**Факультет біомедичної інженерії**

Національного технічного університету України «Київський політехнічний інститут ім. І.Сікорського»

Кафедра біомедичної інженерії

**Дисципліна «Основи програмування»**

*Комп’ютерний практикум №6*

**Тема:** Списки. Інструкція for

Виконав:

студент групи БC-03

Затуловський Георгій Андрійович

Перевірила:

ст. викладач Вдовиченко О.В.

дата xx.xx.2020 р. підпис\_\_\_\_\_\_\_\_\_\_\_

Київ – 2020

**Мета роботи:** Особливості організації циклічної інструкції for. Робота зі

списками. Методи списків.

**Завдання до роботи:** Створити два окремих файли для вирішення завдання та обчислення

значення виразу (див. розд. «Варіанти завдань») при заданих параметрах.

В першому завданні знайти суму n членів ряду, заданого за варіантом.

(Розрахувати число n = варіант +10).

Побудувати блок-схему алгоритму обчислення значень за даними варіантів

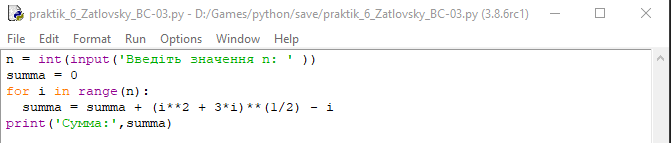
завдань у середовищі Microsoft Visio.

**Варіант № 7**

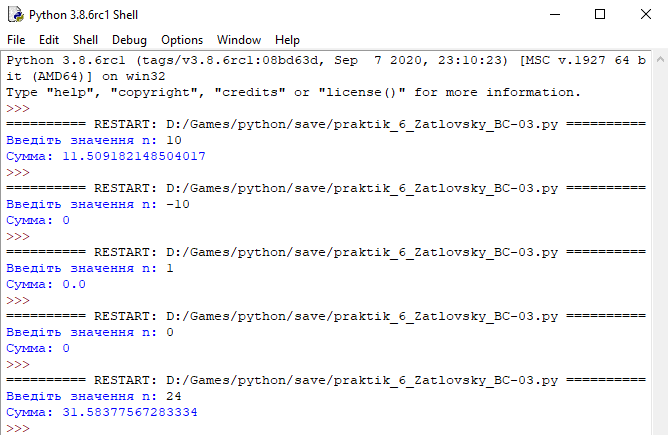
**Завдання 1.**

****

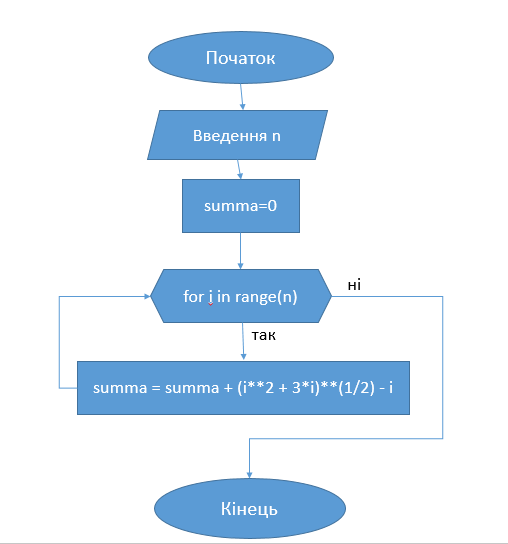
**Лістинг програми:**



**Результати:**

****

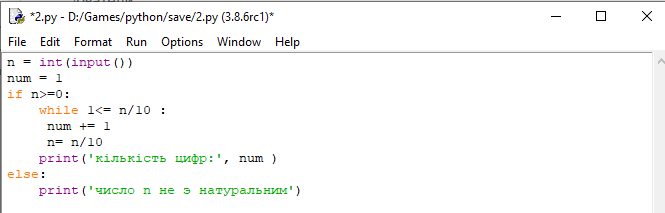
**Блок-схема:**



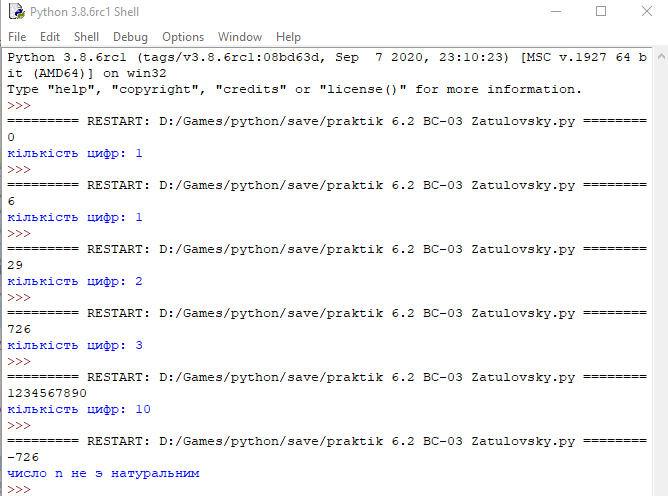
**Завдання 2.**

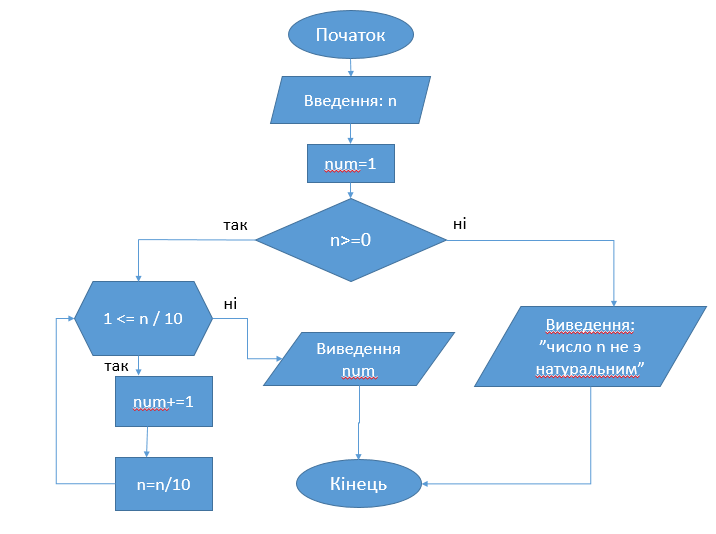


**Лістинг програми:**

****

**Результати:**

****

**Блок-схема:**

**Відповіді на контрольні запитання:**

1. **Як працює оператор for?**

Оператор **for** являє собою цикл обходу заданої множини елементів (символів рядка, об'єктів списку або словника) та виконання в своєму тілі різних операцій над ними.

**for** змінна **in** послідовність**:**

Тіло\_циклу

Наприклад, якщо є список чисел, і необхідно збільшити значення кожного елемента на дві одиниці, то можна перебрати список за допомогою циклу **for**, виконавши над кожним його елементом відповідну дію.

>>> spysok = [0,10,20,30,40,50,60,70,80,90]

>>> i = 0

>>> for element in spysok:

spysok[i] = element+2

i = i + 1

>>> spysok[2, 12, 22, 32, 42, 52, 62, 72, 82, 92]

У прикладі змінна **i** потрібна для того, щоб записати змінене значення елемента в список. У ній зберігається значення індексу чергового елемента списку. У той час, як змінна**element** пов'язується зі значенням чергового елемента даних. У заголовку циклу **for** відбувається звернення до чергового елементу списку. У тілі циклу елементу з індексом **i** присвоюється сума значення поточного (оброблюваного) елемента і двійки. Далі індекс збільшується на одиницю, а потік виконання програми переходить знову в заголовок циклу **for**, де відбувається звернення до іншого елементу списку. Коли всі елементи оброблені цикл **for** припиняє свою роботу. Відсутність чергового елемента є умовою завершення роботи циклу **for**(для порівняння: в циклі **while** умовою завершення служить результат **False** логічного виразу в заголовку).

1. **Для організації яких циклів застосовується оператор for?**

Цикл for в мові програмування призначений для перебору елементів деякої послідовності.

Цикл **for** використовується для роботи зі списками:

Приклад:

**a = [10, 5, 17, 2, 17, 32, 1]**

**for elem in a:**

**if elem%2==0:**

**print(elem, end=' ')**

Результат : 10 2 32

**3. Які методи списків ви знаєте?**

1. **append(item**): додає елемент item до кінця списку
2. **insert(index, item):** додає елемент item до списку за індексом index

3. **remove(item):** видаляє елемент item. Видаляється тільки перше входження елемента. Якщо елемент не знайдено, генерує виключення ValueError

4. **clear():** видалення всіх елементів зі списку

5. **index(item):** повертає індекс елемента item. Якщо елемент не знайдено, генерує виключення ValueError

6. **pop([index]):** видаляє та повертає елемент за індексом index. Якщо індекс не переданий, то видаляє останній елемент

7. **count(item):** повертає кількість входжень елементів ite

m до списку

8. **sort([key]):** сортує елементи. За замовчуванням сортує по зростанню. Але за допомогою параметра key ми можемо передати функцію сортування

9. **reverse():** переставляє всі елементи в зворотньому порядку

**4. Які функції списків ви знаєте?**

1. **len(list):** повертає довжину списку

2. **sorted(list, [key]):** повертає відсортований список

3. **reversed(list):** повертає обернений список

4. **min(list):** повертає найменший елемент списку

5. **max(list):** повертає найбільший елемент списку