|  |  |  |
| --- | --- | --- |
| Защищено:  Гапанюк Ю. Е.  "\_\_"\_\_\_\_\_\_\_\_\_\_\_\_\_2024 г. |  | Демонстрация:  Гапанюк Ю. Е.  "\_\_"\_\_\_\_\_\_\_\_\_\_\_\_\_2024 г. |

**Отчет по лабораторной работе №4 по курсу**

**«Парадигмы и конструкции языков программирования»**

#### Тема работы: " **Инвертирование битов в картинке** "

|  |  |
| --- | --- |
| ИСПОЛНИТЕЛЬ: | Пыжьянов Александр Сергеевич |
| студент группы  ИУ5Ц-52Б | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
|  | (подпись) |
| Гапанюк Ю.Е. | "\_\_"\_\_\_\_\_\_\_\_\_\_\_\_\_2024 г. |

Москва, МГТУ 2024

**СОДЕРЖАНИЕ**

[Цель лабораторной работы 3](#_Toc176887611)

[Практическое задание 3](#_Toc176887612)

[Листинг программы 4](#_Toc176887613)

[Результат работы программы 5](#_Toc176887614)

## Цель лабораторной работы

Научиться работе с библиотекой PIL.

## Практическое задание

Дана картинка. Нужно инвертировать в ней все биты и получить новое изображение.

## Листинг программы

from PIL import Image

image = Image.open("img1.bmp")

pixels = list(image.getdata())

for i in range(len(pixels)):

new\_rgb = list(pixels[i])

for j in range(3):

tec = bin(new\_rgb[j])[2:]

inverse = ''

for k in tec:

if k == '0':

inverse += '1'

else:

inverse += '0'

inverse = int(inverse, 2)

new\_rgb[j] = inverse

pixels[i] = tuple(new\_rgb)

new\_image = Image.new(image.mode, image.size)

new\_image.putdata(pixels)

new\_image.save("inverse.bmp")

## Результат работы программы

Исходная картинка:



Что получилось в результате работы программы:

