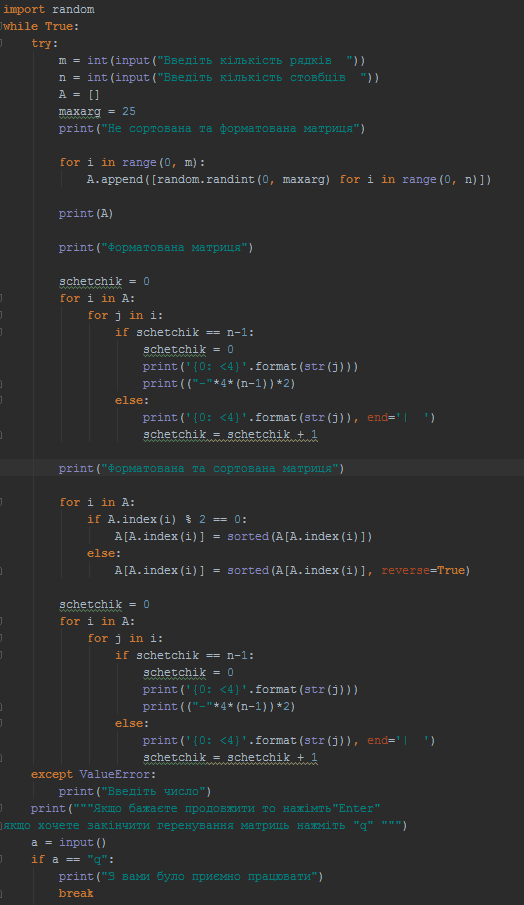
**Алгоритм роботи програми**

* Ввід даних
* Генерація 2-х списків
* Використання циклу **for** для пошуку подібних елементів в обох списках
* Якщо такі елементи є, добавляємо їх в новий список
* В кінець списку записуємо його довжину
* Вивід двох початкових списків і список з подібними елементами

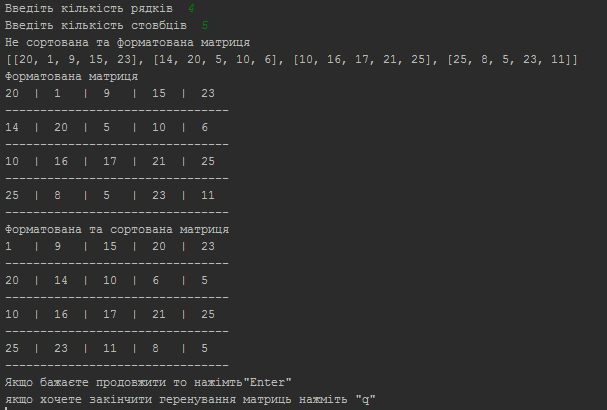
**Завдання 2**

****

**Роздруківка того фрагменту тексту програми, який написаний індивідуально**

****

**Роздруківка результатів виконання програми з контрольним прикладом.**

****

**Алгоритм роботи програми**

* Ввід даних
* Генерація не форматованого та не сортованого списку
* Форматування матриці за допомогою циклу **for**
* Сортування не форматованої матриці за допомогою циклу **for** та форматування сортованої матриці за допомогою циклу **for**
* Вивід форматованої, сортованої та початкової матриці

**Протягом виконня роботи складнощів не було**