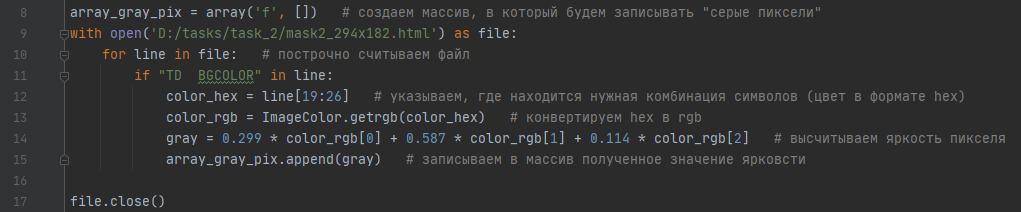
**Программирование в системах информационной безопасности  
 Лабораторный практикум**

Васюткин Александр Олегович, группа 181-331

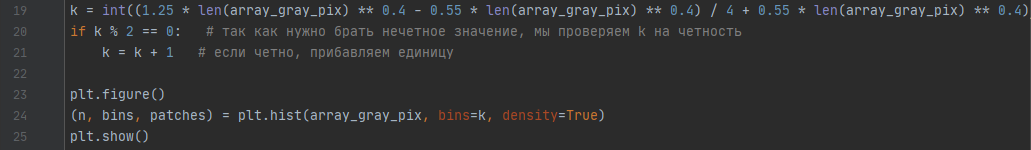
**Задание 2.**

1. Преобразовать цвет каждого пикселя по правилу: Яркость пикселя:

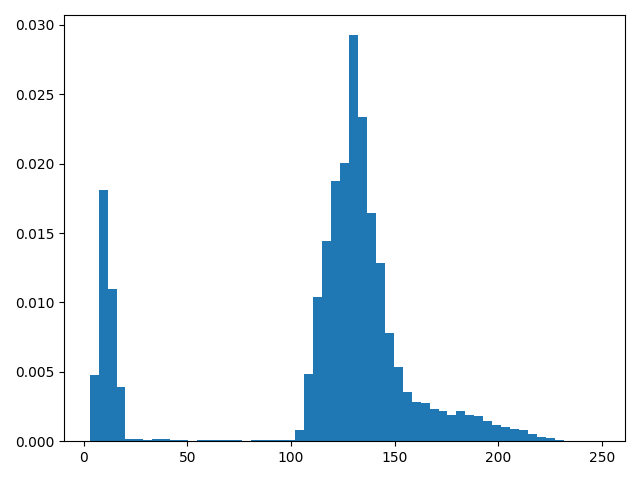


В строке 8 создается массив, в который далее будут записаны яркости пикселей, преобразованных в оттенки серого. Далее следует цикл, который построчно считывает HTML-файл, если строка содержит последовательность символов, сигнализирующих о наличии цвета в строке, то мы вычленяем и записываем в переменную цвет в формате HEX. После преобразовываем его в формат RGB (при помощи метода из библиотеки PIL), высчитываем его яркость по формуле, и записываем в массив, предназначенный для яркостей пикселей. Закрываем файл.

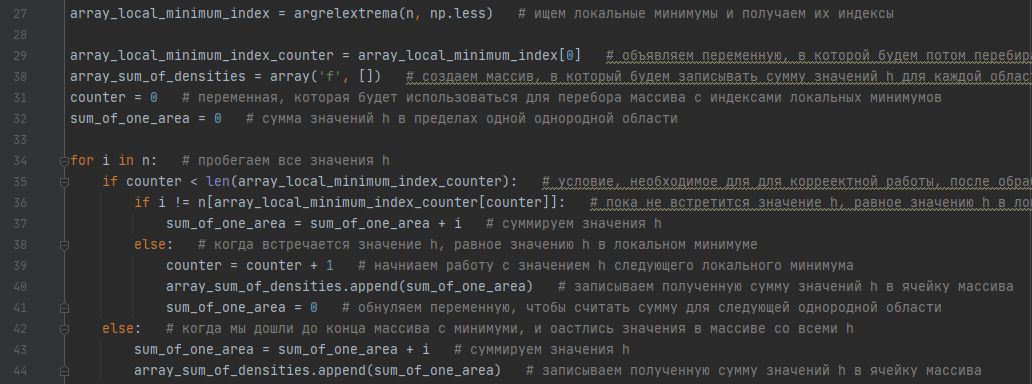
1. По значениям яркости программно построить гистограмму.



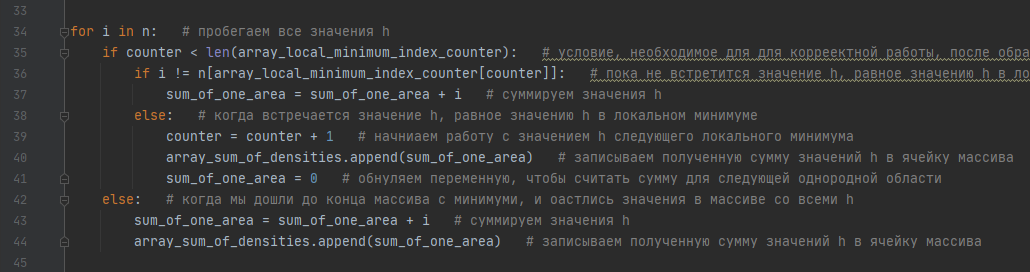
Чтобы построить гистограмму, необходимо определить оптимальное число интервалов k из промежутка , что мы и делаем в 19 строке, объединив формулы нахождения минимального и максимального значений интервала. Так как k должно быть нечетным, мы проверяем его четность, и, если это необходимо, прибавляем единицу. Наконец, найдя количество интервалов, мы можем построить гистограмму (при помощи библиотеки matplotlib), записав в переменную n значения плотности для каждого интервала, a в bins – значения границ каждого интервала. Полученная гистограмма:



1. Выделить по характеристикам гистограммы области качественной однородности пикселей.

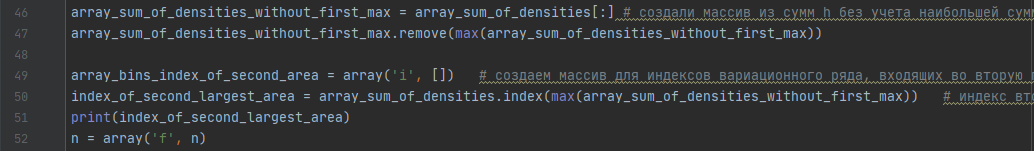


Записываем в массив индексы плотностей h, соответствующих локальным минимумам в общем списке плотностей. Локальные минимумы ищем при помощи метода из библиотеки ScyPi. В строке 29 мы создаем переменную, которую будем использовать для перебора индексов локальных минимумов и присваиваем ей значение, соответствующее индексу первого локального минимума. В 30 строке объявляем массив, в который будем записывать суммы значений плотностей h в промежутках между локальными минимумами. Далее объявляем две переменные, необходимые для определения конца массива с индексами локальных минимумов и суммирования значений плотностей в пределах одной области качественной однородности.

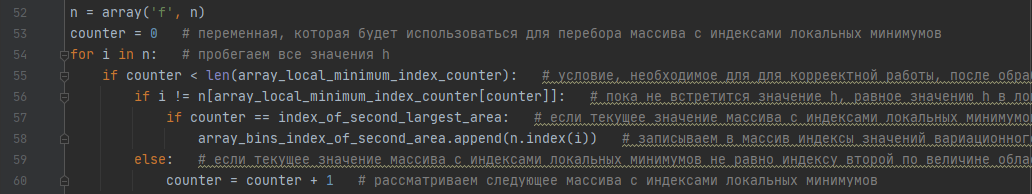


На данном фрагменте программа складывает значения плотностей h в каждой области качественной однородности (это пригодится для нахождения второй по частоте области однородности). В цикле мы поочередно обрабатываем все значения h, полученные из гистограммы, сравнивая каждое значение со значением локального минимума. Пока не встретится значение h, равное значению локального минимума, мы суммируем эти значения h. Когда встречается значение h, равное значению локального минимума, мы записываем полученную сумму плотностей в массив, обнуляем переменную, где хранилась сумма, и переходим к следующему локальному минимуму, чтобы продолжить дальше обрабатывать значения плотностей. Часть цикла (42-44 строки) необходима для обработки значений плотностей последней области однородности (когда все локальные минимумы закончились).

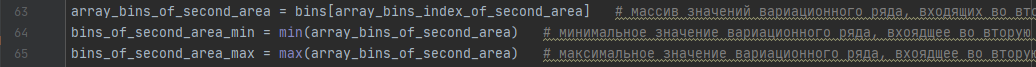
1. Заменить цвета пикселей, соответствующих второй по частоте области однородности, на красный.



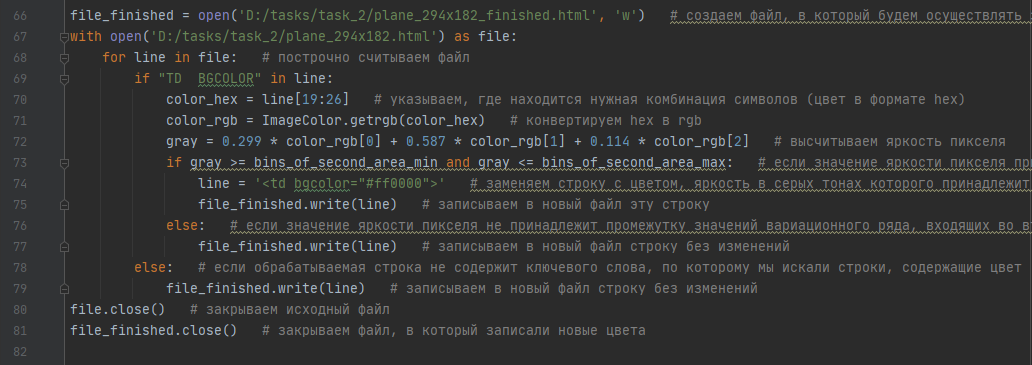
Для начала нам необходимо найти вторую по частоте область однородности, для этого мы копируем массив с суммами плотностей в другой массив, после чего удаляем из него элемент с максимальным значением суммы плотностей. В строке 49 мы объявляем массив, в который будут записаны индексы значений вариационного ряда, входящих во вторую по частоте область однородности. Далее мы ищем значение второго (первой в массиве без максимальной суммы плотностей) максимума из сумм плотностей и находим его индекс в массиве со всеми суммами плотностей (в том, который копировали).



Теперь наша задача состоит в том, чтобы найти значения вариационного ряда, входящие во вторую по величине область однородности. Когда встречается индекс значения локального минимума из вариационного ряда, равный индексу значения локального минимума, сигнализирующего о второй по величине области однородности, значения в вариационном ряду, идущие после него, записываются в массив (до тех пор, пока не встретится следующий локальный минимум). На выходе мы получаем массив индексов значений вариационного ряда, принадлежащих второй по величине области однородности.



Когда мы нашли индексы значений вариационного ряда, соответствующих второй по величине области однородности, мы можем найти сами значения (строка 63). В 64-65 строках мы находим минимальное и максимальное значения вариационного ряда, входящие во вторую по величине область однородности, чтобы знать, цвета из какого интервала яркости пикселей нам нужно заменять.



На представленном фрагменте программы мы создаем файл для записи в него содержимого исходного файла, изменяя цвета пикселей, соответствующих второй по величине области однородности, на красный. В цикле мы построчно считываем исходный файл (и записываем строки в новый файл), находим строки, в которых есть информация о цвете пикселя. После чего мы вычленяем из строки значение цвета в формате HEX, конвертируем его в RGB и по формуле вычисляем яркость пикселя. Если значение яркости рассматриваемого пикселя принадлежит промежутку значений яркости пикселей, соответствующих второй по величине области однородности, то мы заменяем этот цвет на красный, после чего записываем полученную строку в файл. В итоге у нас получается файл, аналогичный исходному, отличающийся от него только тем, что цвета пикселей, соответствующих второй по величине области однородности, заменены на красный. Не забываем закрывать файлы.

Результат работы программы:

