

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.301.174.322.04 ЛР

Виконав студент гр. _____ 322 _____

_____ Зубов Євген Павлович _____
(підпис, дата) (П.І.Б.)

Перевірів

_____ к.т.н., доц. О. В. Гавриленко
_____ ас. В. О. Білозерський
(підпис, дата) (П.І.Б.)

МЕТА РОБОТИ

Вивчено 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.

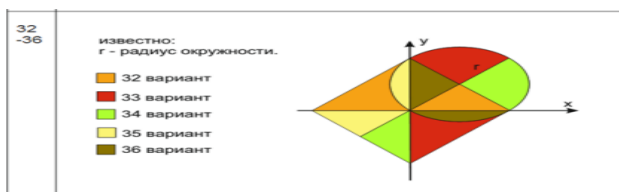
Завдання 2. Дано дійсні числа (x_i, y_i) , $i = 1, 2, \dots, n$, – координати точок на площині. Визначити кількість точок, що потрапляють в геометричну область заданого кольору (або групу областей).

Завдання 3. Дослідити ряд на збіжність. Умова закінчення циклу обчислення суми прийняти у вигляді: $|u_n| < \epsilon$ або $|u_n| > G$ де ϵ – мала величина для переривання циклу обчислення суми сходиться ряду

Завдання 4. Для багаторазового виконання будь-якого з трьох зазначених вище завдань на вибір розробити циклічний алгоритм організації меню в командному вікні.

If16. Дано три змінні дійсного типу: A, B, C. Якщо їх значення впорядковані за зростанням, то подвоїти їх; в іншому випадку замінити значення кожної змінної на протилежне. Вивести нові значення змінних A, B, C.

Graph32



25	$\sum_{n=1}^{\infty} \frac{n!}{7 * 12 * 17 \dots (5n + 2)}$
----	---

Series25

ВИКОНАННЯ РОБОТИ

Завдання 1. Вирішення задачі If16:

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

num1 – число, float

num2 – число, float

num3 – число, float

Вихідні дані (ім'я, опис, тип):

Вивід результату у консолі

Алгоритм вирішення показано нижче чи показано на рис. 1

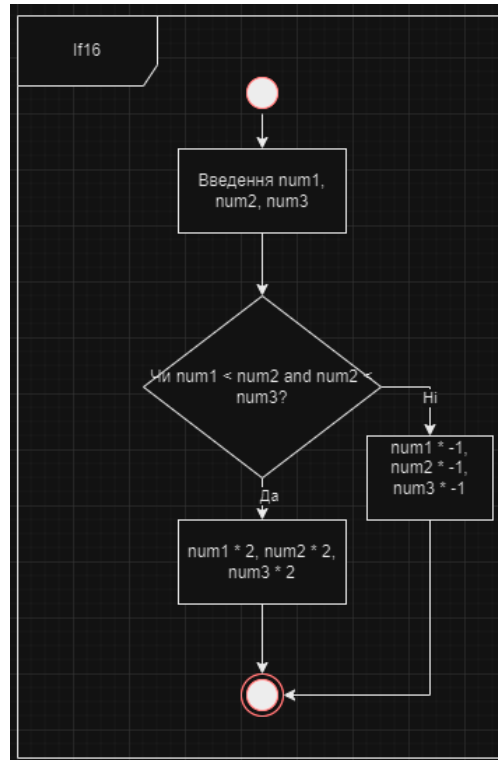


Рисунок 1 – Алгоритм вирішення If16

Лістинг коду вирішення задачі наведено в дод. А (стор. 6).
Екран роботи програми показаний на рис. Б.1.

Завдання 2. Вирішення задачі Graph32:

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

x – змінна, координата x, float

y – змінна, координата y, float

r – змінна, радіус кола, float

Вихідні дані (ім'я, опис, тип):

Вивід інформації на екран в залежності від умови

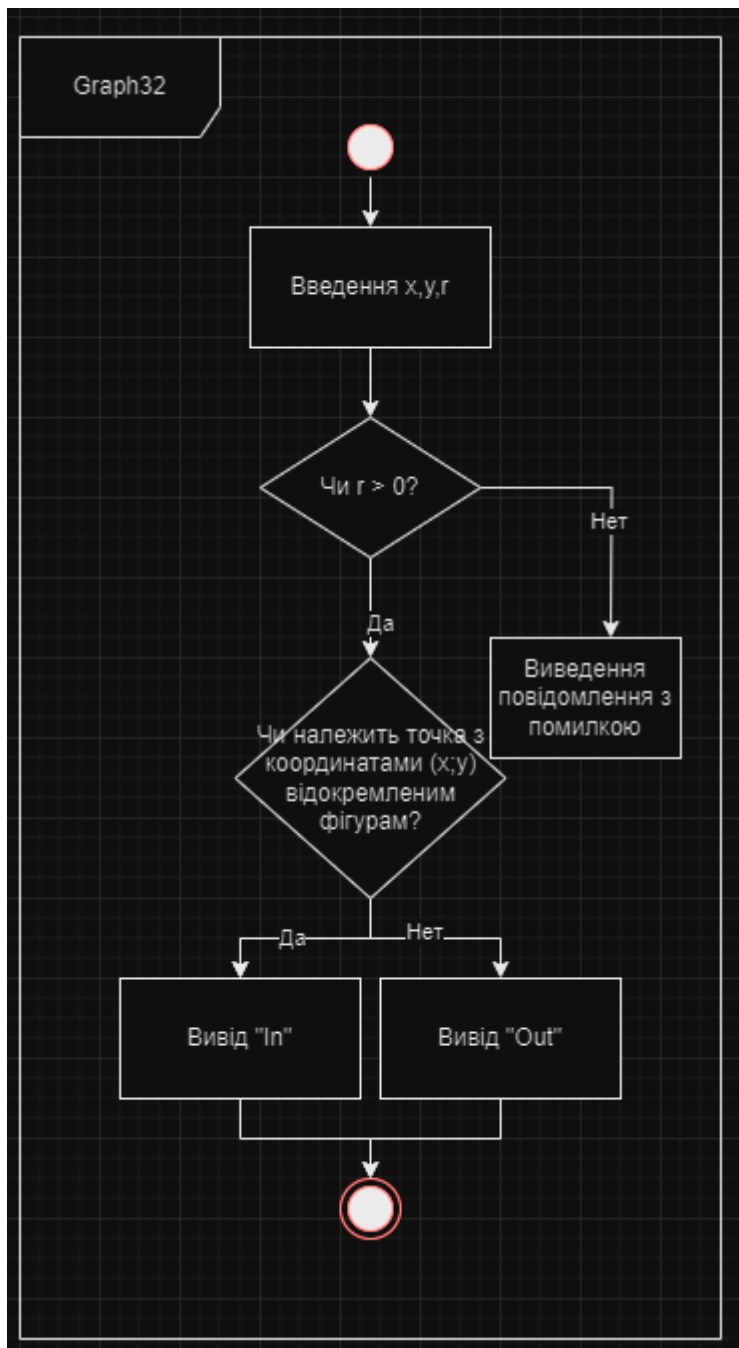


Рисунок 2 – Алгоритм вирішення Graph32

Лістинг коду вирішення задачі наведено в дод. А (стор. 6).
Екран роботи програми показаний на рис. Б.2.

Завдання 3. Вирішення задачі Series25.

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

x – змінна, float

e = 10^5 , const, верхня межа

g = 10^{-5} , const, нижня межа

Вихідні дані (ім'я, опис, тип):

Вивід інформації у консолі

Лістинг коду вирішення задачі наведено в дод. А (стор. 6).

Екран роботи програми показаний на рис. Б.3.

ВИСНОВКИ

Сьогодні ми вивчили теоретичний матеріал щодо синтаксису на мові Python і поданням у вигляді UML діаграм діяльності алгоритмів з розгалуження та циклами, а також навчилися використовувати функції, інструкції умовного переходу і циклів для реалізації інженерних обчислень.

ДОДАТОК А

Лістинг коду виконання задач If16, Graph32, Series25

```
import math

def main():
    while True:
        print(" Choose task: "
              "\n1. Task 1\n2.Task2\n3.Task3\n4.Close")
        ch = int(input("Enter task "))

        if ch == 1:
            (task1())
        elif ch == 2:
            (task2())
        elif ch == 3:
            (task3())
        elif ch == 4:
            exit()

def task1():
    num1 = float(input("Enter first number: "))
    num2 = float(input("Enter second number: "))
    num3 = float(input("Enter third number: "))

    if (num1 < num2 and num2 < num3):
        return num1 * 2, num2 * 2, num3 * 2
    else:
        return num1 * -1, num2 * -1, num3 * -1

def task2():
    try:
        x = float(input())
        y = float(input())
        r = float(input())

        sqrt = math.sqrt(2)

        if (y < 2 * r and x < 0 and x > -2 * r and y > 0 and pow(x - ((r *
sqrt)/2), 2) + pow(
            (y - (r * sqrt/2)), 2) > pow(r, 2) and x + r * sqrt > y):
            return ("In")
        elif pow((x - ((r * sqrt) / 2)), 2) + pow((y - ((r *sqrt)/2)), 2) <
pow(r,2) and x > 0 and y > 0 and x > y and x < (
            -y + ((2 * r * sqrt) / 2)):
            return ("In")
        else:
            return ("Out")
    except:
        print("Помилка")
        return 0

def task3():
    try:
        n = 1
        e = pow(10, 5)
        g = pow(10, -5)
        result = 0
```



```
        while result < e:
            result += factorial(n) / (5*n+2)
            print("Result is ")
            print(result)
            n += 1

        return result
    except:
        print("Помилка")
        return 0

def factorial(number):

    if number == 0:
        return 1
    else:
        return number * factorial(number - 1)

main()
```

ДОДАТОК Б

Скрін-шоти вікна виконання програми

```
Enter task
1
Enter first number:
4
Enter second number:
5
Enter third number:
6
(8.0, 10.0, 12.0)
```

Рисунок Б.1 – Екран виконання програми для вирішення завдання
If16

```
Enter task
2
-3
2
5
In
```

Рисунок Б.2 – Екран виконання програми для вирішення завдання
Graph32

```
Result is
0.14285714285714285
Result is
0.30952380952380953
Result is
0.6624649859943978
Result is
1.7533740769034887
Result is
6.1978185213479335
Result is
28.697818521347934
Result is
164.91403473756415
Result is
1124.9140347375642
```

Рисунок Б.3 – Екран виконання програми для вирішення завдання
Series25

