

Лабораторна робота №4

Тема: «Списки, кортежі, множини і діапазони в Python».

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:

1. **Списки** є змінюваними типами даних.
2. **Кортежі** є незмінюваними типами даних.
3. **Множини** можуть бути як змінюваними, так і незмінюваними.
4. **Діапазони** є наборами чисел, сформованими на основі заданих початкового, кінцевого значень і величини кроку між числами.

Створити список можна такими способами:

1. За допомогою функції `list([<Послідовність>])`.
2. Перелічивши всі елементи списку всередині квадратних дужок.
3. Застосувавши метод `append()` для заповнення списку поелементно.
4. За допомогою генератора списків.

Створити копію списку можна такими способами:

1. **Перший спосіб** – застосуванням операції добування зрізу.
2. **Другий спосіб** – використанням функції `list()`,
3. **Третій спосіб** – застосуванням методу `copy()`.
4. Щоб одержати глибоку копію списку, слід скористатися функцією `deepcopy()` з модуля `copy`.

Операції над списками

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:

Операція добування зрізу: `[<Початок>:<Кінець>:<Крок>]`

```
>>> arr = [2, 6, 10, 14, 18, 22]
>>> arr[1:3] = [8,7]
>>> arr
[2, 8, 7, 14, 18, 22]
```

4. Оператори `+` та `+=`
5. Оператори `in` та `not in`

Багатовимірні списки

Будь-який елемент списку може містити список.

Створити вкладений список можна, наприклад, так:

Двовимірний список:

```
>>> n = [[1,2,3], [4,5,6],[7,8,9]]
>>> n
[[1, 2, 3], [4, 5, 6], [7, 8, 9]]
>>>n[0][0]
1
>>>n[2][1]
8
```

Тривимірний список:

```
>>> t = [[[1,2],[3,4]],[[5,6],[7,8]]]
>>> t
[[[1, 2], [3, 4]], [[5, 6], [7, 8]]]
>>> t[0][0][0]
1
>>> t[1][1][0]
7
```

Чотиривимірний список:

```
>>> f = [[[[1,2],[3,4]],[[5,6],[7,8]]],
[[[9,10],[11,12]],[[13,14],[15,16]]]]
>>> f[0][0][0][0]
1
>>> f[0][1][1][0]
7
>>> f[1][1][1][1]
16
```

Перебір елементів списку

1. За допомогою циклу for: `for i in arr: print(i, end=" ")`
2. За допомогою циклу while.
3. За допомогою генераторів списків

```
>>> arr = [i for i in range(1,10)]
>>> arr
[1, 2, 3, 4, 5, 6, 7, 8, 9]
```

Функція `map()`

Дозволяє застосувати функцію до кожного елемента послідовності

Повертає об'єкт, що підтримує ітерації. Щоб одержати список, необхідно результат передати у функцію `list()`.

```
map(<Функція>, <Послідовність1>[, ..., <Послідовність N>])
```

Функція `zip()`

Вбудована функція `zip()` на кожній ітерації повертає кортеж, що містить елементи послідовностей, які розташовані на однаковому зсуві.

`zip(<Послідовність1>[, ... , <ПослідовністьN>])`

Функція `filter()`

Дозволяє виконати перевірку елементів послідовності.

`filter(<Функція>, <Послідовність>)`

Функція повертає об'єкт, що підтримує ітерації. Щоб одержати список, необхідно результат передати у функцію `list()`.

Функція `reduce()`

Функція `reduce()` з модуля `functools` застосовує зазначену функцію до пар елементів і накопичує результат.

`reduce(<Функція>, <Послідовність>[, <Початкове значення>])`

Додавання й видалення елементів списку

`append (<Об'єкт>)` – додає один об'єкт у кінець списку.

`extend(<Послідовність>)` – додає елементи послідовності в кінець списку.

Оператор `+=` – додає елементи за допомогою операції конкатенації.

`insert (<Індекс>, <Об'єкт>)` – додає один об'єкт у зазначену позицію.

`pop([<Індекс>])` – видаляє елемент, розташований по зазначеному індексу, і повертає його.

`remove (<Значення>)` – видаляє перший елемент, який містить зазначене значення.

`clear()` – видаляє всі елементи списку, очищаючи його.

Методи

`index()` – повертає індекс елемента всередині списку.

`index(<Значення>[, <Початок>[, <Кінець>]])`

`count(<Значення>)` – повертає загальну кількість елементів

Функції

`max()` і `min()` – визначають максимальне й мінімальне значення списку відповідно.

`any (<Послідовність>)` – повертає значення `True`, якщо в послідовності існує хоч один елемент, який у логічному контексті повертає значення `True`.

`all (<Послідовність>)` – повертає значення `True`, якщо всі елементи послідовності в логічному контексті повертають значення `True` або послідовність не містить елементів.

Перевертання й перемішування списку

`reverse()` змінює порядок проходження елементів списку на протилежний.

`reversed(<Послідовність>)` – змінює порядок проходження та створює новий список.

`shuffle(<Список> [, <Число від 0.0 до 1.0>])` – з модуля `random` «перемішує» список випадковим чином.

Вибір елементів випадковим чином

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

`choice (<Послідовність>)` – повертає випадковий елемент із будь-якої послідовності.

`sample(<Послідовність>, <Кількість елементів>)` – повертає список із зазначеної кількості елементів.

Сортування списку

Відсортувати список дозволяє метод `sort()`:

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

Функція `sorted(<Послідовність>[, key=None] [, reverse=False])` – формує новий відсортований список, а початковий залишає без змін.

Заповнення списку числами:

```
>>> list(range(11))  
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
```

Функція `sample(<Послідовність>, <Кількість елементів>)` з модуля `random`. Формує список з випадковими числами (або випадковими елементами з іншого списку)

Перетворення списку на рядок

Метод `join()`:

`<Рядок> = <Роздільник>.join(<Послідовність>)`

Кортежі

Створити кортеж можна такими способами:

1. за допомогою функції `tuple([<Послідовність>])`
2. Указавши всі елементи через кому всередині круглих дужок (або без дужок):

Кортежі підтримують функції:

`len()`, `min()`, `max()`, методи `index()` і `count()`

Множини

Оголосити множину можна за допомогою функції `set()`.

Одержати кількість елементів множини дозволяє функція `len()`.

Оператори `|` і `union()` – поєднують дві множини.

Оператори `a |= b` і `a.update(b)` – додають елементи множини `b` у множину `a`.

Оператори `-` і `difference()` – обчислюють різницю множин.

Оператори `a -= b` і `a.difference_update(b)` – видаляють елементи з множини `a`, які існують і в множині `a`, і в множині `b`.

Оператори `&` і `intersection()` – виконують перетин множин. Дозволяють одержати елементи, які існують в обох множинах.

Оператор `^` і `symmetric_difference()` – повертають усі елементи обох множин, крім елементів, які присутні в обох цих множинах.

Оператори `a ^= b` і `a.symmetric_difference_update(b)` – у множині `a` будуть усі елементи обох множин, крім тих, що містяться в обох цих множинах.

Оператор `in` – перевірка наявності елемента в множині.

Оператор `not in` – перевірка відсутності елемента в множині.

Оператор `==` – перевірка на рівність.

Оператор `a <= b` і `a.issubset(b)` – перевіряють, чи входять усі елементи множини `a` в множину `b`.

Оператор `a < b` – перевіряє, чи входять усі елементи множини `a` в множину `b`, причому множина `a` не повинна дорівнювати множині `b`.

Оператори `a >= b` і `a.issuperset(b)` – перевіряють, чи входять усі елементи множини `b` у множину `a`.

Оператор `a > b` – перевіряє, чи входять усі елементи множини `b` у множину `a`, причому множина `a` не повинна дорівнювати множині `b`.

Оператор `a.isdisjoint(b)` – перевіряє, чи є множини `a` й `b` повністю різними, тобто такими, що не містять жодного однакового елемента.

Методи

Метод `copy()` – створює копію множини.

Метод `add(<Елемент>)` – додає `<Елемент>` у множину.

Метод `remove(<Елемент>)` – видаляє `<Елемент>` із множини.

Метод `discard(<Елемент>)` – видаляє `<Елемент>` із множини, якщо він присутній.

Метод `pop()` – видаляє довільний елемент із множини й повертає його.

Метод `clear()` – видаляє всі елементи із множини.

Тип множин `frozenset`. На відміну від типу `set`, множину типу `frozenset` не можна змінити.

Оголосити множину можна за допомогою функції `frozenset()`.

Функція `frozenset()` дозволяє перетворити елементи послідовності в множину.

Множини `frozenset` підтримують оператори, які не змінюють саму множину, а також наступні методи:

`copy()`, `difference()`, `intersection()`, `issubset()`,
`issuperset()`, `symmetric_difference()` та `union()`.

Діапазони

Для створення діапазону застосовується функція `range()`:

`range([<Початок>,] <Кінець> [, <Крок>])`

Діапазони підтримують:

- доступ до елемента по індексу,
- одержання зрізу (у результаті повертається також діапазон),
- перевірку на входження
- перевірку на невходження,
- функції `len()`, `min()`, `max()`,
- методи `index()` і `count()`.

Оператор `==` – повертає `True`, якщо діапазони рівні, і `False` в протилежному випадку.

Оператор `!=` – повертає `True`, якщо діапазони не рівні, і `False` в протилежному випадку.

Модуль `itertools`

Перш ніж використовувати функції, необхідно підключити модуль за допомогою інструкції: `import itertools`

1. Функція `count([start=0][, step=1])`.

Створює нескінченно наростаючу послідовність значень.

2. Функція `cycle(<Послідовність>)`

На кожній ітерації повертається черговий елемент послідовності.

3. Функція `repeat(<Об'єкт>[, <Кількість повторів>])`

Повертає об'єкт зазначену кількість разів.

Основні поняття про матрицю

Нехай є дві скінченні множини:

$$M = \{1, 2, \dots, m\} \text{ і } N = \{1, 2, \dots, n\}$$

де m і n — натуральні числа.

Розглянемо таблицю, яка складається з m рядків та n стовпців.

Кожен елемент цієї таблиці характеризується двома індексами: i та j , які визначають положення даного елемента в таблиці. Будемо позначати через a_{ij} елемент, який знаходиться на перетині i -го рядка та j -го стовпця. Далі будемо називати елемент a_{ij} елементом матриці, а саму таблицю — матрицею A .

Отже, матриця $m \times n$ має такий вигляд:

$$A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{m,1} & a_{m,2} & \dots & a_{m,n} \end{pmatrix}$$

В мові програмування Python матриця A може бути представлена двовимірним списком або кортежем. Оскільки [нумерація елементів у списках починається з нуля](#), то індекси елементів матриці A матимуть такий вигляд:

$$A_{\text{Python}} = \begin{pmatrix} a_{00} & a_{01} & \dots & a_{0n-1} \\ a_{10} & a_{11} & \dots & a_{1n-1} \\ \dots & \dots & \dots & \dots \\ a_{m-1,0} & a_{m-1,1} & \dots & a_{m-1,n-1} \end{pmatrix}$$

Отже, для задавання матриці $A(m, n)$ розмірності $m \times n$ потрібно створити список з діапазоном індексів $(0 \dots (m-1))$, кожен елемент якого є списком, що містить елементи з діапазоном індексів $(0 \dots (n-1))$

Приклад, що буде корисний при виконанні лабораторної роботи:

Спосіб "виокремлення" цифр з числа, якщо відома кількість розрядів (цифр) числа.

1. Залишок від ділення на 10 початкового числа дає останню цифру числа.
2. Якщо знайти залишок від ділення на 100 початкового числа, то ми отримаємо останні дві цифри числа. Далі можна просто розділити отримане двозначне число без остачі на 10, і у нас виявиться друга цифра числа.
3. Якщо початкове тризначне число розділити без остачі на 100, то отримаємо першу цифру числа.

```
n = input("Введіть тризначне число: ")
n = int(n)

d1 = n % 10
d2 = n % 100 // 10
d3 = n // 100

print("Сума цифр числа:", d1 + d2 + d3)
```

Результат роботи скрипта:

```
Введіть тризначне число: 742
Сума цифр числа: 13
```

Індивідуальні завдання

Завдання 1

Відповідно до номера в списку вибрати індивідуальне завдання. Написати програму мовою Python. Забезпечити ввід даних з клавіатури комп'ютера та друк результатів обчислень. У звіті до лабораторної роботи описати алгоритм, за яким побудована програма. **При виводі даних обов'язково використати форматування.**

№	Завдання
1	Випадковим чином створити два цілочисельних списки. Сформувати новий список, на парних місцях якого будуть елементи з непарними індексами з першого списку, а на непарних – з парними індексами з другого. Сформувати з нового списку вдвічі коротший список з випадковими елементами. Перетворити цей список на рядок.
2	Випадковим чином створити список, що складається з елементів цілого типу. Отримати новий список, який складається з цифр, що знаходяться в молодших розрядах елементів початкового списку. Сформувати з нього список з випадковими елементами, вдвічі довший за початковий.

№	Завдання
3	Випадковим чином створити цілочисельний список, що складається з двозначних цілих чисел. Обчислити суму чисел цього списку. Суму чисел списку вставити в середину початкового списку, якщо його довжина – парне число, і в кінець, якщо його довжина непарна.
4	Випадковим чином створити два списки дійсних чисел. Сформувати третій список з упорядкованих за зростанням значень елементів обох списків. Визначити, чи є серед елементів елемент зі значенням 0.
5	Випадковим чином створити два списки X і Y, що складаються з елементів цілого типу. Сформувати список S, що складається з однакових елементів початкових списків. Визначити довжину отриманого списку, вставити значення в кінець списку.
6	Розрахувати значення елементів списку Y за формулою $y_i = i^2 - 2i + 19.3 \cos i$. Сформувати новий список, розмістивши в ньому спочатку елементи, значення яких менші за середнє арифметичне, а потім – ті значення, що дорівнюють або більші за середнє арифметичне елементів початкового списку. Вивести обидва списки.
7	Випадковим чином створити список дійсних чисел. Визначити різницю між сумою елементів з парними індексами і сумою елементів, індекси яких кратні трьом. Вставити цю різницю в кінець списку.
8	Випадковим чином створити цілочисельний список. Визначте індекси найбільшого з непарних за значенням додатних елементів та найбільшого з парних за значенням додатних елементів. Знайдіть індекс елемента, найближчого за значенням до різниці двох попередніх елементів. Видаліть цей елемент, а також сформууйте новий список з тих елементів, що у сумі з виведеним перевищують число 10.
9	Випадковим чином створити список $X = [x_1, x_2, \dots]$, який складається з елементів цілого типу. Розрахувати елементи списку $Y = [y_1, y_2, \dots]$ за формулою $y_i = \cos x_i^2 + 2.971 \lg^2 i^2$. Сформувати третій список з упорядкованих за спаданням значень з обох списків. Вибрати із сформованого списку випадковим чином два елементи і визначити, чи входять вони у список X.
10	Випадковим чином створити список $X = [x_1, x_2, \dots]$, що складається з елементів цілого типу. Розрахувати елементи списку $Y = [y_1, y_2, \dots]$ за формулою: $y_i = \begin{cases} (x_i)^3 - 7.5 & \text{якщо } \cos(x_i) > 0, \\ x_i^2 - 5e^{\sin(x_i)} & \text{якщо } \cos(x_i) \leq 0. \end{cases}$ Впорядкувати список Y за зростанням, список X за спаданням і сформувати новий список R, елементами якого є парні за індексами елементи списків X і Y. Вибрати зі списку R випадковим чином один елемент і визначити, чи входить він у список Y.

№	Завдання
11	Випадковим чином створити список, що складається з двозначних чисел цілого типу. Отримати новий список, що складається з сум цифр відповідних елементів початкового списку. Створити список, елементами якого є кортежі, що складаються з елементів, які знаходяться на однаковому зсуві з елементами початкових списків.
12	Випадковим чином створити три списки, що складаються 1) з довільних додатних елементів дійсного типу, 2) з довільних елементів цілого типу, 3) з елементів логічного типу. Створити список, що утворюється додаванням елементів першого та другого списків, які знаходяться на однаковому зсуві, якщо елемент третього списку, який знаходиться на тому ж зсуві, дорівнює <code>True</code> , та відніманням у протилежному випадку.
13	Випадковим чином створити цілочисельний список. Обчислити суму цифр елементів. Якщо сума цифр елемента виявиться непарною, то замінити нею дане число. Сформувати з отриманого списку список з випадковими елементами, вдвічі коротший за початковий.
14	Випадковим чином створити три списки, що складаються з довільних елементів дійсного типу. Замінити місцями першу половину першого списку на другу симетрично. Створити список, що утворюється добутком суми елементів першого та другого списків, які знаходяться на однаковому зсуві, на елемент третього списку, який знаходиться на тому ж зсуві.
15	Випадковим чином створити список дійсних чисел. Визначити елемент списку (значення і індекс), який найбільш віддалений від заданого дійсного числа S . Перемішати список випадковим чином, визначити елемент списку (значення і індекс), який найбільш віддалений від цього ж заданого дійсного числа S .
16	Випадковим чином створити цілочисельний список. Визначити суму і кількість елементів, величина яких є меншою за середнє арифметичне елементів списку. Дописати значення суми і кількість елементів як елементи в кінець початкового списку та сформувати третій список з тих елементів, які дорівнюють середньому арифметичному елементів одержаного списку та початкового списку.
17	Визначити кількість локальних мінімумів у випадковим чином створеному списку чисел. (Локальний мінімум в списку чисел – це послідовність трьох чисел, що стоять поруч та в яких середнє число менше за тих, що стоять зліва і справа від нього). Перемішати список випадковим чином, визначити кількість локальних мінімумів в отриманому списку чисел. Якщо перший локальний мінімум більший за другий, то сформувати список з елементами, що містять кортежі з елементами першого та другого списку, які знаходяться на однаковому зсуві. У протилежному випадку сформувати третій список з сумами елементів першого та другого списку.

№	Завдання
18	<p>Випадковим чином створити три цілочисельних списки. Створити список, що утворюється піднесенням до степеня різниці елементів першого та другого списків, які знаходяться на однаковому зсуві. Значення степеня дорівнює елементу третього списку, який знаходиться на тому ж зсуві. (Наприклад: перший список: 8, 9. Другий список: 10, 5. Третій список: 2, 3. Створений список: перший елемент: $(8-10)^{**2}$, другий елемент: $(9-5)^{**3}$)</p>
19	<p>Створити випадковим чином список з додатних, від'ємних і нульових чисел. Визначити суму елементів. Якщо одержана сума буде додатною, то вивести послідовність значень елементів, які розташовані між першим від'ємним і першим нульовим елементами. Якщо одержана сума буде від'ємною, то сформувати третій список, кожен елемент якого буде дорівнювати середньому арифметичному від елементів створеного списку.</p>
20	<p>У створеному випадковим чином списку чисел визначити послідовності чисел, які монотонно зменшуються (кожне наступне число менше за попереднє). Створити списки, елементами якого є кортежі, кожен з яких містить індекс та значення відповідного елемента. Якщо сума елементів послідовності виявиться парною, то для кожної з них створити третій список, що містить як елементи суму елементів одержаних кортежів. В протилежному випадку суму замінити різницею.</p>
21	<p>У створеному випадковим чином списку цілих чисел видалити ті елементи, які зустрічаються більше, ніж двічі. Створити списки, відсортовані за зростанням та за спаданням. З даних списків сформувати третій список, кожен елемент якого дорівнює середньому геометричному двох попередніх елементів, які знаходяться на однаковому зсуві.</p>
22	<p>Сформувати випадковим чином список, що складається з елементів цілого типу та сформувати новий список, розташувавши спочатку всі від'ємні елементи, нульові, а далі – додатні, зберігаючи порядок їх слідування в початковому списку. Сформувати кортеж, що містить як елементи кількість додатних чисел, кількість нулів та кількість від'ємних чисел. Якщо сума першого та другого елементу цього кортежу буде парною, то сформувати третій список з подвоєними елементами початкового списку. У протилежному випадку третій список повинен містити елементи, які є остачами від ділення першого списку на 2.</p>
23	<p>Сформувати випадковим чином список X, що складається з елементів цілого типу. Обчислити елементи списку Y за формулою</p>

№	Завдання
	$y_i = x_i^2 + 0.3$ і знайти Р за формулою $P = \frac{x_1 y_1 \cdot x_3 y_3 \cdot \dots \cdot x_9 y_9}{x_0 y_0 \cdot x_2 y_2 \cdot \dots \cdot x_8 y_8}.$ Визначити залишок від ділення. Записати його після кожного непарного числа у списку Х.
24	Випадковим чином сформувати список, що складається з двозначних чисел цілого типу. Отримати новий список, який складається з різниць цифр елементів початкового списку. Випадковим чином вибрати елемент з новоутвореного списку та видалити перший елемент списку, який містить це значення.
25	Випадковим чином сформувати список, що складається з елементів цілого типу. Впорядкувати список так, щоб всі від'ємні числа були розташовані з початку за зростанням, а всі додатні – в кінці за спаданням. Після ділянки з від'ємними числами вставити як елемент списку їх кількість.
26	Випадковим чином сформувати два списки дійсних чисел. Замінити нулями ті елементи першого списку, які є у другому. Створити список, елементами якого є кортежі, що складаються з елементів цих двох списків, які знаходяться на однаковому зсуві.
27	Випадковим чином заданий цілочисельний список. Визначити кількість ділянок списку, на яких елементи монотонно зростають (кожне наступне число більше за попереднє). Після кожної з ділянок вставити як елемент списку кількість елементів цієї ділянки та знайти суму елементів одержаного списку.
28	Випадковим чином створити два цілочисельні списки. Створити список, елементами якого є залишки від ділення елементів з парними індексами одного списку на елементи з непарними індексами другого. Впорядкувати його за зростанням та визначити суму елементів, додавши її вкінці списку.
29	Випадковим чином створити цілочисельний список. Визначити процентний вміст елементів, що перевищують середнє значення суми всіх елементів списку. Вставити це значення в кінець списку.
30	Випадковим чином створити два списки дійсних чисел. Визначити максимальні елементи в кожному списку і поміняти їх місцями.

Завдання 2

Відповідно до номера в списку вибрати індивідуальне завдання. Написати програму мовою Python. Забезпечити ввід даних з клавіатури комп'ютера та друк результатів обчислень. У звіті до лабораторної роботи описати алгоритм, за яким побудована програма. **При виводі даних обов'язково використати форматування.**

№вар	Завдання
1	Згенерувати матрицю $A(m,n)$ розмірністю $m \times n$ з випадкових елементів, що є цілими числами від 0 до 10. Видалити рядки, які містять елементи, що повторюються.
2	Згенерувати матрицю $A(m,n)$ розмірністю $m \times n$ з випадкових елементів, що є від'ємними та додатними цілими числами. Побудувати список, елементами якого є сума модулів від'ємних чисел у кожному рядку матриці $A(m,n)$
3	Згенерувати матрицю $A(m,n)$ розмірністю $m \times n$ з випадкових елементів, що є цілими числами. Побудувати список, елементами якого є середнє арифметичне рядків матриці $A(m,n)$
4	Згенерувати матрицю $A(m,n)$ розмірністю $m \times n$ з випадкових елементів, що є цілими числами. Знайти мінімальний елемент на кожній з діагоналей матриці.
5	Згенерувати матрицю $A(m,n)$ розмірністю $m \times n$ з випадкових елементів, що є цілими числами. Відсортувати рядки з непарним індексом за зростанням, а з парним індексом – за спаданням.
6	Згенерувати матрицю $A(m,n)$ розмірністю $m \times n$ з випадкових елементів, що є цілими числами. Знайти мінімальні елементи в кожному стовпці та рядку матриці.
7	Згенерувати матрицю $A(m,n)$ розмірністю $m \times n$ з випадкових елементів, що є цілими числами. Поміняти місцями мінімальний та максимальний елементи у кожному стовпці матриці
8	Згенерувати матрицю $A(m,n)$ розмірністю $m \times n$ з випадкових елементів, що є дійсними числами. Знайти спочатку мінімальні елементи в кожному рядку матриці, а потім обчислити їх добуток.
9	Згенерувати матрицю $A(m,n)$ розмірністю $m \times n$ з випадкових елементів, що є дійсними числами. Визначити середнє арифметичне кожного стовпця, визначити максимум і мінімум кожного рядка.
10	Згенерувати матрицю $A(m,n)$ розмірністю $m \times n$ з випадкових елементів, що є цілими числами. Визначити кількість непарних елементів кожного стовпця.
11	Згенерувати матрицю $A(m,n)$ розмірністю $m \times n$ з випадкових елементів, що є цілими числами. Знайти кількість парних і непарних чисел у матриці.
12	Згенерувати матрицю $A(m,n)$ розмірністю $m \times n$ з випадкових елементів, що є цілими числами. Підібрати спільний дільник для

№вар	Завдання
	всіх елементів матриці таким чином, щоб одержати принаймні дві остачі від ділення, які дорівнюють 2.
13	Згенерувати матрицю $A(m,n)$ розмірністю $m \times n$ з випадкових елементів, що є цілими числами. Знайти рядок матриці, у якому середнє арифметичне є найбільшим.
14	Згенерувати матрицю $A(m,n)$ розмірністю $m \times n$ з випадкових елементів, що є цілими числами. Сформувати нову матрицю, кожен елемент якої є середнім арифметичним даного та попереднього рядка. Для першого рядка одержати середнє арифметичне з останнім.
15	Згенерувати квадратну матрицю $A(m,m)$ розмірністю $m \times m$ з випадкових елементів, що є цілими числами. Сформувати нову матрицю, рядками якої будуть упорядковані за зростанням стовпці початкової матриці.
16	Згенерувати квадратну матрицю $A(m,m)$ розмірністю $m \times m$ з випадкових елементів, що є цілими числами. Сформувати нову матрицю, стовпцями якої будуть упорядковані за спаданням рядки початкової матриці.
17	Згенерувати квадратну матрицю $A(m,m)$ розмірністю $m \times m$ з випадкових елементів, що є цілими числами. Поміняти місцями елементи, які стоять на діагоналях на відповідних місцях.
18	Згенерувати квадратну матрицю $A(m,m)$ розмірністю $m \times m$ з випадкових елементів, що є додатними та від'ємними цілими числами. Поміняти знаки чисел у кожному рядку та сформувати нову матрицю, у якій кожний стовець відповідає рядку початкової матриці після зміни знаків.
19	Згенерувати квадратну матрицю $A(m,m)$ розмірністю $m \times m$ з випадкових елементів, що є цілими числами. Знайти рядок з максимальним елементом та стовець з мінімальним елементом. Поміняти місцями знайдений рядок зі знайденим стовпцем.
20	Згенерувати матрицю $A(m,m)$ розмірністю з випадкових елементів, що є цілими числами. Замінити максимальний елемент у кожному рядку на середнє арифметичне стовпця, який містить даний максимальний елемент.
21	Згенерувати матрицю $A(m,n)$ розмірністю $m \times n$ з випадкових елементів, що є цілими числами. Замінити мінімальний елемент у кожному стовпці на середнє арифметичне рядка, який містить даний мінімальний елемент.

№вар	Завдання
22	Згенерувати матрицю $A(m,n)$ розмірністю $m \times n$ з випадкових елементів, що є цілими числами. Знайти мінімальне значення серед сум елементів рядків матриці і замінити ним максимальний елемент у кожному рядку.
23	Згенерувати матрицю $A(m,n)$ розмірністю $m \times n$ з випадкових елементів, що є цілими числами. Порахувати кількість елементів у кожному рядку матриці та кількість елементів у всій матриці, які перевищують середнє арифметичне матриці.
24	Згенерувати матрицю $A(m,n)$ розмірністю $m \times n$ та матрицю $B(n,m)$ розмірністю $n \times m$ з випадкових елементів, що є цілими числами. Сформувати матрицю $C(m,n)$, кожен елемент якої є остачею від ділення розміщених на відповідних позиціях елементів рядка матриці A на елементи стовпця матриці B
25	Згенерувати матриці $A(m,n)$ та $B(m,n)$ розмірністю $m \times n$. Визначити середнє арифметичне для кожного стовпця матриці A та поділити на нього елементи відповідного рядка матриці B
26	Згенерувати матрицю $A(m,n)$ розмірністю $m \times n$ з випадкових елементів, що є цілими двохранрядними числами. Сформувати нову матрицю, кожен елемент якої містить число, яке є остачею від ділення елемента матриці A на суму його цифр.
27	Згенерувати матрицю $A(m,m)$ розмірністю $m \times m$ з випадкових елементів, що є цілими двохранрядними числами. Знайти максимальний елемент на головній діагоналі та поділити на цей елемент всі елементи матриці, крім нього самого.
28	Згенерувати матрицю $A(m,n)$ розмірністю $m \times n$ з випадкових елементів, що є цілими двохранрядними числами. Знайти суму непарних елементів у кожному рядку і поділити її на кожен з парних елементів у цьому ж рядку.
29	Згенерувати матрицю $A(m,n)$ розмірністю $m \times n$ з випадкових елементів, що є цілими числами. Знайти суму елементів матриці і створити нову матрицю, кожен елемент якої дорівнює відсотку, який складає відповідний елемент матриці A по відношенню до суми її елементів.
30	Згенерувати матриці $A(m,m)$ та $B(m,m)$ розмірністю $m \times m$. Сформувати матрицю C , кожен елемент якої дорівнює середньому арифметичному між відповідними елементами стовпця матриці A та рядка матриці B .

Зміст звіту:

1. Титульний лист повинен мати такий вигляд:

Національний технічний університет України
«Київський політехнічний інститут»
Факультет інформатики та обчислювальної техніки
Кафедра обчислювальної техніки

Програмування
Лабораторна робота №4
«Списки, кортежі, множини і діапазони в Python»

Виконав:
студент групи ІО-__

Залікова книжка № ____
Перевірів Новотарський М.А.

Київ 2016р.

2. Мета лабораторної роботи та загальне завдання
3. Короткі теоретичні відомості, які відображають операції, функції та методи обробки даних, які **були використані** при написанні лабораторної роботи.
4. Роздруківка того фрагменту тексту програми, який написаний індивідуально.
5. Роздруківка результатів виконання програми з контрольним прикладом.
6. Аналіз результатів та висновки.

Контрольні питання (не відображаються у звіті)

1. Створення списків.
2. Операції над списками
3. Багатовимірні списки
4. Перебір елементів списку
5. Додавання й видалення елементів списку
6. Перевертання й перемішування списку
7. Вибір елементів випадковим чином
8. Кортежі
9. Операції над множинами
10. Діапазони