МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Національний аерокосмічний університет ім. М. Є. Жуковського

«Харківський авіаційний інститут»

Кафедра систем управління літальними апаратами

Лабораторна робота № 1

з дисципліни «Об'єктно-орієнтоване програмування СУ»

Тема: «Розробка програм для математичних обчислень в Python»

ХАІ.301 .173. 310ст.1 ЛР

Виконав студент гр. \_\_\_\_\_\_310ст\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_Возвишаєв Олексій Андрійович

(підпис, дата) (П.І.Б.)

Перевірив

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ к.т.н., доц. О. В. Гавриленко

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ас.  В. О. Білозерський

(підпис, дата) (П.І.Б.)

2024

# 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.), вважаютьс 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 значенняFalse в іншому випадку. Всі числа, для яких вказано кількість цифр (двозначнечисло, тризначне число і т.ін.), вважаються цілими додатними. Завданняпредставлено в табл.3.

# ВИКОНАННЯ РОБОТИ

Завдання 1. Вирішення задачі 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.

Вхідні дані: Дано тризначне число. Знайти суму і добуток його цифр.

num – тризначне число, введене користувачем. Тип: int. Діапазон допустимих значень: від 100 до 999 (включно).

hundreds – кількість сотень у тризначному числі. Тип: int. Діапазон: від 1 до 9.

tens – кількість десятків у тризначному числі. Тип: int. Діапазон: від 0 до 9.

ones – кількість одиниць у тризначному числі. Тип: int. Діапазон: від 0 до 9.

digit\_sum – сума цифр тризначного числа. Тип: int. Діапазон: від 1 до 27 (мінімум 1 + 0 + 0, максимум 9 + 9 + 9).

digit\_product – добуток цифр тризначного числа. Тип: int. Діапазон: від 0 до 729 (мінімум 0 \* будь-яка цифра, максимум 9 \* 9 \* 9).Вихідні дані (ім’я, опис, тип):

num – тризначне число, яке вводить користувач.

Тип: int

Опис: Введене користувачем тризначне число, яке підлягає розбиттю на цифри.

hundreds – кількість сотень у тризначному числі.

Тип: int

Опис: Цифра, що відповідає сотням у тризначному числі. Вираховується діленням числа на 100.

tens – кількість десятків у тризначному числі.

Тип: int

Опис: Цифра, що відповідає десяткам у тризначному числі. Вираховується діленням числа на 10 та залишком від ділення на 10.

ones – кількість одиниць у тризначному числі.

Тип: int

Опис: Цифра, що відповідає одиницям у тризначному числі. Вираховується як залишок від ділення числа на 10.

digit\_sum – сума цифр тризначного числа.

Тип: int

Опис: Сума цифр (сотень, десятків та одиниць) тризначного числа.

digit\_product – добуток цифр тризначного числа.

Тип: int

Опис: Добуток цифр (сотень, десятків та одиниць) тризначного числа.

Відповідні текстові сповіщення:

"Enter a three-digit number:" – запит на введення тризначного числа.

"The number must be three digits!" – повідомлення про помилку, якщо введене число не є тризначним.

"Sum of digits: {digit\_sum}" – повідомлення з результатом суми цифр числа.

"Product of digits: {digit\_product}" – повідомлення з результатом добутку цифр числа.

Алгоритм вирішення показано нижче

Початок.

Введення: Запросити користувача ввести тризначне число.

Перевірка:

Якщо число < 100 або число > 999:

Вивести повідомлення про помилку.

Завершити виконання програми.

Витягування цифр:

Знайти сотні: hundreds = num // 100

Знайти десятки: tens = (num // 10) % 10

Знайти одиниці: ones = num % 10

Обчислення:

Обчислити суму цифр: digit\_sum = hundreds + tens + ones

Обчислити добуток цифр: digit\_product = hundreds \* tens \* ones

Вивід результату:

Вивести суму цифр.

Вивести добуток цифр.

Кінець.

Лістинг коду вирішення задачі наведено в дод. А стор.9. Екран роботи програми показаний на рис. Б.1.

# ВИСНОВКИ

Було вивчено та закріплено на практиці методи роботи з цілочисельними змінними в Python. Під час вирішення задачі відпрацьовано алгоритм розбиття тризначного числа на окремі цифри для подальшого обчислення їх суми та добутку. Отримано практичні навички перевірки вхідних даних і застосування арифметичних операцій для вирішення завдань з цілими числами.

Завдання 2. Вирішення задачі 2 (math) 6

Вхідні дані (ім’я, опис, тип, обмеження):

x — значення аргументу, яке вводиться користувачем для обчислення функції.

Тип: float

Обмеження: 𝑥>7.5 (оскільки логарифм не може бути визначений для значень 𝑥+7.5≤0

Вихідні дані (ім’я, опис, тип):

y — результат обчислення математичного виразу за заданим 𝑥.

Тип: float, якщо обчислення успішні.

error\_message — текстове повідомлення, яке відображається у разі помилки (наприклад, недопустимі значення x, що можуть призвести до математичних помилок).

Тип: str, якщо виникає помилка.

Алгоритм вирішення показано нижче

Введенняданих: Користувач вводить значення змінної x.

Перевірка, чи x>−7.5x . Якщо умова не виконується, видається повідомлення про помилку.

Вирахування чисельника:

Обчислюється 2x

Обчислюється ∣sin(x)∣.

Обчислюється ∣tan(x)∣.

Обчислюється 2.5.

Обчислюється сума і множення компонентів чисельника.

Підноситься до кореня 5-го ступеня.

Вирахування знаменника:

* Обчислюється (x+7.5), додається до 0.625.
* Якщо x+7.5≤0, видається повідомлення про помилку.

Обчислення виразу:

* Ділимо чисельник на знаменник.

Виведення результату:

* Виводиться результат y, або повідомлення про помилку, якщо є недопустимі значення.

Лістинг коду вирішення задачі наведено в дод. А стор.9. Екран роботи програми показаний на рис. Б.2.

# ВИСНОВКИ

Було вивчено та закріплено на практиці методи роботи з математичними функціями в Python. У ході виконання завдання відпрацьовано обчислення складних виразів, таких як корені, логарифми та тригонометричні функції, а також реалізовано обробку виняткових ситуацій у коді програми. Отримано навички перевірки коректності введених даних та обчислення значень для різних змінних.

Завдання 3. Вирішення задачі (Boolean) 13

Вхідні дані:

A — перше ціле число, введене користувачем.

Тип: int

Обмеження: будь-яке ціле число.

B — друге ціле число, введене користувачем.

Тип: int

Обмеження: будь-яке ціле число.

C — третє ціле число, введене користувачем.

Тип: int

Обмеження: будь-яке ціле число.

Вихідні дані:

res — результат перевірки істинності висловлювання «Хоча б одне з чисел A, B, C позитивне».

Тип: bool (значення True або False).

error\_message — текстове повідомлення про помилку у випадку неправильного введення (наприклад, якщо введено не ціле число).

Тип: str, у випадку виникнення помилки.

Алгоритм вирішення показано нижче

Введення даних: Користувач вводить три цілі числа A,B,C

У разі некоректного введення (не ціле число) виводиться повідомлення про помилку.

Перевірка істинності висловлювання: Виконується логічна операція: перевіряється, чи хоча б одне з чисел A,B,C є позитивним. Якщо умова виконується (хоча б одне число більше за 0), результатом є True, інакше — False.

Виведення результату: Виводиться результат перевірки істинності висловлювання.

Лістинг коду вирішення задачі наведено в дод. А стор.9. Екран роботи програми показаний на рис. Б.3.

# ВИСНОВКИ

Було вивчено методи перевірки умов за допомогою логічних операторів. На практиці закріплено навички введення та обробки цілих чисел, а також перевірки їх на позитивність. В коді програми відпрацьовано логіку перевірки істинності висловлювання та обробку помилок при введенні некоректних даних.

ДОДАТОК А

Лістинг коду програми до задач Завдання 1

def task\_integer11():

"""Given a three-digit number, find the sum and product of its digits."""

try:

num = int(input("Enter a three-digit number: "))

print(f"You entered: {num}") # Додаємо перевірку на введене число

if num < 100 or num > 999:

raise ValueError("The number must be three digits!")

except ValueError as e:

print(e)

return # Зупиняємо виконання, якщо є помилка

else:

# Витягуємо цифри

hundreds = num // 100

tens = (num // 10) % 10

ones = num % 10

# Сума та добуток цифр

digit\_sum = hundreds + tens + ones

digit\_product = hundreds \* tens \* ones

# Обчислюємо чисельник

print(f"Sum of digits: {digit\_sum}")

print(f"Product of digits: {digit\_product}")

task\_integer11()

Enter a three-digit number: 123

You entered: 123

Sum of digits: 6

Product of digits: 6

Лістинг коду програми до задач Завдання 2

import math

def calculate\_y(x):

try:

# Обчислюємо чисельник

numerator = (2 \* x\*\*2 - abs(math.sin(x)) \* abs(math.tan(x)) \* 2.5\*\*math.cos(x))\*\*(1/5)

# Обчислюємо знаменник

denominator = 0.625 + 2 \* math.log2(x + 7.5)

# Повний вираз

y = numerator / denominator

return y

except ValueError:

return " Помилка: неприпустимі значення для логарифму чи інших математичних операцій."

# Приклад використання

x = float(input("Введіть значення x: "))

result = calculate\_y(x)

print(f"y = {result}")

Лістинг коду програми до задач Завдання 3

def check\_positive():

try:

# Ввод трёх целых чисел A, B, C

A = int(input("Введите число A: "))

B = int(input("Введите число B: "))

C = int(input("Введите число C: "))

except ValueError:

print("Усі значення мають бути цілими числами!")

return

# Перевірка умови: хоча б одне із чисел позитивне

res = A > 0 or B > 0 or C > 0

# Висновок результату

print( res)

# Визов функції перевірки

check\_positive()

ДОДАТОК Б

Скрін-шоти вікна виконання програми

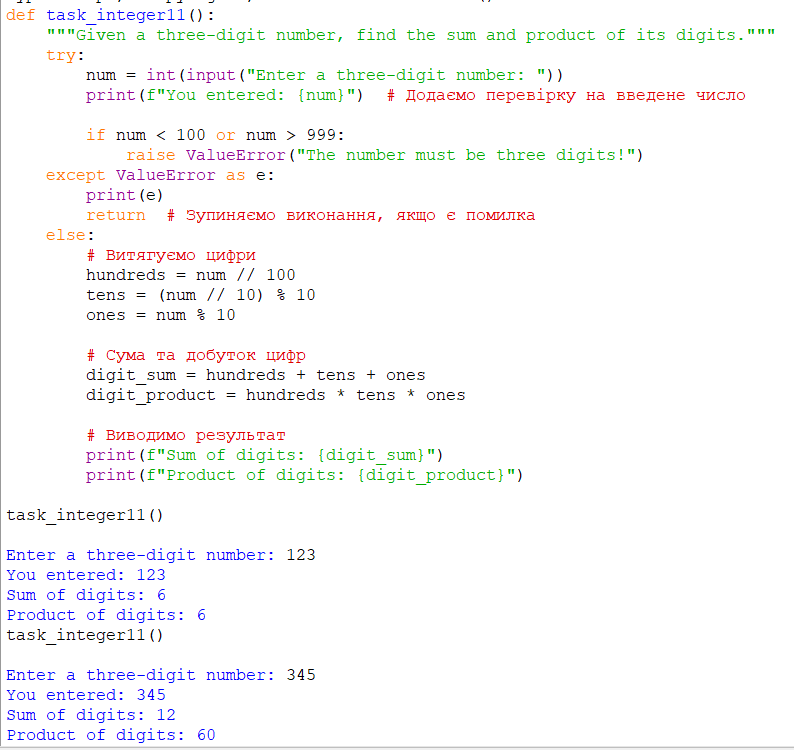


Рисунок Б.1 – Екран виконання програми для вирішення завдання   
Завдання1

(Integer)

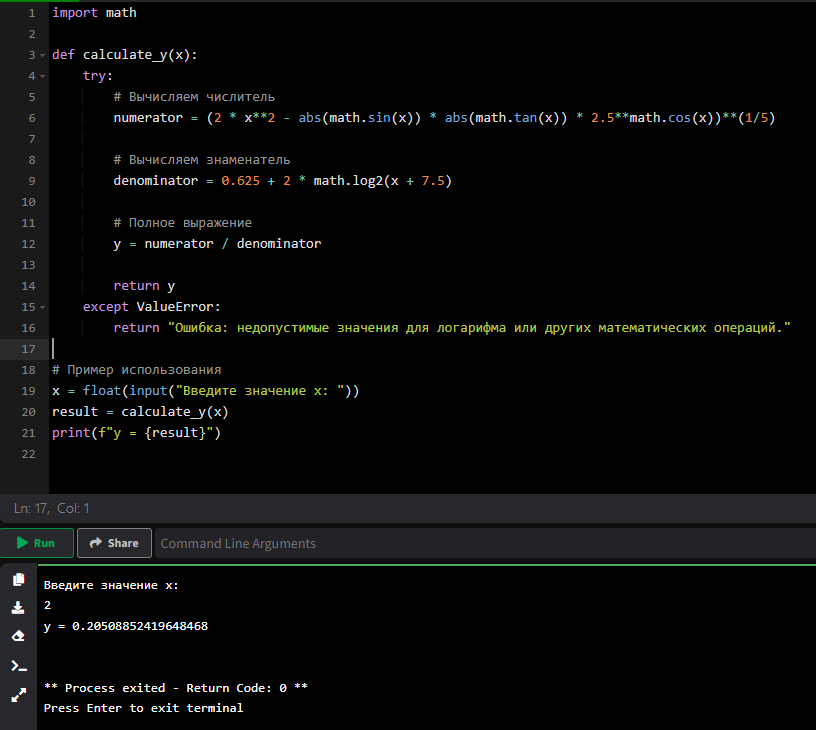


Рисунок Б.2 – Екран виконання програми для вирішення завдання   
Завдання 2  
(math)

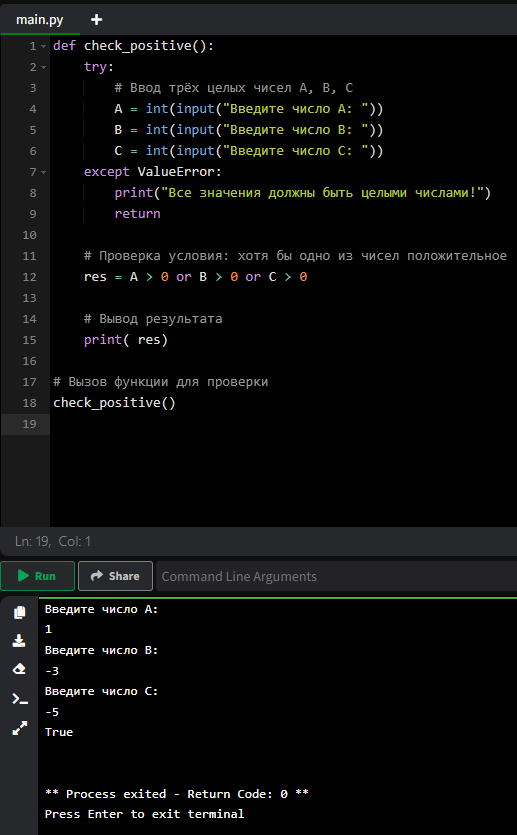


Рисунок Б.3 – Екран виконання програми для вирішення завдання   
Завдання 3

(Boolean)