1. Сначала пользователю предлагается ввести количество градаций серого цвета, с которыми программа будет работать.
2. Затем с помощью функции zeros создаются пустые массивы, в которые будут заноситься данные:  
   -images – матрица, которая хранит бинарные комбинации, характеризующие изображения писателей. Одной строке матрицы соответствует свое изображение.

-notpreparedimages – трехмерный массив. Первая координата отвечает за порядковый номер изображения писателя, вторая и третья – интенсивность серого цвета на пикселе изображения, лежащем на данных координатах.

-Buf – вектор-строка, которая хранит бинарную комбинацию, характеризующую изображение писателя.

1. Работа с каждым эталонным изображением.
2. Читаем из файла полноцветное изображение и переводим его в массив.
3. Изменяем размер полученного изображения (массива) на размер 100\*100 пикселей.
4. Преобразуем полноцветное изображение в полутоновое, изменяя R-, G-, В-составляющие пикселов полноцветного изображения в соответствующие им значение яркости.
5. Преобразуем полутоновое изображение в палитровое, сохранив саму палитру в свою матрицу map. Палитра mар содержит градации серого цвета равномерно распределенные в диапазоне [0, 1].
6. Выводим полученное изображение в папку, чтобы убедиться, что изображение стало палитровым с размером 100\*100.
7. Переводим значения массива в тип данных double.
8. Сохраняем изображение в массив notpreparedimages .
9. Переводим массив в вектор-строку размером 10000 и сохраняем в массив Buf.
10. Переводим вектор-строку в бинарную вектор-строку размером 10000\*< количество градаций серого цвета> по математической модели получения бинарной строки для полутоновых черно-белых изображений с последующим сохранением в массив images.
11. Переводим массив images к типу double.
12. Описываем эталонные портреты писателей для последующего вывода, если тестовое изображение будет распознано.
13. Работа с тестовым изображением.
14. Читаем из файла полноцветное изображение и переводим его в массив.
15. Изменяем размер полученного изображения (массива) на размер 100\*100 пикселей.
16. Преобразуем полноцветное изображение в полутоновое, изменяя R-, G-, В-составляющие пикселов полноцветного изображения в соответствующие им значение яркости.
17. Преобразуем полутоновое изображение в палитровое, сохранив саму палитру в свою матрицу map. Палитра mар содержит градации серого цвета равномерно распределенные в диапазоне [0, 1].
18. Переводим массив в вектор-строку размером 10000 и сохраняем в массив Buf.
19. Приводим значения массива Buf к типу данных double.
20. Переводим вектор-строку Buf в бинарную вектор-строку размером 10000\*< количество градаций серого цвета> по математической модели получения бинарной строки для полутоновых черно-белых изображений с последующим сохранением в массив testimage.
21. Перемножаем эталонные и тестовую бинарную комбинацию для нахождения среди эталонных той, которая максимально похожа на тестовую.
22. Назначаем порог схожести, позволяющий определить, было ли распознано тестовое изображение или нет.
23. Если порог превышен, то выводим изображение из массива notpreparedimages на экран, в консоли выводим соответствующее описание портрета писателя.
24. Иначе в консоли выводим «не удалось распознать».