# Лабораторна робота 7 Обробка списків в Python

**Мета роботи:** Ознайомитись з технологією обробки списків в Python. Побудувати блок-схеми алгоритмів, розробити програми з використанням функцій і методів обробки списків у вигляді одновимірних масивів та матриць.

## Рейтинг лабораторної роботи №7

|  |  |  |  |
| --- | --- | --- | --- |
| № п.п | Вид діяльності студента | Рейтинговий бал | Deadline |
| 1 | Написання коду | 2 | Листопад |
| 2 | Захист роботи | 1 |
| 3 | Звіт з роботи | 1 |
| 4 | Якість роботи | 1 |
| Разом за роботу | | 5 |  |

## Теоретичний матеріал

1. Список у Python – це впорядкована колекція (тобто індексована послідовність) об'єктів довільних типів (в більшості мов програмування використовується термін "масив")
2. Список являє собою послідовність елементів, пронумерованих від 0, як символи в рядку.
3. Списки у Python належать до **змінюваних (mutable) типів.**
4. Списки можуть складатися з різних об’єктів: чисел, рядків та інших списків, в останньому випадку списки називають вкладеними.
5. Списки служать для того, щоб зберігати об’єкти в певному порядку, особливо якщо порядок або вміст можуть змінюватися.
6. Одне і те саме значення може зустрічатися в списку кілька разів.
7. Операції зі списками:
   * змінити список,
   * додати в нього нові елементи,
   * видалити або перезаписати існуючі елементи,
   * змінити кількість елементів у списку,
   * змінити окремі елементи.

### Методи роботи зі списками

|  |  |
| --- | --- |
| Метод | Призначення |
| list.append(x) | Додає елемент в кінець списку |
| list.extend(L) | Розширює список list, додаючи в кінець все елементи списку L |
| list.insert(i, x) | Вставляє значення x на i-ий індекс |
| list.remove(x) | Видаляє перший елемент у списку, який має значення x. Повертає ValueError, якщо такого елемента не існує |
| list.pop([i]) | Видаляє i-ий елемент і повертає його. Якщо індекс не вказано, видаляється останній елемент |
| list.index(x, [start [, end]]) | Повертає положення першого елемента зі значенням x (при цьому пошук ведеться від start до end) |
| list.count(x) | Повертає кількість елементів зі значенням x |
| list.sort([key=функция]) | Сортує список на основі функції |
| list.reverse() | Перевертає список |
| list.copy() | Копія списку |
| list.clear() | Очищає список |

#### Введення списків

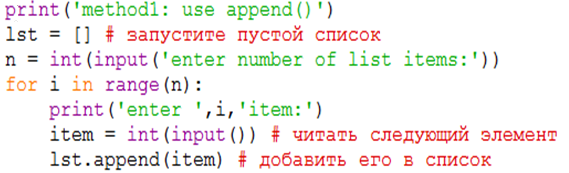


Рис.1. Введення списку з клавіатури через метод append()

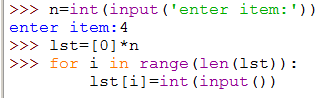


Рис.2. Введення списку з клавіатури через операцію присвоєння

#### Генерація списків

1. Для генерації застосовують функції **randint()** або **random()** модуля **random**
2. Генерація списку може здійснюватися циклос з умовою або без неї

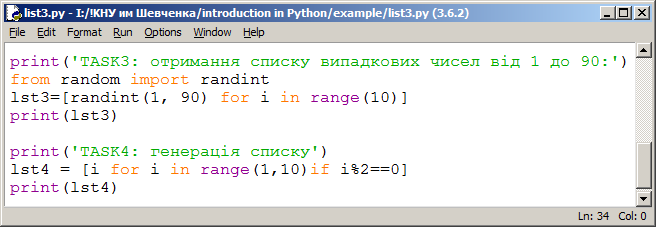


Рис.3. Приклади генерації списку

#### Операції над списками

##### Конкатенація, множення на ціле число та функція len()

Конкатенація двох списків утворює новий список, що складається з першого списку до кінця якого дописано другий список.

Множення списку на ціле число утворює новий список, до кінця якого дописано той самий список.

Функція len() повертає кількість елементів (довжину) списку, який передається як аргумент функції

##### Перевірка входження елементу у список

Операція **in** повертає значення **True**, якщо елемент **x** входить до списку **lst.**

Операція **not in** повертає значення **True**, якщо елемент **x** не входить до списку **lst**

##### Присвоєння списків

Операція **=** повертає копію списку, при цьому зміна елементів одного списку **приводить** до зміни відповідного елемента копії. Операція **=** зрізу всього списку повертає копію списку, при цьому зміна елементів одного списку **не приводить** до зміни відповідного елемента зрізу (тобто копії) списку.

#### Приклад програми1 обробки списків (одновимірних масивів)

##### Умова

1. Ввести список цілих чисел з клавіатури.
2. Вивести усі непарні числа та числа, що мають парні індекси.
3. Обчислити суму від'ємних елементів списку.
4. Знайти перший додатний елемент списку та його індекс.
5. Знайти мінімальний додатний елемент списку та його індекс.

##### Код

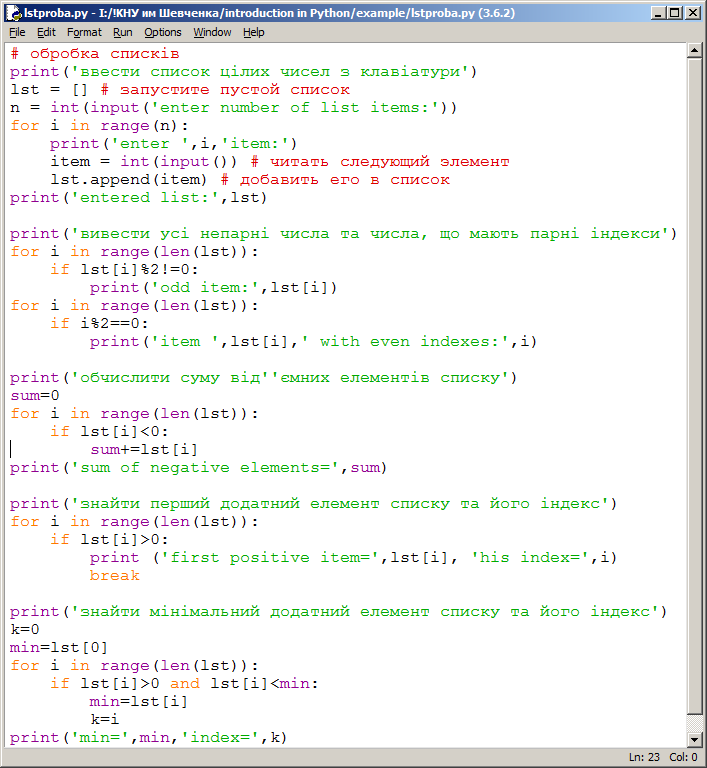


Рис.4. Приклад коду програми.

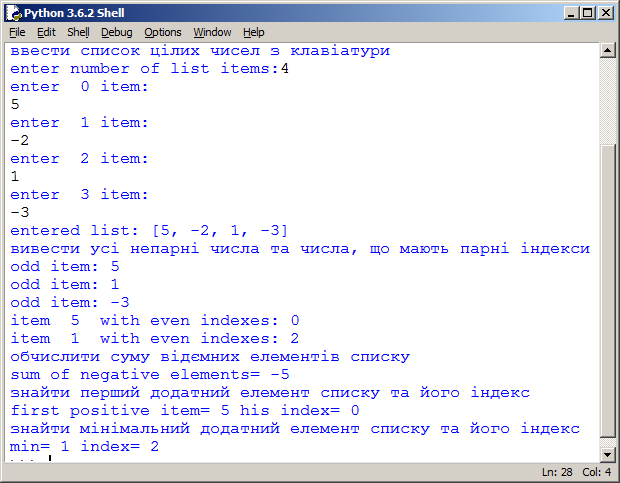


Рис.5. Результат роботи програми.

#### Приклад програми2 обробки матриць (двовимірних списків)

##### Умова

Перемножити дві матриці, які вводяться з клавіатури. У випадку несумісності матриць (кількість стовпців першої матриці не дорівнює кількості рядків другої) вивести відповідне повідомлення.

##### Код

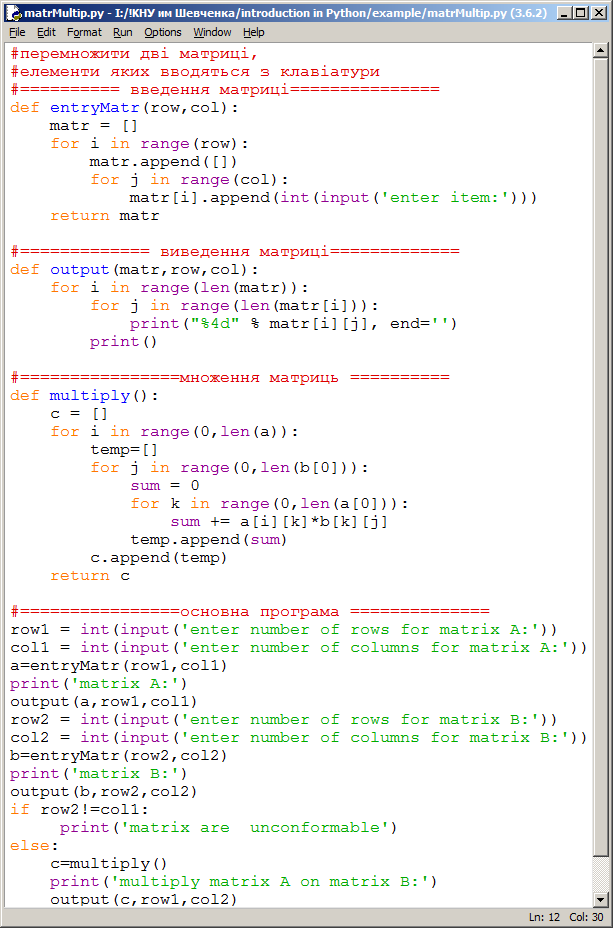


Рис. 6 Код програми множення двох матриць

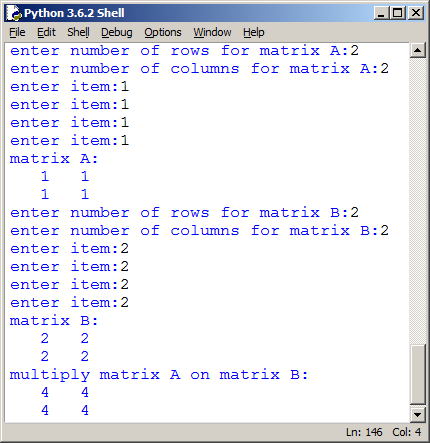


Рис. 7. Результат роботи програми множеня матриць

## Завдання для самостійної роботи за варіантами

Кожне завдання складається з двох задач: обробка одновимірних масивів (списків), обробка матриць (складених списків). Завдання має бути реалізовано у вигляді одної програми, яка має меню вибору задач. Код завдання повинний складатися з функцій, наприклад, введення вхідних даних, виведення результатів, розрахунок результату тощо.

### Варіанти завдань

1. **Скласти програму для обробки одновимірного масиву та матриці. Передбачити меню вибору задач.**
2. Ввести список чисел з клавіатури. Елементи списку є додатними, від’ємними та рівними нулю числами. З введеного списку сформувати новий список, взявши в нього тільки ті елементи із вхідного списку, які більше за модулем введеного з клавіатури числа M. Вивести на екран число M, вхідний та отриманий списки.
3. Побудувати квадратну матрицю, в якій елементи на головній діагоналі, що проходить з лівого верхнього кута в правий нижній мають значення 1, елементи, що знаходяться вище головної діагоналі - значення 0, елементи, що знаходяться нижче головної діагоналі - значення 2.
4. **Скласти програму для обробки одновимірного масиву та матриці. Передбачити меню вибору задач.**
5. Створити список чисел, кількість елементів якого задана користувачем. Визначити максимальне та мінімальне значення серед елементів із парними та непарними індексами. Вивести список, мінімальний, максимальний елементи та їх індекси.
6. Ввести вимірність і значення елементів квадратної матриці з клавіатури. Перевірити, чи є ця матриця симетричною відносно головної діагоналі. Вивести слово "YES", якщо матриця симетрична, і слово "NO" в іншому випадку.
7. **Скласти програму для обробки одновимірного масиву та матриці. Передбачити меню вибору задач.**
8. Створити числовий список, кількість елементів якого задана користувачем. Обчислити суму елементів між максимальним та мінімальними значеннями списку. Вивести на екран список, максимальний та мінімальний елементи масиву та їх індекси.
9. Згенерувати матрицю розміром n × m. Значення n, m вводяться з клавіатури. Побудувати транспоновану матрицю і вивести її на екран. В транспонованій матриці рядки вихідного масиву стають стовпцями транспонованого, стовпці вихідного масиву стають рядками транспонованого. Рішення оформити у вигляді функції, яка отримує на вхід даний масив і повертає новий масив.
10. **Скласти програму для обробки одновимірного масиву та матриці. Передбачити меню вибору задач.**
11. Створити два числових списки, кількість елементів яких задана користувачем. Знайти найменший серед тих елементів першого списку, які співпадають із значеннями елементів другого списку. Вивести на екран списки, найменший елемент та його індекс.
12. Ввести вимірність і квадратну матрицю з клавіатури. Поміняти місцями елементи, які стоять на головній і побічній діагоналях, при цьому в кожному стовпці потрібно поміняти місцями елементи на головній діагоналі і на побічній діагоналях .Рішення оформите у вигляді функції.
13. **Скласти програму для обробки одновимірного масиву та матриці. Передбачити меню вибору задач.**
14. Створити список цілих чисел. Побудувати новий список із елементів першого списку. В новому списку спочатку стоять числа, що діляться на 2, потім ті, що діляться на 2 та 3, потім на 3. Надрукувати вхідний та вихідний списки.
15. У лівому верхньому кутку дошки розміром n × m розташований шаховий король. За один хід він може зробити хід вправо, вниз або на одну клітину по діагоналі вправо-вниз. Порахуйте кількість маршрутів, що веде з лівого верхнього кута дошки в правий нижній. Програма отримує на вхід розміри дошки (числа n і m) і повинна вивести кількість шуканих маршрутів.
16. **Скласти програму для обробки одновимірного масиву та матриці. Передбачити меню вибору задач.**
17. Створити список. Обчислити кількість елементів у найдовшій серії. Серія — це послідовність однакових елементів, розташованих поспіль. Надрукувати список, усі серії однакових елементів та довжину найдовшої серії.
18. Згенерувати матрицю розміром n × m. Повернути її на 90 градусів за годинниковою стрілкою, записавши результат в нову матрицю розміром m × n. Вивести згенеровану та повернуту матриці. Рішення оформите у вигляді функції, яка отримує на вхід дану матрицю і повертає нову матрицю.
19. **Скласти програму для обробки одновимірного масиву та матриці. Передбачити меню вибору задач**
20. У згенерованому числовому списку знайти найбільший серед від’ємних та найменший серед додатних елементів списку. Вивести список, значення знайдених елементів та їх індекси.
21. В кінотеатрі n рядів по m місць в кожному. У двовимірному масиві зберігається інформація про продані квитки: число 1 означає, що квиток на дане місце вже проданий, число 0 означає, що місце вільне. Надійшов запит на продаж k квитків на сусідні місця в одному ряду. Визначте, чи можна виконати такий запит.
22. **Скласти програму для обробки одновимірного масиву та матриці. Передбачити меню вибору задач**
23. Згенерувати список з цілих чисел та вивести його на екран. Знайти максимальний елемент та його індекс та надрукувати їх. Вивести на екран кількість елементів, які менші за значення максимального і більше за значення максимального елемента.
24. У заданій квадратній матриці значення деяких діагональних елементів дорівнюють нулю. Переставити рядки або стовпці матриці таким чином, щоб діагональні елементи стали ненульовими. Якщо це неможливо зробити, вивести відповідне повідомлення.
25. **Скласти програму для обробки одновимірного масиву та матриці. Передбачити меню вибору задач**
26. Згенерувати два упорядкованих за зростанням числових списків. Побудувати упорядкований за зростанням третій список злиттям двох згенерованих.
27. Для заданої цілочислової матриці визначити індекси та значення її сідлових точок. Елемент матриці називається сідловою точкою, якщо він є найменшим у своєму рядку та найбільшим у своєму стовпчику.
28. **Скласти програму для обробки одновимірного масиву та матриці. Передбачити меню вибору задач**
29. Ввести з клавіатури два списки цілих чисел. Кількість елементів списків вводиться з клавіатури. Побудувати третій список, в якому кожний елемент дорівнює найбільшому спільному дільнику відповідних елементів введених списків. Використати алгоритм Евкліда. Надрукувати усі три списки.
30. Задана дійсна квадратна матриця розмірності n\*n. Знайти суми тих елементів, які розміщені відповідно нижче, вище та на головній діагоналі, в рядках, що починаються з від’ємного елементу.
31. **Скласти програму для обробки одновимірного масиву та матриці. Передбачити меню вибору задач**
32. Створити числовий список, кількість елементів якого вводиться користувачем. Значення елементів в списку можуть повторюватися. Визначити індекс першого та останнього входження заданого числа в список. Знайти елемент, значення якого повторюється найбільшу кількість разів. Вивести на екран список, шуканий елемент та його індекс.
33. Для заданої матриці та вектора довільної вимірності здійснити множення матриці та вектор та вектора на матрицю. У разі несумісності вимірності матриці та вектора вивести відповідне повідомлення.
34. **Скласти програму для обробки одновимірного масиву та матриці. Передбачити меню вибору задач**
35. Створити список, значеннями елементів якого є коефіцієнти многочлена I:\!TeachingStudentKPI2018\course1_A&P\A&Psemestr1\LabA&P_semestr1\MethodLabWorkC++Semestr1Html\labs\lab_07\pictures\var11.gif. Знайти значення многочлена при введеному з клавіатури аргументу *x* та похідної від многочлена *A*(*x*) при заданому *x* . Вивести на екран список, значення многочлена та його похідної.
36. В заданій матриці довільної вимірності обчислити суму елементів парних рядків, найбільші значення у непарних рядках, середнє арифметичне серед від’ємних елементів матриці та індекси нульових елементів.
37. **Скласти програму для обробки одновимірного масиву та матриці. Передбачити меню вибору задач**
38. Згенерувати список цілих чисел, кількість елементів якого ввести з клавіатури. Вивести список на екран. Отримати інший список, що складається тільки з непарних чисел вихідного списку або повідомити, що таких чисел немає. Отриманий список вивести на екран в порядку спадання елементів.
39. Згенерувати дійсну матриця розміром n х m Переставляючи її рядки і стовпці, домогтися того, щоб найбільший елемент (або один з них) перемістився у верхній лівий кут. Ввести на екран згенеровану та отриману матриці.
40. **Скласти програму для обробки одновимірного масиву та матриці. Передбачити меню вибору задач**
41. Ввести з клавіатури список цілих чисел. Знайти в списку все прості числа і скопіювати їх в новий список. Надрукувати вхідний та вихідний списки.
42. . значення. Знайти найбільший по модулю елемент. Отримати квадратну матрицю порядку n – 1 шляхом відалення з вихідної матриці рядки і стовпці, на перетині яких розташований елемент зі знайденим значенням. Надрукувати вхідну та отриману матриці.
43. **Скласти програму для обробки одновимірного масиву та матриці. Передбачити меню вибору задач**
44. Згенерувати список чисел, кількість елементів якого ввести з клавіатури. У списку чисел перевірити, чи всі елементи є унікальними, тобто кожне число зустрічається тільки один раз. Надрукувати згенерований список та повідомлення «Усі елементи унікальні» або «Не усі елементи унікальні».
45. Латинським квадратом порядку n називається квадратна матриця розміром n x n, в якій кожний рядок і кожний стовпець містять усі числа від 1 до n. Перевірити, чи є задана матриця латинським квадратом. Надрукувати вхідну матрицю і відповідний висновок.
46. **Скласти програму для обробки одновимірного масиву та матриці. Передбачити меню вибору задач**
47. Згенерувати числовий список, кількість елементів якого ввести з клавіатури. Визначити індекси елементів списку, значення яких не менше заданого мінімуму і не більше заданого максимуму. Вивести на екран список, кількість, індекси та значення елементів, що належать заданому діапазону.
48. Заповнити матрицю випадковими цілими числами. Надрукувати матрицю. Знайти максимальний серед елементів тих стовпців, які впорядковані або за зростанням, або за спаданням. Якщо впорядковані стовпці в матриці відсутні, то вивести відповідне повідомлення.
49. **Скласти програму для обробки одновимірного масиву та матриці. Передбачити меню вибору задач**
50. Згенерувати числовий список з елементами від -100 до +100. Надрукувати список. Вивести послідовність елементів між першим від’ємним і нульовим елементом, між першим додатним і нульовим елементом. Якщо таких елементів немає, по вивести відповідне повідомлення.
51. Ввести розмірність та згенерувати матрицю. У кожному стовпці і кожному рядку матриці мають міститися строго по одному нульового елементу. Перестановкою рядків домогтися розташування всіх нулів на головній діагоналі матриці.
52. **Скласти програму для обробки одновимірного масиву та матриці. Передбачити меню вибору задач**
53. Ввести з клавіатури числовий список з трьох елементів. У кожного елемента списку є вага, яка збільшується на одиницю, коли користувач вибирає цей елемент списку. Початкове значення ваг дорівнює нулю. Користувач має n спроб вибрати кожний елемент списку. Вибір елементів здійснюється випадково за допомогою генератора псевдовипадкових чисел. При випаданні потрібного елемента, його вага збільшується, отже в наступних проходах він буде випадати частіше. Вивести на екран значення ваг елементів після n спроб вибрати їх.
54. Згенерувати квадратну матрицю, задавши її вимірність з клавіатури. Привести матрицю до верхньої трикутної шляхом застосування допустимих перетворень матриць. У верхній трикутній матриці всі елементи, що знаходяться нижче головної діагоналі, дорівнюють нулю
55. **Скласти програму для обробки одновимірного масиву та матриці. Передбачити меню вибору задач**
56. Ввести з клавіатури список чисел. Визначити, скільки в цьому списку елементів, які більше двох своїх сусідів справа та зліва. Вивести кількість таких елементів. Крайні елементи списку и не враховуються, оскільки у них недостатньо сусідів.
57. Згенерувати матрицю розміром N \* N, що містить додатні та від’ємні числа. Здійснити пошук такої підматриці, сума елементів якої є максимальною.
58. **Скласти програму для обробки одновимірного масиву та матриці. Передбачити меню вибору задач**
59. Згенерувати список чисел, кількість елементів якого ввести з клавіатури. Вивести згенерований список. Обчислити кількість пар елементів, рівних один одному. Вважається, що будь-які два елементи, які рівні один одному, утворюють одну пару. Вивести індекси елементів, що утворюють пару та кількість пар.
60. Дана дійсна квадратна матриця порядку 2n. Отримати нову матрицю, переставляючи її блоки розміру n × n за годинниковою стрілкою, починаючи з блоку в лівому верхньому кутку.
61. **Скласти програму для обробки одновимірного масиву та матриці. Передбачити меню вибору задач**
62. Згенерувати список цілих додатних та від’ємних чисел, кількість елементів якого ввести з клавіатури. Знайти суму елементів, розташованих до першого від’ємного елементу та добуток елементів, розташованих після останнього від’ємного елемента. Якщо від’ємних елементів немає, то видати відповідне повідомлення.
63. Дана дійсна квадратна матриця порядку 2n. Отримати нову матрицю, переставляючи її блоки розміру n × n хрест-навхрест
64. **Скласти програму для обробки одновимірного масиву та матриці. Передбачити меню вибору задач**
65. Ввести з клавіатури список чисел. Видалити зі списку всі числа, які повторюються більше двох разів. Надрукувати вхідний список і список після видалення елементів.
66. Задати цілі додатні числа M, N, число D і набір з M чисел. Сформувати матрицю розміру M × N, у якій перший стовпець збігається з вихідним набором чисел, а елементи кожного наступного стовпця дорівнюють сумі відповідного елемента попереднього стовпчика і числа D (в результаті кожен рядок матриці буде містити елементи арифметичної прогресії).
67. **Скласти програму для обробки одновимірного масиву та матриці. Передбачити меню вибору задач**
68. Згенерувати список цілих чисел в діапазоні від -50 до +50, кількість елементів задати з клавіатури. Із отриманого списку скопіювати в інший список елементи, значення яких знаходяться в діапазоні від -5 до 5 включно і підрахувати їх кількість. Надрукувати згенерований і побудований списки.
69. Задати матрицю цілих чисел розміру M × N. Знайти номер першого з її рядків, що містять однакову кількість додатних і від’ємних елементів (нульові елементи матриці не враховуються). Якщо таких рядків немає, то вивести відповідне повідомлення.
70. **Скласти програму для обробки одновимірного масиву та матриці. Передбачити меню вибору задач**
71. Дан список цілих чисел. Розмір списку і його елементи вводяться користувачем. Упорядкувати за зростанням елементи списку, що розташовані до максимального елемента, і за спаданням – ті елементи, що розташовані після максимального елементу. Надрукувати список до та після сортування.
72. Задати квадратну матрицю порядку M. Обнулити елементи матриці, що лежать одночасно нижче головної діагоналі (включаючи цю діагональ) і вище побічної діагоналі (також включаючи цю діагональ). Умовний оператор не використовувати. Надрукувати вхідну та отриману матрицю
73. **Скласти програму для обробки одновимірного масиву та матриці. Передбачити меню вибору задач**
74. Дан список дійсних чисел. Розмір списку і значення його елементів вводяться користувачем. Знайти суму всіх локальних мінімумів списку. Локальний мінімум – це елемент списку, який менший за значенням двох сусідніх елементів даного списку (виключаючи крайні елементи). Надрукувати вхідний список, всі його локальні мінімуми та їх індекси.
75. Задати матрицю розміру M × N. Упорядкувати її рядки так, щоб їх перші елементи утворювали зростаючу послідовність. Надрукувати вхідну та упорядковану матриці.

## Список літератури

1. <https://github.com/tkovalyuk/Basics-of-programming>
2. **МакГрат М**. Программирование на Python для начинающих. –Москва: Эксмо. – 192 с.
3. **Лутц М.** Изучаем Python, 4-е издание. – Пер. с англ. – СПб.: Символ-Плюс, 2011. – 1280 с.
4. **Лутц М.** Программирование на Python, том I, 2. – Пер. с англ. – СПб.: Символ-Плюс, 2011. – 992 с.
5. **Васильев А. Н.** Python на примерах. Практический курс ·по программированию. - СПб.:Наука и Техника, 2016. - 432 с.
6. **Доусон М.** Программируем на Python. - СПб.: Питер, 2014. - 416 с.
7. **Рейтц К., Шлюссер Т.** Автостопом по Python. — СПб.: Питер, 2017. — 336 с.