

Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Сибирский государственный университет телекоммуникаций и  
информатики»  
(СибГУТИ)

Кафедра прикладной математики и кибернетики

Представление Графической Информации  
Лабораторная работа №3

Выполнил:  
Студент IV курса ИВТ,  
группы ИП-713  
Михеев Никита Алексеевич

Работу проверил:  
доцент кафедры ПМиК  
Перцев И.В.

Новосибирск 2020 г.

## **Задание**

Преобразовать BMP файл, развернув растр на 90 градусов.

## **Результат работы**

В результате работы была реализована программа на языке Python версии 3.9. Для того, чтобы развернуть растр на 90 градусов был написан цикл, который считывал по «столбцу» изображение и помещал пиксели в новую переменную, но уже построчно. Данная новая переменная была записана в отдельный файл и в результате чего изображение оказалось перевернутым на 90 градусов против часовой стрелки:



Рис.1 – результат работы программы.

Немного изменив алгоритм, можно развернуть изображение по часовой стрелке:



Рис.2 – результат работы измененного алгоритма

## Листинг

### Lab3.py:

```
from PIL import Image

def main() -> int:
    image = Image.open("image2.bmp")
    image_matrix = image.load()

    rotated_matrix = []
    for i in range(image.width):
        rotated_matrix.extend([image_matrix[i, j] for j in
range(image.height)])

    rotated_image = Image.new('RGB', (image.height, image.width))
    rotated_image.putdata(rotated_matrix)
    rotated_image.save("image_rotated.bmp", "bmp")
    return 0

if __name__ == '__main__':
    exit(main())
```

### Измененный алгоритм, для поворота «по часовой»:

```
from PIL import Image

def main() -> int:
    image = Image.open("image2.bmp")
    image_matrix = image.load()

    rotated_matrix = []
    for i in range(image.width):
        rotated_matrix.extend([image_matrix[image.width - i - 1,
image.height - j - 1] for j in range(image.height)])

    rotated_image = Image.new('RGB', (image.height, image.width))
    rotated_image.putdata(rotated_matrix)
    rotated_image.save("image_rotated.bmp", "bmp")
    return 0

if __name__ == '__main__':
    exit(main())
```