

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ  
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ  
ІМЕНІ ІГОРЯ СІКОРСЬКОГО»

Факультет прикладної математики

Кафедра прикладної математики

Звіт

до лабораторної роботи № 3

із дисципліни «Математичні основи комп'ютерної графіки та мультимедіа»

на тему «Афінні перетворення»

Виконав:

студент групи КМ-91

*Пиндиківський Т. Р.*

Керівник:

*доцент Сирота С. В.*

## ЗМІСТ

МЕТА РОБОТИ.....	2
ПОСТАНОВКА ЗАДАЧІ.....	3
ОСНОВНА ЧАСТИНА.....	4
ДОДАТОК 1 .....	6
ДОДАТОК 2 .....	8

## **МЕТА РОБОТИ**

Розробити програмний засіб, який здійснює афінне перетворення множини точок заданих своїми координатами та відображує її на координатній площині і зберігає зображення в одному з графічних форматів.

## ПОСТАНОВКА ЗАДАЧІ

1. Скачати файл з датасетом (датасет №4). Файл в текстовому форматі містить пари цілих чисел які є координатами точок.
2. Необхідно написати програму будь якою мовою з використанням будь яких бібліотек, яка:
  - Зчитує датасет з файлу;
  - Виконує задане афінне перетворення з датасетом;
  - Встановлює розміри вікна (полотна – canvas size) **960x960** пкс;
  - Відображає датасет після афінного перетворення;
  - Виводить результати у файли будь-якого графічного формату.

## ОСНОВНА ЧАСТИНА

1. Для виконання програми спочатку імпортується модуль *Pillow* та модуль *math* (забезпечує роботу з математичними функціями).  
  
*from PIL import Image, ImageColor*, що забезпечує функціонал для роботи з графічними зображеннями, зокрема їх створенням та редагуванням.
2. Створюється об'єкт зображення *im = Image.new('1', (540,960))* з режимом «1», що визначає тип та глибину пікселів нового зображення : 1-бітні пікселі чорного або білого кольору, які зберігаються одним пікселем на байт. Також визначається розмір зображення, яке створюється: у даному випадку це 540 – ширина, 960 – довжина.
3. Конструкція *with open("DS4.txt", "r") as file*: дозволяє взаємодіяти з файлами і забезпечує роботу із вмістом текстових даних. Також гарантує закриття файлу в будь-якому випадку.
4. Зчитування вмісту файлу відбувається поки не буде досягнуто його кінця.
5. З використанням циклу *for* виконується зчитування вмісту файлу по рядку : *line=file.readline()*.
6. Шляхом форматування кожного отриманого рядка отримуємо координати пікселя : *x, y = (int(i) for i in line.strip().split(' '))*
7. Із застосуванням методу *im.putpixel((x,y), ImageColor.getcolor('yellow', '1'))* відбувається замалювання пікселя на зображенні, що має координати *x,y* у колір, переданий у метод.
8. Отримане заображення зберігається після виконання методу *im.save('image.png')*, якому передається назва та формат для збереження графічного зображення.
9. Потім відбувається перетворення кута градусної міри у радіанну: *angle = math.radians(50)*
10. Зчитуються розміри звичайної фотографії : *(x,y) = im.size*

11. Трансформуються координати «крайніх» точок з використанням власних функцій *rot\_x*, *rot\_y*, що обчислюють нові координати точки після повороту на кут *angle*.
12. Знаходяться максимальні значення координат зображення по кожній з осей :  $m_{nx} = \min(x_{extremes})$ ,  $m_{xx} = \max(x_{extremes})$ ,  $m_{ny} = \min(y_{extremes})$ ,  $m_{xy} = \max(y_{extremes})$
13. З використанням функції *transform()* відбувається безпосередньо виконання афінного перетворення :  
  

$$im = im.transform((int(round(m_{xx}-m_{nx})),int(round((m_{xy}-m_{ny})))),Image.AFFINE,(math.cos(angle),math.sin(angle),-m_{nx},-math.sin(angle),math.cos(angle),-m_{ny}),resample=Image.BILINEAR)$$
14. Отримане зображення зберігається у форматі «.png» із використанням функції *save()*.

## ДОДАТОК 1



Рис.1 – Результат генерації початкового зображення



Рис. 2 – Результат виконання афінного перетворення



## ДОДАТОК 2

Текст програми, що реалізує малювання зображення за отриманим переліком пікселів :

```
from PIL import Image, ImageColor

im = Image.new('1', (540,960))

with open("DS4.txt", "r") as file:
    for line in file:
        line=file.readline()
        x, y = (int(i) for i in line.strip().split(' '))
        im.putpixel((x,y), ImageColor.getcolor('white', '1'))

im.save('image.png')
```

Текст програми, що реалізує афінне перетворення :

```
from PIL import Image
import math

def rot_x(angle,ptx,pty):
    return math.cos(angle)*ptx + math.sin(angle)*pty

def rot_y(angle,ptx,pty):
    return -math.sin(angle)*ptx + math.cos(angle)*pty

angle = math.radians(50)

im = Image.open('image.png')

(x,y) = im.size

xextremes      =      [rot_x(angle,0,0),rot_x(angle,0,y-1),rot_x(angle,x-1,0),rot_x(angle,x-1,y-1)]
yextremes      =      [rot_y(angle,0,0),rot_y(angle,0,y-1),rot_y(angle,x-1,0),rot_y(angle,x-1,y-1)]
mnx = min(xextremes)
mxx = max(xextremes)
mny = min(yextremes)
mxy = max(yextremes)
im      =      im.transform((int(round(mxx-mnx)),int(round((mxy-mny))),Image.AFFINE,(math.cos(angle),math.sin(angle),-mnx,-math.sin(angle),math.cos(angle),-mny),resample=Image.BILINEAR)
im.save('transformation.png')
```