**ЛАБОРАТОРНА РОБОТА №1.**

**Тема:** Введення у python. Арифметичні вирази, управляючі конструкції та масиви у мові Python

**Мета:** навчитися створювати найпростіші програми на мові Python,  
використовуючи оператори вибору і циклів, арифметичні вирази та масиви

**Теоретичні відомості**

**Python** – високорівнева мова програмування загального призначення,  
орієнтована на підвищення продуктивності розробника і читання коду.  
Python підтримує декілька парадигм програмування, в тому числі  
структурну, об'єктно-орієнтовану, функціональну, імперативну та аспектно-  
орієнтовану.

**Основні принципи синтаксису мови Python:**  
1. Кінець рядка є кінцем інструкції (крапка з комою не потрібна).

х = 5

print(2 + х)

2. Вкладені інструкції об'єднуються у блоки за величиною відступів.  
Відступ може бути будь-яким, головне, щоб в межах одного вкладеного блоку  
відступ був однаковий (рекомендується робити відступ 4 пробіли).

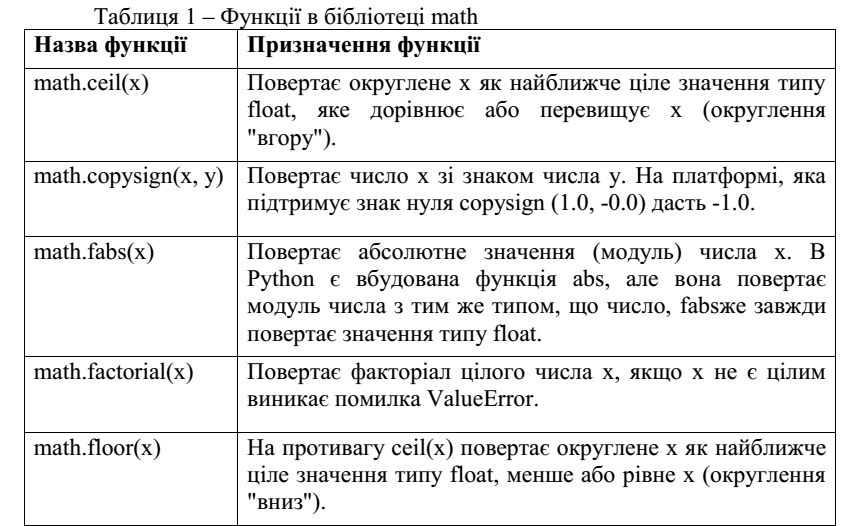
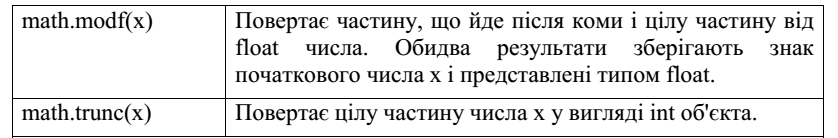
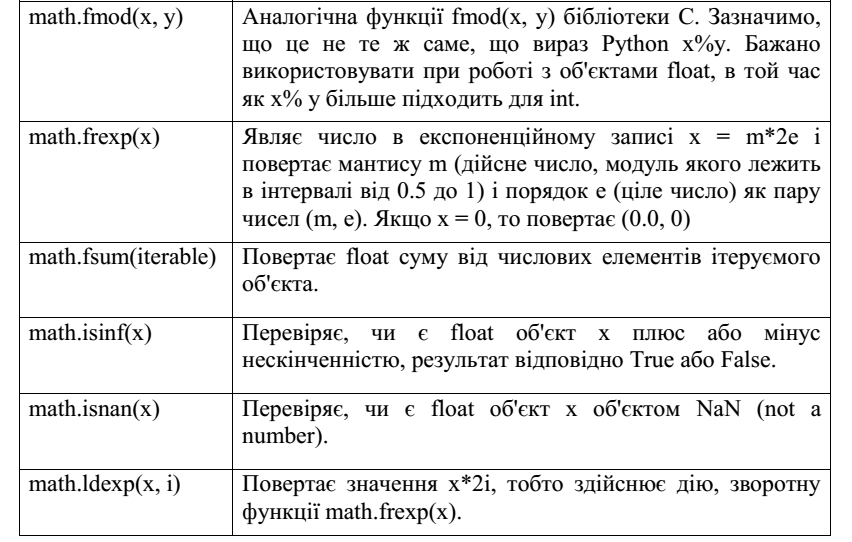
if х == 10:print('yes')

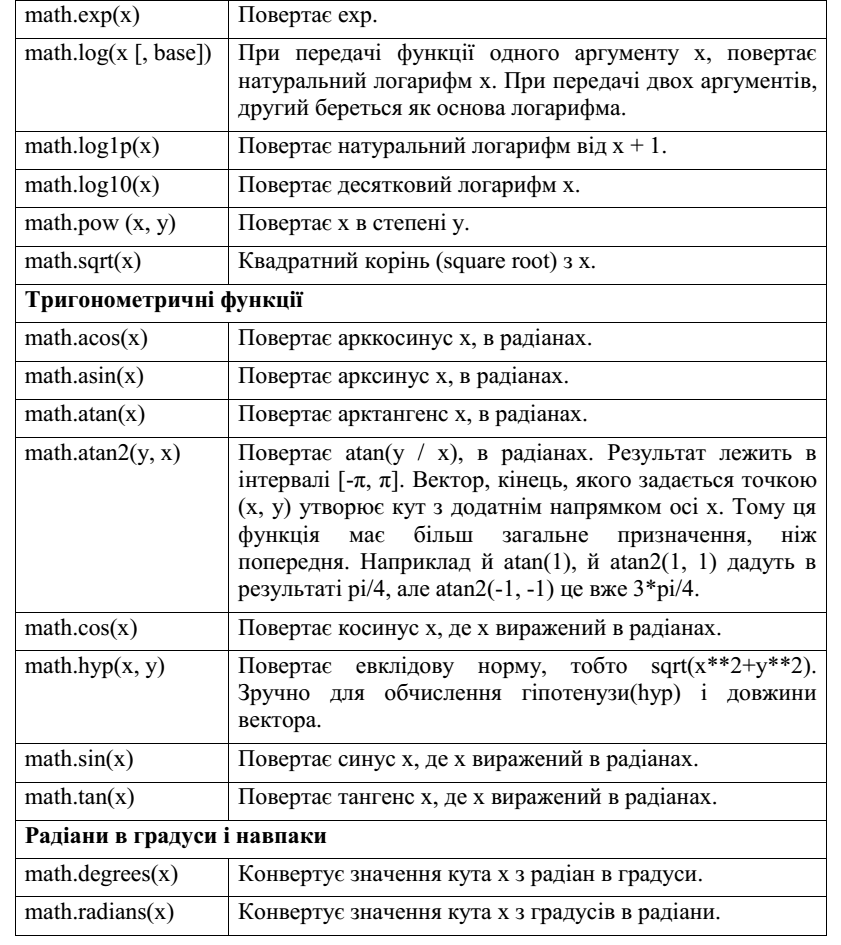
3. Вкладені інструкції в Python записуються відповідно до одного і того ж шаблону, коли основна інструкція завершується двокрапкою, слідом за чим  
розташовується вкладений блок коду, зазвичай з відступом під рядком основної інструкції.

Розглянемо на прикладах основи роботи з Python.

**#Базові типи даних (вказуються неявно)**  
a = 5 # int  
b = 7.0 # float  
c = 2>4 # boolean  
d = "World" # string  
e = 1.5 + 0.5j # complex  
print(type(a))  
print(type(b))  
print(type(c))  
print(type(d))  
print(type(e))  
4  
print(a, b, c, d, e.real, e.imag)  
**#Перетворення типів даних**  
int(True) # == 1  
float(True) # == 1.0  
str(True) # = ‘True’  
bool(0) # == False  
bool(0.0) # == False  
bool(1) # == True  
bool(10) # == True  
**В Python є особливий спосіб обміну змінних значеннями:**  
(a, b) = (b, a)  
Він використовується дуже часто. Даний метод працює завжди, навіть  
якщо змінні різних типів (в цьому випадку вони обмінюються не тільки  
значеннями, але і типами). Круглі дужки в цьому записі можна опустити:  
a, b = b, a  
**#Основні арифметичні операції**  
a = 5 # int  
b = 7.0 # float  
c = 1 + 2 # 3  
d = 5 - 3 # 2  
e = a \* b # 35.0  
f = 3.0 / 2 # 1.5  
g = 3 / 2 # 1  
h = 5 % 3 # 2  
j = 10\*\*7.3 # 19952623.1497  
print(type(a))  
print(type(b))  
print(type(c))  
print(type(d))  
print(type(e))  
print(type(f))  
print(type(g))  
print(type(h))  
print(type(j))  
print(a, b, c, d, e, f, g, h, j)

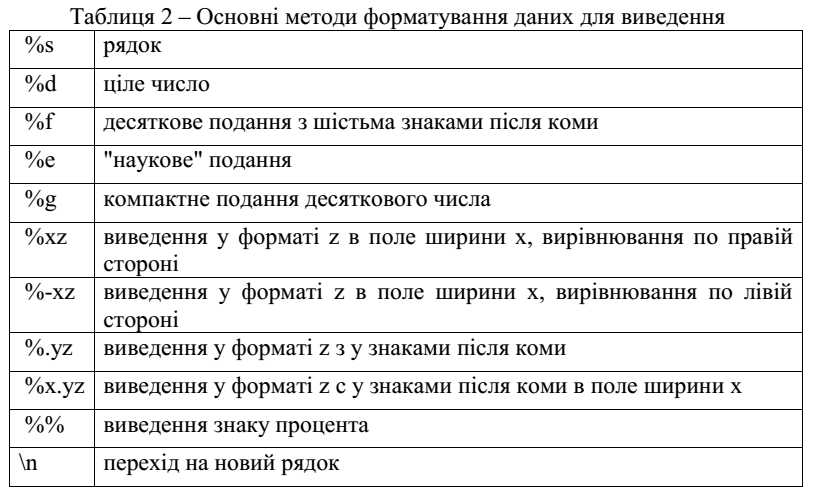
**#Вбудовані математичні функції (необхідне підключення бібліотеки math)**  
from math import \*  
5  
a = 1  
b = 2  
x = sqrt(a\*b)/(exp(a)\*b)+a\*exp((2\*a)/b)  
print(x)

****



**#Введення даних**  
x = input('Введіть х\n') #дані, що вводяться мають тип рядка  
y = input('Введіть y\n')  
x = int(x) #здійснюємо перетворення типів

y = int(y)  
print (x+y) #додаємо два числа, що були введені користувачем  
**#Виведення даних за форматом**  
for i in range(10):  
A = i\*18  
print("%02i\t%.1f" % (i, A))



**#Оператор умови if**  
for i in range(-50,51,10):  
if i<0:  
print("%i - холодно" % (i))  
elif i>0:  
print("%i - тепло" % (i))  
else:  
print("%i - нормально" % (i))  
**#Цикл for**  
for i in range(10):  
print(i)  
A = []  
for i in range(10):  
A.append(i\*\*i)  
B = zeros(10, dtype=int)

for i in range(10):  
B[i] = i\*\*i  
print(A)  
print(B)  
**#Цикл while**  
i = 0  
while i<10:  
print(i)  
i = i + 1  
i = 0  
A = []  
while i<10:  
A.append(i\*\*i)  
i = i + 1  
print(A)

У Python замість масивів використовуються списки, словники, кортежі. Багатомірний масив реалізується за допомогою списку списків (вкладеного списку). За необхідності використовувати саме масиви, напр., при обробці великого об’єму даних, слід застосовувати бібліотеку NumPy, яка дозволяє з ними працювати.

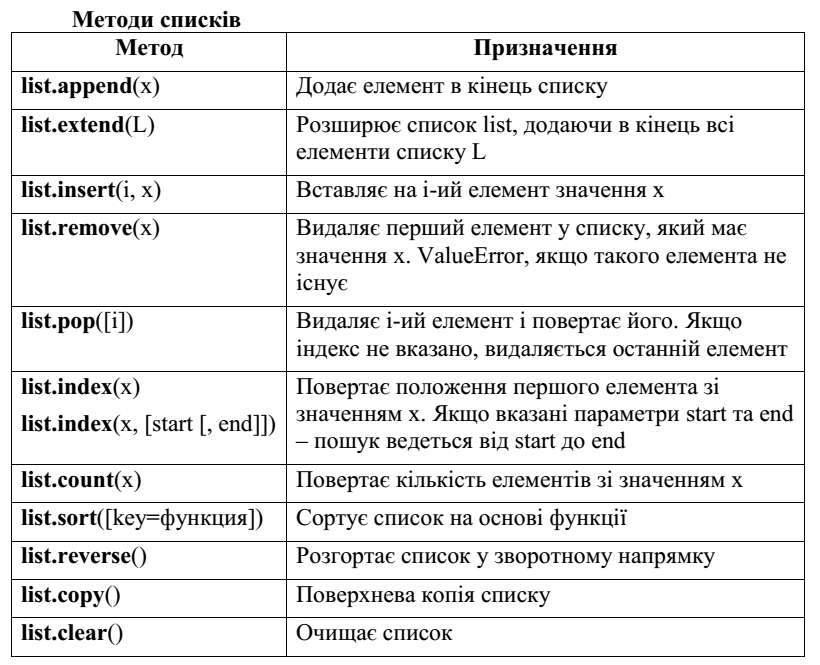
**#Списки. Приклад 1**  
a = [1, 2, 3]  
b = [4, 5]  
c = a + b  
d = b \* 3  
print(a) # буде виведено [1, 2, 3]  
print(b) # буде виведено [4, 5]  
print(d) # буде виведено [4, 5, 4, 5, 4, 5]  
print([7, 8] + [9]) # буде виведено [7, 8, 9]  
print([0, 1] \* 3) # буде виведено [0, 1, 0, 1, 0, 1]  
a2 = [1, 2, 3, 4, 5]  
for i in range(len(a2)):  
print(a2[i])  
for elem in a:  
print(elem, end=' ')

**#Списки. Приклад 2**  
A = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]  
print(A[0:9:2]) # буде виведено [0, 2, 4, 6, 8]  
print(A[0:9:3]) # буде виведено [0, 3, 6]  
print(A[2:4]) # буде виведено [2, 3]  
print(A[-8:-6]) # буде виведено [2, 3]  
print(A[-3:-1]) # буде виведено [7, 8]  
b=list('список')  
print(b) # буде виведено ['с', 'п', 'и', 'с', 'о', 'к']

**#Списки. Приклад 3**  
clouds = ["St", "Sc", "Ns", "As", "Ac"]  
print(type(clouds)) # буде виведено <class 'list'>  
print(clouds) # буде виведено ['St', 'Sc', 'Ns', 'As', 'Ac']  
clouds.append("Cu")  
print(clouds) # буде виведено ['St', 'Sc', 'Ns', 'As', 'Ac', 'Cu']  
clouds.append("Cb")  
print(clouds) # буде виведено ['St', 'Sc', 'Ns', 'As', 'Ac', 'Cu', 'Cb']  
clouds.sort()  
print(clouds) # буде виведено ['Ac', 'As', 'Cb', 'Cu', 'Ns', 'Sc', 'St']  
print(clouds[2]) # буде виведено Cb  
print(clouds[2:5]) # буде виведено ['Cb', 'Cu', 'Ns']  
print(clouds[2:]) # буде виведено ['Cb', 'Cu', 'Ns', 'Sc', 'St']  
print(clouds[:5]) # буде виведено ['Ac', 'As', 'Cb', 'Cu', 'Ns']  
print(clouds[2:5:2]) # буде виведено ['Cb', 'Ns']  
print(clouds[2::2]) # буде виведено ['Cb', 'Ns', 'St']  
print(clouds[::2]) # буде виведено ['Ac', 'Cb', 'Ns', 'St']

**#Списки списків (можна використати, напр., замість двовимірного**  
**масиву)**  
a = [[1, 2, 3], [4, 5, 6]]  
print(a) # буде виведено [[1, 2, 3], [4, 5, 6]]  
print(a[0]) # буде виведено [1, 2, 3]  
print(a[1]) # буде виведено [4, 5, 6]  
print(a[0][0]) # буде виведено 1  
print(a[0][1]) # буде виведено 2  
print(a[1][0]) # буде виведено 4  
  
**#Списки списків. Приклад створення матриці:**  
**#1 0 0 0**  
**#2 1 0 0**  
**#2 2 1 0**  
**#2 2 2 1**  
n = 4  
a = [[0] \* n for i in range(n)]  
for i in range(n):  
for j in range(n):  
if i < j:  
a[i][j] = 0  
elif i > j:  
a[i][j] = 2  
else:  
a[i][j] = 1  
for row in a:  
print(' '.join([str(elem) for elem in row]))

**Зрізи для списків**  
**example\_list[i:j]** # вибере всі елементи списку з i-го (включно) по j-й  
(виключно)  
**example\_list[i:]** # вибере всі елементи списку з i-го (включно) до кінця,  
**example\_list[:j]** # вибере всі елементи списку з початку по j-й (виключно).  
**example\_list[i:j:k]** # вибере кожний k-й елемент списку з i-го (включно)  
по j-й (виключно)  
**example\_list[i::k]** # вибере кожний k-й елемент списку з i-го (включно) до  
кінця  
**example\_list[:j:k]** # вибере кожний k-й елемент списку з початку по j-й  
(виключно),  
**example\_list[::k]** # вибере кожний k-й елемент списку.  
#k також може бути від’ємним, що призведе до формування нового  
списку із зворотним порядком елементів. Перевірте самі: **example\_list[::2]**,  
**example\_list[:4:3]**, **example\_list[1::-1]**, **example\_list[3:0:-1]**.  
**Операції зі списками**  
**x in A** – перевірити, чи міститься елемент в списку. Повертає True або  
False.  
**x not in A** – те ж саме, що not(x in A).  
**min(A)** – найменший елемент списку.  
**max(A)** – найбільший елемент списку



**Завдання**

1. Обчислити значення z та вивести його на екран.
2. Виконати завдання згідно варіанту застосовуючи розгалуження та цикли.
3. Згідно варіанту одномірні масиви слід реалізувати за допомогою списків, а матриці – за допомогою вкладених списків.

**Варіанти завдань:**

**Варіант 1**  
  
Число вводиться користувачем у консолі Python.  
2) Вивести n-ий елемент послідовності Фібоначчі (у послідовності Фібоначчі кожне наступне число дорівнює сумі двох попередніх: 0, 1, 1, 2, 3, 5,8, 13, 21, 34, 55, 89, 144, 233, 377. F0=1, F1=1, F2=1, Fn=Fn-1+Fn-2, n>=2)  
3) Дано матрицю розміром 5х4. Поміняти місцями перший рядок і третій  
рядок.

**Варіант 2**  


Число вводиться користувачем у консолі Python.  
2) Знайти суму всіх чисел від 1 до n, кратних числу k.  
3) Дано одномірний масив, що складається з N дійсних елементів.  
− Знайти максимальний елемент.  
− Обчислити середнє арифметичне від’ємних елементів масиву.  
− Вивести додатні елементи на екран у зворотному порядку.

**Варіант 3**



Числа m та n вводяться користувачем у консолі Python.  
2) Комп'ютер «загадав» число від 1 до 100 (використати функцію random).  
Користувач вводить із клавіатури деяке число й одержує одну з відповідей:  
“Моє число більше”, “Моє число менше”, “Ви вгадали”. Гра повторюється доти  
поки число не вгадане.  
3) Дано одномірний масив, що складається з N цілочисельних елементів.  
− Знайти мінімальний елемент.  
− Обчислити суму додатніх непарних елементів масиву.  
− Вивести додатні елементи на екран.

**Варіант 4**  


Число m вводиться користувачем у консолі Python.  
2) Знайти добуток всіх непарних чисел від х до у  
3) Знайти середнє арифметичне елементів кожного рядка матриці Q(l,m) і  
відняти його з елементів цього рядка.

**Варіант 5**



Числа х та у вводяться користувачем у консолі Python.  
2) Знайти найбільший спільний дільник чисел x та y  
3) Дано двовимірний масив розмірністю 4х6, заповнений цілими числами.  
Сформувати одномірний масив, кожний елемент якого дорівнює найбільшому елементу відповідного рядка.

**Варіант 6**



2) Дано натуральне число n, обчислити у = 1⋅3⋅5…(2⋅n-1).  
3) Дано одномірний масив, що складається з N цілочисельних елементів.  
− Знайти максимальний додатний елемент.  
− Обчислити суму додантіх парних елементів масиву.  
− Вивести від’ємні елементи на екран у зворотному порядку.  
  
**Варіант 7**



Число m вводиться користувачем у консолі Python.  
2) Почавши тренування, спортсмен у перший день пробіг 10 км. Щодня  
він збільшував денну норму на 10% норми попереднього дня. Який сумарний  
шлях пробіжить спортсмен за n днів?  
3) Дано одномірний масив, що складається з N дійсних елементів.  
− Знайти мінімальний додатний елемент.  
− Обчислити середнє арифметичне додатніх елементів масиву.  
− Вивести ненульові елементи на екран у зворотному порядку.

**Варіант 8**



Числа х та у вводяться користувачем у консолі Python.  
2) Обчислити факторіал, використовуючи цикл. Число n вводить  
користувач.  
3) Дано двовимірний масив розмірністю 4х6, заповнений цілими числами.  
Сформувати одномірний масив, кожний елемент якого дорівнює сумі елементів відповідного рядка.

**Варіант 9**



2) Обчислити , використавши цикл, числа x та n вводяться  
користувачем.  
3) Визначити, чи є в двовимірному масиві стовпець, що складається  
тільки з додатніх або нульових елементів.

**Варіант 10**



Число вводиться користувачем у консолі Python.  
2) Щомісячна стипендія студента становить A грн., а витрати на проживання перевищують стипендію й становлять В грн. на місяць. Ріст цін щомісяця збільшує витрати на 5%. Складіть програму розрахунку суми грошей,яку необхідно одноразово попросити в батьків, щоб можна було прожити навчальний рік (10 місяців), використовуючи тільки ці гроші й стипендію.  
3) Дано одномірний масив, що складається з N дійсних елементів.  
− Знайти мінімальний додатний елемент.  
− Обчислити добуток непарних елементів масиву.  
− Вивести від’ємні елементи на екран.

**Варіант 11**



Число m вводиться користувачем у консолі Python.  
2) Визначити,чи являється n простим числом.  
3) Дано одномірний масив, що складається з N дійсних елементів.  
– Знайти мінімальний додатний елемент.  
– Обчислити суму парних елементів масиву.  
– Вивести масив на екран у зворотному порядку.

**Варіант 12**



Число вводиться користувачем у консолі Python.  
2) Одноклітинна амеба ділиться кожні 3 години на 2 клітини. Визначити  
скільки буде амеб через n годин.  
3) Дано одномірний масив, що складається з N дійсних елементів.  
− Знайти максимальний від’ємний елемент.  
− Обчислити середнє арифметичне непарних елементів масиву.  
− Вивести від’ємні елементи на екран.

**Варіант 13**



Числа та вводиться користувачем у консолі Python.  
2) Визначити, чи являється число n надлишковим числом. Надлишкове  
число – додатне ціле число n, сума додатних дільників (відмінних від n) якого перевищує n. Число 48, наприклад, є надлишковим, оскільки  
1+2+3+4+6+8+12+16+24=76, 76 > 48.  
3) Дано одномірний масив, що складається з N дійсних елементів.  
− Знайти мінімальний від’ємний елемент.  
− Обчислити середнє арифметичне додатних елементів масиву.  
− Вивести додатні елементи на екран.

**Варіант 14**



Числа та вводиться користувачем у консолі Python.  
2) Спортсмен пробігає за 1 -й день М км, кожний наступний день він  
збільшує норму пробігу на К%. Визначите через скільки днів норма пробігу  
може стати більше 50 км.  
3) Дано одномірний масив, що складається з N цілочисельних елементів.  
− Знайти максимальний від’ємний елемент.  
− Обчислити добуток від’ємних елементів масиву.  
− Вивести ненульові елементи на екран у зворотному порядку.

**Варіант 15**



Числа та вводиться користувачем у консолі Python.  
2) Знайти найменше спільне кратне чисел x та y.  
3) Дано одномірний масив, що складається з N цілочисельних елементів.  
− Знайти мінімальний елемент.  
− Обчислити добуток ненульових непарних елементів масиву.  
− Вивести масив на екран у зворотному порядку.

**Варіант 16**



Число m вводиться користувачем у консолі Python.  
2) Дано натуральне число n, обчислити у = 2⋅4⋅6…(2⋅n).  
3) Дано одномірний масив, що складається з N цілочисельних елементів.  
− Знайти максимальний від’ємний елемент.  
− Обчислити середнє арифметичне парних елементів масиву.  
− Вивести ненульові елементи на екран у зворотному порядку.

**Варіант 17**



Числа a та b вводяться користувачем у консолі Python.  
2) Дано натуральне число n, обчислити .  
3) Дано одномірний масив, що складається з N дійсних елементів.  
− Знайти максимальний елемент.  
− Обчислити суму парних елементів масиву.  
− Вивести від’ємні елементи на екран у зворотному порядку.

**Варіант 18**



Числа та вводяться користувачем у консолі Python.  
2) Знайти суму S квадратів чисел від 1 до N  
3) Дано одномірний масив, що складається з N цілочисельних елементів.  
− Знайти максимальний додатний елемент.  
− Обчислити добуток елементів масиву.  
− Вивести додатні елементи на екран.

**Варіант 19**



Число x вводиться користувачем у консолі Python.  
2) Знайти суму всіх парних чисел від x до y.  
3) Дано одномірний масив, що складається з N цілочисельних елементів.  
− Знайти мінімальний додатний елемент.  
− Обчислити суму додатних елементів масиву, кратних 3.  
− Вивести не нульові елементи на екран.

**Варіант 20**



Числа m та n вводяться користувачем у консолі Python.  
2) Визначити, чи являється число n досконалим. Досконале число –  
натуральне число, яке дорівнює сумі всіх своїх дільників, напр., 6 (1 + 2 + 3 =  
6), 28 (1 + 2 + 4 + 7 + 14 = 28).  
3) Дано одномірний масив, що складається з N дійсних елементів.  
− Знайти мінімальний елемент.  
− Обчислити добуток не нульових елементів масиву.  
− Вивести додатні елементи на екран у зворотному порядку.

**Варіант 21**



Число х вводиться користувачем у консолі Python.  
2) Знайти перше число Фібоначчі, що буде більше заданого числа р (у  
послідовності Фібоначчі кожне наступне число дорівнює сумі двох попередніх:  
0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377. F0=1, F1=1, F2=1, Fn=Fn-1+Fn-2, n>=2).  
3) Дано одномірний масив, що складається з N дійсних елементів.  
− Знайти максимальний елемент.  
− Обчислити середнє арифметичне від’ємних елементів масиву.  
− Вивести масив на екран у зворотному порядку.

**Варіант 22**



Число х вводиться користувачем у консолі Python.  
2) Визначити, чи являється число n недостатнім числом. Недостатнє  
число – натуральне число, сума власних дільників якого менша за саме число.  
Напр., 15 – недостатнє число, його дільниками є 1, 3 та 5, їх сума рівна 9, що  
менше 15.  
3) Дано одномірний масив, що складається з N цілочисельних елементів.  
− Знайти максимальний додатний елемент.  
− Обчислити суму елементів масиву.  
− Вивести ненульові елементи на екран у зворотному порядку.

**Варіант 23**



Число х вводиться користувачем у консолі Python.  
2) Помiняти порядок цифр числа n на зворотнiй.  
3) Дано одномірний масив, що складається з N цілочисельних елементів.  
− Знайти максимальний від’ємний елемент.  
− Обчислити суму від’ємних елементів масиву.  
− Вивести додатні елементи на екран.

**Варіант 24**



Число х вводиться користувачем у консолі Python.  
2) Дано натуральне число N. Визначити найбільшу цифру і її позицію в  
числі (напр., N=573863, найбільшою є цифра 8, її позиція - четверта зліва).  
Для виконання завдання використати операції цілочисельного ділення та  
знаходження залишку від ділення.  
3) Дано одномірний масив, що складається з N цілочисельних елементів.  
− Знайти максимальний елемент.  
− Обчислити середнє арифметичне непарних елементів масиву.  
− Вивести від’ємні елементи на екран.

**Варіант 25**



Число х вводиться користувачем у консолі Python.  
2) Дано ціле число М. Потрібно знайти найменше ціле від’ємне число k,  
при якому 3k> M.  
3) Дано одномірний масив, що складається з N дійсних елементів.  
− Знайти мінімальний від’ємний елемент.  
− Обчислити добуток ненульових елементів масиву, кратних 3.  
− Вивести від’ємні елементи на екран у зворотному порядку.

**Варіант 26**



Число х вводиться користувачем у консолі Python.  
2) Знайти суму всіх чисел від х до у, кратних числу 3.  
3) Дано одномірний масив, що складається з N дійсних елементів.  
− Знайти мінімальний додатний елемент.  
− Обчислити добуток не нульових елементів масиву.  
− Вивести ненульові елементи на екран у зворотному порядку.

***Контрольні питання:***  
1. Які особливості та переваги мови Python Ви знаєте?  
2. Назвіть основні принципи синтаксису мови Python.  
3. Як здійснюється введення/виведення даних у мові Python?  
4. Який синтаксис циклу for у мові Python?  
5. Назвіть принципи роботи зі списками у мові Python.