

Лабораторна робота № 1

з дисципліни «Об'єктно-орієнтоване програмування СУ»

Тема: «Розробка програм для математичних обчислень в Python»

Виконав студент гр. 312ст_____

Твердохліб Максим

(П.И.Б.)

Перевірів

к.т.н.,доц. О. В.Гавриленко

ас. В. О. Білозерський

(П.И.Б.)

МЕТА РОБОТИ

Ознайомитися з основами розробки інтерпретованих програм і реалізувати скрипт для введення / виведення даних і виконання найпростіших математичних обчислень на мові програмування Python.

ПОСТАНОВКА ЗАДАЧІ

Завдання 1. Вирішити завдання з цілочисельними змінними. Всі вхідні і вихідні дані в задачах цієї групи є цілими числами. Всі числа, для яких вказано

кількість цифр (двозначне число, тризначне число та ін.), вважаються додатними. Завдання представлено в табл.1.

Завдання 2. Обчислити математичне вираз зі змінними дійсного типу, використовуючи стандартну бібліотеку math. Вирази представлено в табл.2.

Завдання 3. Вирішити завдання з логічними змінними. У всіх завданнях даної групи потрібно вивести логічне значення True, якщо наведене висловлювання для запропонованих вихідних даних є істинним, і значення

False в іншому випадку. Всі числа, для яких вказано кількість цифр (двозначне число, тризначне число і т.ін.), вважаються цілими додатними. Завдання представлено в табл.3.

ВИКОНАННЯ РОБОТИ

Завдання 1. Вирішення задачі Integer 8

Вхідні дані: Дано двозначне число. Вивести число, отримане при перестановці цифр вихідного числа.

`number` — двозначне число, введене користувачем. Тип даних: `int`.

Обмеження: повинно бути в діапазоні від 10 до 99 включно.

Вихідні дані:

`swapped_number` — число, отримане після перестановки цифр вихідного числа. Тип даних: `int`.

`error_message` — текстове повідомлення про помилку, якщо введене число не є двозначним. Тип даних: `str`.

Алгоритм вирішення:

Введення даних:

Користувач вводить число, яке має бути двозначним. Це число зберігається в змінній `number`.

Перевірка коректності введення:

Програма перевіряє, чи входить введене число в діапазон від 10 до 99.

Якщо число не відповідає цим умовам, програма виводить повідомлення про помилку і завершує виконання.

Розбиття числа на цифри:

Щоб отримати десятки, програма ділить число націло на 10:

`tens=number//10`
`tens = number // 10`
`tens=number//10.`

Щоб отримати одиниці, програма обчислює залишок від ділення на 10:

`ones=number%10`
`ones = number % 10`
`ones=number%10.`

Перестановка цифр:

Для перестановки цифр місцями, програма формує нове число:

$$\text{swapped_number} = \text{ones} \times 10 + \text{tens}$$
$$\text{swapped_number} = \text{ones} \times 10 + \text{tens}$$

Виведення результату:

Якщо число було введене коректно, програма виводить нове число з переставленими цифрами.

Якщо введене число було некоректним, виводиться повідомлення про помилку.

Лістинг коду вирішення задачі наведено в дод.А стор. 10. Екран роботи програми показаний на рис. Б.1.

ВИСНОВКИ

Було вивчено процес роботи з двозначними числами та їх обробку в Python. На практиці закріплено навички розбиття числа на цифри, перевірки вхідних даних і перестановки цифр місцями. У коді програми відпрацьовано обробку можливих помилок та реалізацію алгоритму для виведення результату.

Завдання 2 Вирішення задачі math 12

Вхідні дані:

x — змінна, введена користувачем. Тип даних: float. Обмеження: $x > 2.5$. Використовується для обчислення математичного виразу.

numerator — чисельник виразу, що включає тангенс абсолютного значення виразу $2x^2 + 5x - 31.152x^2 + 5x - 31.152x^2 + 5x - 31.15$ і логарифм за основою 5 від модуля $x - 2.5x - 2.5x - 2.5$. Тип даних: float.

denominator — знаменник виразу, який є кубічним коренем суми квадрата синуса $x^3x^3x^3$ та частини виразу, що включає логарифм за основою 5 від $x^2 - 2.5x^2 - 2.5x^2 - 2.5$. Тип даних: float.

Вихідні дані:

y — результат обчислення математичного виразу. Тип даних: float.

error_message — текстове повідомлення про помилку, якщо введене число не відповідає умовам або виникає помилка в процесі обчислення. Тип даних: str.

Алгоритм вирішення:

Введення значення змінної:

Користувач вводить значення змінної xxx , яке має бути більше 2.5.

Програма перевіряє коректність введення (чи є введене значення числом і чи відповідає воно обмеженням). Якщо введене значення не відповідає

умовам, виводиться повідомлення про помилку, і програма припиняє виконання.

Обчислення чисельника:

Обчислюється чисельник виразу: тангенс абсолютного значення виразу $2x^2 + 5x - 31.152x^2 + 5x - 31.152x^2 + 5x - 31.15$ плюс логарифм за основою 5 від модуля $x - 2.5x - 2.5x - 2.5$.

Обчислення знаменника:

Обчислюється знаменник виразу: кубічний корінь з суми квадрата синуса $x^3 x^3$ та частини виразу, що містить логарифм за основою 5 від $x^2 - 2.5x^2 - 2.5x^2 - 2.5$.

Перевірка на коректність обчислень:

Програма перевіряє, чи не виникають помилки під час виконання математичних операцій (наприклад, якщо знаменник дорівнює нулю або виникають інші помилки при логарифмуванні чи тригонометричних функціях).

Обчислення результату:

Якщо всі обчислення пройшли успішно, програма обчислює значення ууу, яке дорівнює відношенню чисельника до знаменника.

Виведення результату:

Якщо обчислення були виконані коректно, програма виводить значення ууу.

Якщо виникає помилка на будь-якому етапі, програма виводить відповідне повідомлення про помилку і завершує роботу.

Лістинг коду вирішення задачі наведено в дод. А.стор 10. Екран роботи програми показаний на рис. Б.2.

ВИСНОВКИ

У результаті виконаної роботи було вивчено і застосовано на практиці обчислення складних математичних виразів за допомогою модуля ``math`` у Python. Було закріплено навички роботи з логарифмами, тригонометричними функціями, обробкою винятків, а також проведено перевірку на коректність введених користувачем даних. Реалізовано алгоритм для обчислення заданого виразу, що дозволило отримати правильні результати або відповідні повідомлення про помилки.

Завдання 3 Вирішення задачі Boolean 19

Вхідні дані:

Надаються три цілі числа, для яких необхідно перевірити наявність хоча б однієї пари взаємно протилежних чисел.

Тип даних:

a — ціле число, може бути як додатним, так і від'ємним.

b — ціле число, може бути як додатним, так і від'ємним.

c — ціле число, може бути як додатним, так і від'ємним.

Обмеження:

Цілі числа a , b , c належать множині цілих чисел \mathbb{Z} , тобто можуть приймати як додатні, так і від'ємні значення або бути нулями.

Числа можуть бути різними або однаковими.

Вихідні дані:

Опис: Необхідно вивести результат у вигляді логічного значення:

True — якщо серед трьох наданих чисел є хоча б одна пара взаємно протилежних чисел (тобто таких чисел, що одне з них є протилежним значенню іншого).

False — якщо серед трьох чисел немає жодної пари взаємно протилежних чисел.

Тип даних: Логічне значення (True або False).

Алгоритм вирішення:

Введення трьох цілих чисел:

Програма приймає три числа a , b , c як вхідні параметри. Вхідні значення можуть бути додатними, від'ємними а бо рівними нулю.

Перевірка взаємно протилежних чисел:

Крок 1: Перевіряємо першу пару a та b : якщо $a = -b$ або $b = -a$, то це взаємно протилежні числа. Якщо це так, то результатом буде True.

Крок 2: Якщо перша пара не є протилежною, перевіряємо наступну пару a та c : якщо $a = -c$ або $c = -a$, то це взаємно протилежні числа. Якщо це так, то результатом буде True.

Крок 3: Якщо жодна з попередніх пар не є протилежною, перевіряємо третю пару b та c : якщо $b = -c$ або $c = -b$, то це взаємно протилежні числа. Якщо це так, то результатом буде True.

Виведення результату:

Якщо хоча б одна з перевірок була успішною (умови виконалися), виводимо True.

Якщо жодна з умов не виконалася, виводимо False.

Лістинг коду вирішення наведено в дод. А. стор. 10. Екран роботи програми показаний на рис. Б.3

ВИСНОВОК

Було вивчено та закріплено на практиці алгоритм перевірки взаємно протилежних чисел серед трьох цілих значень. Алгоритм було відпрацьовано в коді програми, що дозволило отримати навички обробки умовних операторів і роботи з логічними значеннями. Результати роботи демонструють ефективність перевірки чисел на взаємну протилежність.

ДОДАТОК А

Лістинг коду програми для задачі Завдання 1

```
def swap_digits():

    try:

        # Введення двозначного числа

        number = int(input("Введіть двозначне число: "))

        # Перевірка, чи є введене число двозначним

        if number < 10 or number > 99:

            raise ValueError("Число повинно бути двозначним.")

        # Отримання десятків та одиниць

        tens = number // 10

        ones = number % 10

        # Перестановка цифр

        swapped_number = ones * 10 + tens

        # Виведення результату

        print("Число після перестановки цифр:", swapped_number)

    except ValueError as e:

        print(e)

# Виклик функції

swap_digits()
```

Лістинг коду для Завдання 2

```
import math

def task2_0(): """
```

Обчислення математичного виразу:

$$y = \operatorname{tg}(|2 * x^2 + 5 * x - 31.15| + \log_5|x - 2.5|) /$$

$$(\text{корінь кубічний з } (\sin^2(x^3) + 1/3 * \log_5|x^2 - 2.5|))$$

```
"""
```

```
try:
```

```
# Введення значення змінної
```

```
    x = float(input("Введіть значення x (x > 2.5): "))
```

```
    if x <= 2.5:
```

```
        raise ValueError("x має бути більше 2.5")
```

```
except ValueError as e:
```

```
    print("Помилка:", e)
```

```
    input("Натисніть Enter для виходу...")
```

```
else:
```

```
    try:
```

```
        # Чисельник виразу
```

```
        numerator = math.tan(abs(2 * x**2 + 5 * x - 31.15) +
math.log(abs(x - 2.5), 5))
```

```
        # Знаменник виразу
```

```
        denominator = (math.sin(x**3)**2 + (1/3) * math.log(abs(x**2 -
2.5), 5)) ** (1/3)
```

```
        # Обчислення y
```

```
        y = numerator / denominator
```

```
except Exception as e:
```

```
    print("Помилка в обчисленнях:", e)
```

```

        input("Натисніть Enter для виходу...")

    else:

        print(f"Результат: y = {y}")

# Виклик функції

task2_0()

```

Лістинг коду програми до Завдання 3

```

def check_opposite_pair(a, b, c):

    """

    Перевіряє, чи є серед трьох цілих чисел хоча б одна пара взаємно
    протилежних.

    Параметри:

    a (int): Перше ціле число.

    b (int): Друге ціле число.

    c (int): Третє ціле число.

    Повертає:

    bool: True, якщо є хоча б одна пара взаємно протилежних чисел, інакше
    False.

    """

    if a == -b or a == -c or b == -c:

        return True

    else:

        return False

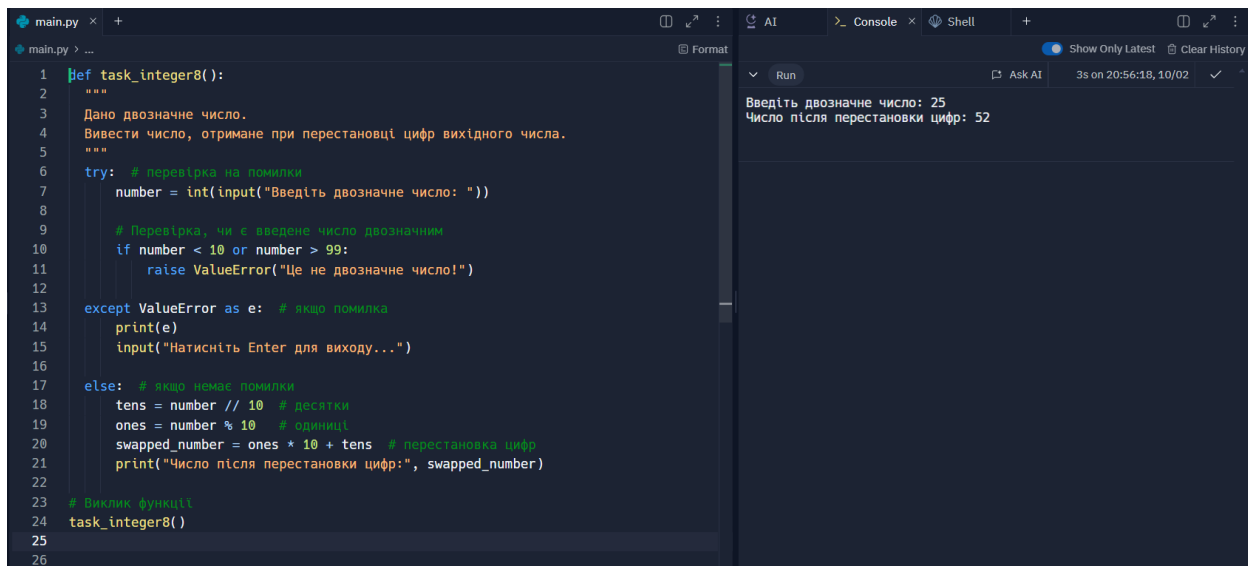
# Приклад використання

```

```
a = int(input("Введіть перше число: "))  
  
b = int(input("Введіть друге число: "))  
  
c = int(input("Введіть третє число: "))  
  
result = check_opposite_pair(a, b, c)  
  
print(f"Чи є хоча б одна пара взаємно протилежних чисел? {result}")
```

ДОДАТОК Б

Скрін-шоти вікна роботи програми



```
1 def task_integer8():
2     """
3     Дано двозначне число.
4     Вивести число, отримане при перестановці цифр вихідного числа.
5     """
6     try: # перевірка на помилки
7         number = int(input("Введіть двозначне число: "))
8
9         # Перевірка, чи є введене число двозначним
10        if number < 10 or number > 99:
11            raise ValueError("Це не двозначне число!")
12
13    except ValueError as e: # якщо помилка
14        print(e)
15        input("Натисніть Enter для виходу...")
16
17    else: # якщо немає помилки
18        tens = number // 10 # десятки
19        ones = number % 10 # одиниці
20        swapped_number = ones * 10 + tens # перестановка цифр
21        print("Число після перестановки цифр:", swapped_number)
22
23 # Виклик функції
24 task_integer8()
25
26
```

Введіть двозначне число: 25
Число після перестановки цифр: 52

Рисунок Б.1 – Екран виконання програми для вирішення Завдання 1
(Integer)

```

1  import math
2
3  def task2_0():
4      """
5      Обчислення математичного виразу:
6       $y = \frac{\tan(|2 \cdot x^2 + 5 \cdot x - 31.15| + \log_5|x - 2.5|)}{(\sqrt[3]{\sin^2(x^3) + 1/3 \cdot \log_5|x^2 - 2.5|})}$ 
7      """
8
9      try:
10         # Введення значення змінної
11         x = float(input("Введіть значення x (x > 2.5): "))
12
13         if x <= 2.5:
14             raise ValueError("x має бути більше 2.5")
15
16     except ValueError as e:
17         print("Помилка:", e)
18         input("Натисніть Enter для виходу...")
19
20     else:
21         try:
22             # Чисельник виразу
23             numerator = math.tan(abs(2 * x**2 + 5 * x - 31.15) + math.log(abs(x - 2.5), 5))
24
25             # Знаменник виразу
26             denominator = (math.sin(x**3)**2 + (1/3) * math.log(abs(x**2 - 2.5), 5)) ** (1/3)
27
28             # Обчислення y
29             y = numerator / denominator
30
31         except Exception as e:
32             print("Помилка в обчисленнях:", e)
33             input("Натисніть Enter для виходу...")
34         else:
35             print(f"Результат: y = {y}")
36
37     # Виклик функції
38     task2_0()
39

```

```

AI  Console  Shell
Show Only Latest  Clear History
Run  Ask AI  5s on 20:57:33, 10/02  ✓
Введіть значення x (x > 2.5): 3
Результат: y = 5.999094564422023

```

Рисунок Б.2 – Екран виконання програми для вирішення Завдання 2 (math)

```
1 def check_opposite_pair(a, b, c):
2     """
3     Перевіряє, чи є серед трьох цілих чисел хоча б одна пара взаємно протилежних.
4
5     Параметри:
6     a (int): Перше ціле число.
7     b (int): Друге ціле число.
8     c (int): Третє ціле число.
9
10    Повертає:
11    bool: True, якщо є хоча б одна пара взаємно протилежних чисел, інакше False.
12    """
13    if a == -b or a == -c or b == -c:
14        return True
15    else:
16        return False
17
18    # Приклад використання
19    a = int(input("Введіть перше число: "))
20    b = int(input("Введіть друге число: "))
21    c = int(input("Введіть третє число: "))
22
23    result = check_opposite_pair(a, b, c)
24    print(f"Чи є хоча б одна пара взаємно протилежних чисел? {result}")
25
```

Введіть перше число: 1
Введіть друге число: 4
Введіть третє число: -5
Чи є хоча б одна пара взаємно протилежних чисел? False

Рисунок Б.3 – Екран виконання програми для вирішення Завдання 3 (Boolean)