МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Національний аерокосмічний університет ім. М. Є. Жуковського

«Харківський авіаційний інститут»

Кафедра систем управління літальними апаратами

Лабораторна робота № 2

з дисципліни Об'єктно-орієнтоване програмування СУ

Тема: “"Розробка структурованих програм з розгалуженням та повтореннями"

ХАІ.301 . <спец.>. 322. 1 ЛР

Виконав студент гр. \_\_\_\_\_\_322\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Кулагін Олексій\_\_\_

(підпис, дата) (П.І.Б.)

Перевірив

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

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

(підпис, дата) (П.І.Б.)

2023

# МЕТА РОБОТИ

# Вивчити теоретичний матеріал щодо синтаксису на мові Python і поданням

# у вигляді UML діаграм діяльності алгоритмів з розгалуження та циклами, а

# також навчитися використовувати функції, інструкції умовного переходу і

# циклів для реалізації інженерних обчислень. ПОСТАНОВКА ЗАДАЧІ

Завдання 1. Вирішити завдання на алгоритми з розгалуженням. Завдання

представлено в табл.1.  
Завдання 2. Дано дійсні числа (xi, yi), i = 1,2, ... n, – координати точок на

площині. Визначити кількість точок, що потрапляють в геометричну область

заданого кольору (або групу областей). Варіанти геометричних областей

представлені в табл.2.  
Завдання 3. Дослідити ряд на збіжність. Умова закінчення циклу

обчислення суми прийняти у вигляді: | un | <E або | un | > G де е – мала величина

для переривання циклу обчислення суми сходиться ряду (е = 10-5... 10-20); g –

величина для переривання циклу обчислення суми розходиться ряду (g = 102...105). Варіанти представлено в табл.3.

# ВИКОНАННЯ РОБОТИ

Завдання 1. Вирішення задачі Integer(19)

Вхідні дані:

nums: список із чотирьох цілих чисел, одне з яких відмінне від трьох інших, рівних між собою.

Опис: чотири цілі числа, одне з яких відрізняється від інших.

Тип: список цілих чисел (list of integers).

Обмеження: кожне число повинно бути цілим, три з них повинні бути рівними, а одне відмінне від інших трьох. Усі числа в межах типу int в Python.

Вихідні дані:

· unique\_num\_index: порядковий номер числа, яке відрізняється від трьох інших, рівних між собою.

· Опис: індекс числа у списку, яке відрізняється від інших трьох чисел.

Тип: ціле число (integer).

· ValueError: текстове повідомлення про помилку.

· Опис: повідомлення про помилку у випадку, якщо введені дані не є цілими числами.

Тип: рядок (string).

Завдання 2  
Вхідні данні

r: радіус кола та катети прямокутного трикутника.

Опис: радіус кола, який також є довжиною катетів прямокутного трикутника.

Тип: дійсне число (float).  
Вихідні данні:  
· area\_circle: площа кола з радіусом rrr.

· Опис: площа кола

Тип: дійсне число (float).

· area\_triangle: площа прямокутного трикутника з катетами rrr.

· Опис: площа прямокутного трикутника

Тип: дійсне число (float).

· ValueError: текстове повідомлення про помилку.

Тип: рядок (string).

Завдання 3  
Вхідні данні:  
n\_max: максимальне значення nnn для обчислення суми ряду.

Опис: верхня межа для значення nnn, до якої буде обчислюватися сума.

Тип: ціле число (integer).  
Вихідні дані:

Опис: підсумкова сума ряду для заданого діапазону.

Тип: дійсне число (float).

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

# ВИСНОВКИ Було вивчено теоретичний матеріал щодо синтаксису мови Python та представлення алгоритмів у вигляді UML діаграм діяльності з розгалуженнями та циклами. На практиці закріплено вміння використовувати функції, інструкції умовного переходу та цикли для реалізації інженерних обчислень. Відпрацьовано навички розробки програмного коду для розв'язання задач з використанням умовних операторів та циклічних конструкцій, що дозволило отримати необхідні навички для виконання інженерних обчислень на Python. ДОДАТОК А

Лістинг коду програми до задач if19

# module m.py

def task\_if19():

    try:

        # Вводим четыре целых числа

        nums = [int(input(f"Число {i+1}: ")) for i in range(4)]

        # Определяем порядковый номер отличного числа

        if nums[0] == nums[1] == nums[2]:

            unique\_num\_index = 3

        elif nums[0] == nums[1] == nums[3]:

            unique\_num\_index = 2

        elif nums[0] == nums[2] == nums[3]:

            unique\_num\_index = 1

        else:

            unique\_num\_index = 0

        print("Порядковый номер отличного числа:", unique\_num\_index + 1)

    except ValueError:

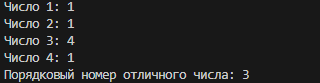
        print("Ожидается целое число!")

# Вызываем функцию для проверки

task\_if19()

ДОДАТОК Б

Скрін-шоти вікна виконання програми Integer19


ДОДАТОК А.2

Лістинг коду програми до задач variant12  
  
  
import math

def task\_variant\_12(r):

    try:

        # Проверяем, что радиус положительный

        if r <= 0:

            raise ValueError("Радиус должен быть положительным числом")

        # Площадь круга

        area\_circle = math.pi \* r\*\*2

        # Площадь прямоугольного треугольника

        area\_triangle = (r \* r) / 2

        print(f"Площадь круга с радиусом {r} равна: {area\_circle:.2f}")

        print(f"Площадь прямоугольного треугольника с катетами {r} равна: {area\_triangle:.2f}")

        return area\_circle, area\_triangle

    except ValueError as ve:

        print(f"Ошибка: {ve}")

        return None

# Пример вызова функции

r = float(input("Введите радиус r: "))

task\_variant\_12(r)

ДОДАТОК Б.2

Скрін-шоти вікна виконання програми task\_expression27


ДОДАТОК А.3

Лістинг коду програми до задач math28  
  
  
import math

def compute\_series(n\_max=100):

    total = 0

    for n in range(1, n\_max + 1):

        term = (math.exp(2 \* n + 1) + n) / math.sqrt(math.factorial(n))

        total += term

    return total

# Пример использования:

n\_max = int(input("Введите максимальное значение n для вычисления суммы: "))

result = compute\_series(n\_max)

print(f"Результат суммы для n от 1 до {n\_max}: {result:.6f}")

ДОДАТОК Б.3

Скрін-шоти вікна виконання програми boolean28