

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ Національний
аерокосмічний університет ім. М. Є. Жуковського «Харківський авіаційний
інститут»

Кафедра систем управління літальними апаратами

Лабораторна робота № 3

з дисципліни «Об'єктно-орієнтоване програмування СУ»

Тема: Тема «Реалізація класу і робота з об'єктами»

XAI.301.173.322.02 ЛР

Виконав студент гр. _____322_____

_____Гусар Анастасія_____

(підпис, дата)

(П.І.Б.)

Перевірив

_____к.т.н., зав. кафедри Пявка Є.В

__ (підпис, дата)

(П.І.Б.)

МЕТА РОБОТИ

Застосувати теоретичні знання з основ програмування на мові Python з використанням об'єктів і класів, навички використання бібліотеки для візуалізації масивів даних, і навчитися розробляти скрипти для роботи з об'єктами призначених для користувача класів.

ПОСТАНОВКА ЗАДАЧІ

Завдання 1. Визначити клас Point_n (n – номер варіанту), який реалізує абстракцію з атрибутами:

- 1) дві дійсні координати точки на площині (властивості, приховані змінні екземпляра),
 - для кожної метод-геттер (повертає відповідну координату),
 - для кожної метод-сеттер (записує відповідну координату, якщо вона у межах [-100, 100], інакше – дорівнює 0))
- 2) кількість створених екземплярів точки (змінна класу),
- 3) метод класу (повертає кількість створених примірників),
- 4) конструктор з двома параметрами (за замовчуванням),
- 5) деструктор, що виводить відповідне повідомлення,
- 6) метод, що змінює координати точки з двома вхідними дійсними параметрами:
 - зсув по x,
 - зсув по y.

Завдання 2. Виконати операції з об'єктами даного класу відповідно до варіанту (див. таб.1).

Завдання 3. Використовуючи пакет `matplotlib`, відобразити створені об'єкти в графічному вікні до і після змін.

Завдання 4. Зберегти координати точок у текстовому файлі у форматі:

номер: координата_x; координата_y – для непарних варіантів
(номер) координата_x:координата_y – для парних варіантів

ВИКОНАННЯ РОБОТИ

12. Створити список з чотирьох точок, порахувати відстань між другою і четвертою, пересунути третю на 36 вправо.

Вхідні дані:

- 1)Створити список із чотирьох точок.
- 2)Обчислити відстань між другою та четвертою точками.
- 3)Перемістити третю точку на 36 вправо.

Алгоритм вирішення:

```
import math
import matplotlib.pyplot as plt

class Point12:
    instance_count = 0 # Лічильник створених екземплярів

    def __init__(self, x=0, y=0):
        self.__x = x
        self.__y = y
        Point12.instance_count += 1

    def __del__(self):
        Point12.instance_count -= 1
        print(f"Point {self.__x, self.__y} destroyed")

    @property
    def x(self):
        return self.__x

    @x.setter
    def x(self, value):
        self.__x = value if -100 <= value <= 100 else 0

    @property
    def y(self):
        return self.__y

    @y.setter
    def y(self, value):
        self.__y = value if -100 <= value <= 100 else 0

    def move(self, dx, dy):
        self.x += dx
        self.y += dy

    @staticmethod
    def distance(p1, p2):
```

```

        return math.sqrt((p2.x - p1.x) ** 2 + (p2.y - p1.y) ** 2)

    @classmethod
    def get_instance_count(cls):
        return cls.instance_count

# Створюємо список точок
points = [Point12(10, 10), Point12(20, 30), Point12(15, 25), Point12(40, 50)]

# Обчислюємо відстань між другою та четвертою точками
distance_2_4 = Point12.distance(points[1], points[3])
print(f"Відстань між другою і четвертою точками: {distance_2_4:.2f}")

# Переміщуємо третю крапку на 36 вправо
points[2].move(36, 0)

# Візуалізація точок до і після зміни
x_coords_before = [10, 20, 15, 40]
y_coords_before = [10, 30, 25, 50]

x_coords_after = [p.x for p in points]
y_coords_after = [p.y for p in points]

plt.figure(figsize=(10, 5))

# До зміни
plt.subplot(1, 2, 1)
plt.scatter(x_coords_before, y_coords_before, color="blue", label="До зміни")
plt.title("До зміни")
plt.xlabel("X")
plt.ylabel("Y")
plt.grid()
plt.legend()

# Після зміни
plt.subplot(1, 2, 2)
plt.scatter(x_coords_after, y_coords_after, color="red", label="Після зміни")
plt.title("Після зміни")
plt.xlabel("X")
plt.ylabel("Y")
plt.grid()
plt.legend()

plt.tight_layout()
plt.show()

# Збереження координат у файл
with open("coordinates.txt", "w") as f:
    for i, p in enumerate(points):
        f.write(f"{i + 1} {p.x} {p.y}\n")

```

```
Відстань між другою і четвертою точками: 28.28  
/home/main.py:83: UserWarning: Matplotlib is currently using agg, which is a non-GUI backend, so cannot show the figure.  
  plt.show()  
Point (40, 50) destroyed  
Point (51, 25) destroyed  
Point (20, 30) destroyed  
Point (10, 10) destroyed
```

```
1) 10:10  
2) 20:30  
3) 51:25  
4) 40:50
```

Висновок

У цій роботі було створено клас для роботи з точками на площині, що включає перевірку та зміну координат, облік кількості створених об'єктів і можливість їх модифікації. У ході виконання завдання було сформовано список із чотирьох точок, обчислено відстань між другою і четвертою точками, а також виконано зсув третьої точки на 36 одиниць вправо.

Результати відобразили на графіках до і після змін, а координати точок збережено у текстовий файл.