**Георгиевский региональный колледж**

**«Интеграл»**

ОТЧЕТ

по дисциплине «Flash-программирование»

Студента 4 курса группы ПК-81

Гончарова Александра

2021

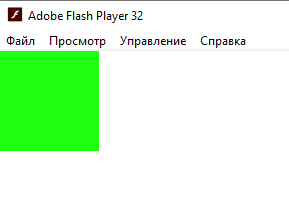
## Лабораторная работа №12 «Работа с изображениями»

**Цель работы:** изучить принципы работы с изображениями в ActionScript.

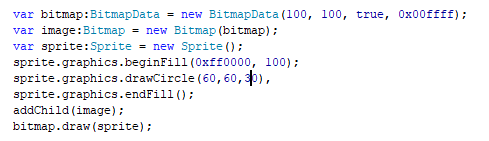
Ход работы

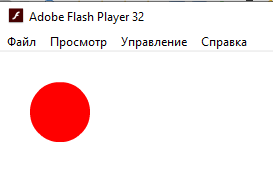
1. Разработайте код программы, в которой создается BitmapData с зелёной заливкой, после чего он выводится на экран посредством Bitmap.



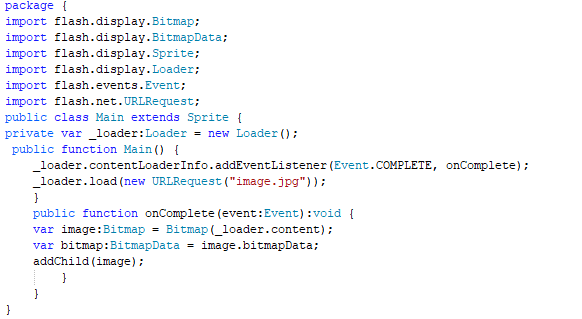


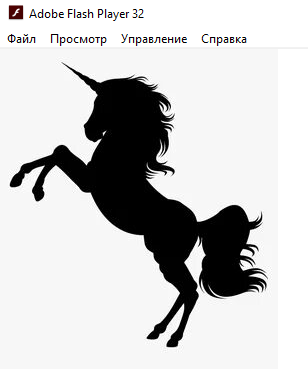
1. Разработайте код программы, в которой создаются BitmapData и Sprite. Затем методами программного рисования в спрайте рисуется круг и этот спрайт отрисовывается в BitmapData.



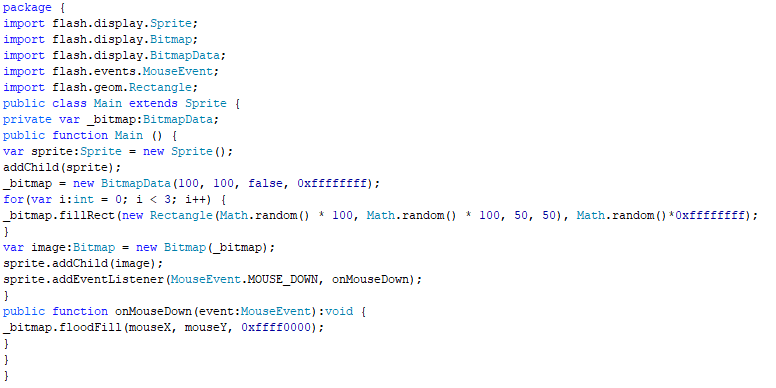


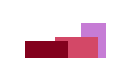
1. Разработайте код программы, в которой загружается внешнее изображение в экземпляр класса Bitmap.



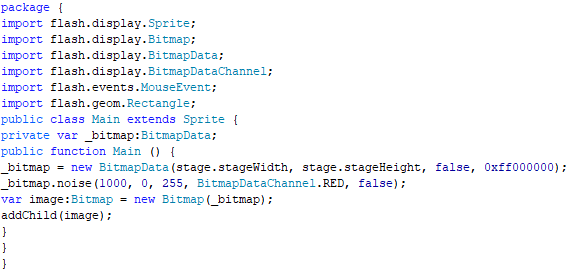


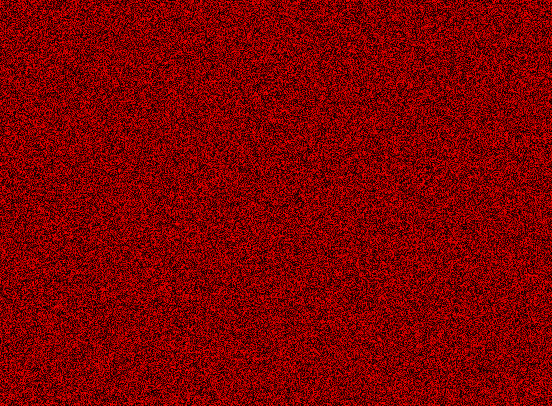
1. Разработайте код программы, в которой последовательно создаётся 3 заливки разного цвета.



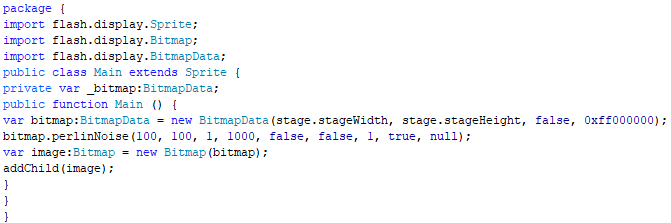


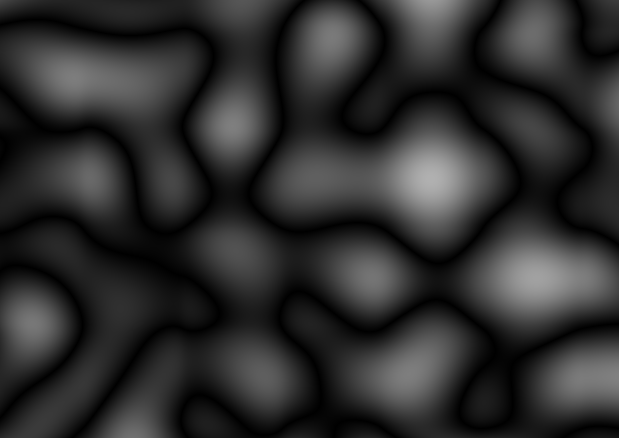
1. Разработайте код программы, в которой загружается внешнее изображение и к нему применяется эффект шума.



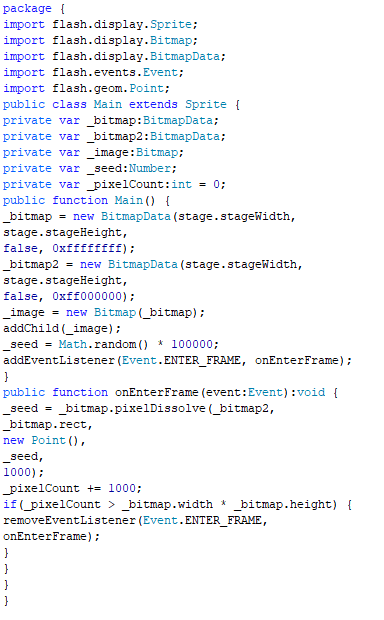


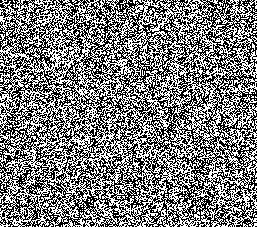
1. Разработайте код программы, в которой загружается внешнее изображение и к нему применяется эффект шума Перлина.



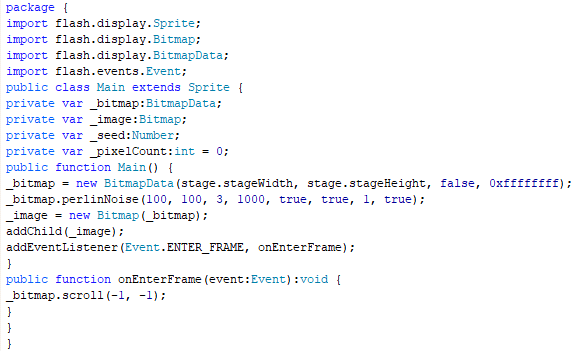


1. Разработайте код программы, в которой загружается два внешних изображения и происходит растворение одного изображения в другом.





1. Разработайте код программы, в которой создаётся эффект двигающихся облаков.





### Контрольные вопросы

1. Какой класс AS 3 представляет пикселы растрового изображения?

***Класс BitmapData*** представляет пикселы растрового изображения и содержит множество методов для добавления содержимого и работы с этим изображением.

1. Опишите процесс добавления растрового изображения в список отображения.

Создать экземпляр класса можно, вызвав конструктор:

var bitmap:BitmapData = new BitmapData (width, height, transparent, fillColor);

1. Перечислите действия, которые можно применить к растровым изображениям.

* Наполнение растрового изображения графическим содержимым
* Загрузить внешнее изображение в экземпляр класса Bitmap
* Создание прямоугольной заливки
* Создание заливки области
* Создание шума
* Создание шума Перлина
* Применение фильтра к растровому изображению
* Растворение одного растрового изображения в другом
* Прокрутка растрового изображения

1. Опишите процесс создания прямоугольной заливки и заливки по области.

Класс BitmapData имеет базовый инструмент прямоугольной заливки. В него достаточно передать прямоугольную область, которую требуется закрасить, и цвет заливки:

\_bitmap.fillRect(прямоугольник, цвет);

Прямоугольник должен быть экземпляром класса flash.geom.Rectangle. Экземпляр этого класса можно создать посредством его конструктора, задав х и у

Заливка области: сначала создается растровое изображение и ряд случайных квадратов.

1. Опишите отличие простого шума от шума Перлинга.

Как и метод noise(), метод perlinNoise() создает случайные узоры в растровом изображении. Однако для создания шума Перлина применяется алгоритм, в результате работы которого получаются гладкие, естественные на вид текстуры. Алгоритм был разработан Кеном Перлином (Ken Perlin) для сознания текстур в фильме «Тгоп». Эти текстуры идеально подходят для моделирования взрывов, дыма, воды и многих других естественных эффектов, а поскольку они генерируются алгоритмом, то требуют намного меньше памяти для хранения

1. Каким образом можно применить фильтры к растровому изображению?

Cуществует два метода применения фильтра к растровому изображению.

Первый - при помощи метода applyFilter( ) непосредственно к самому BitmapData. Как и некоторые другие методы класса BitmapData, этот метод применяется к тому объекту BitmapData, который является целевым изображением, и может принимать другой объект BitmapData в качестве исходного растрового изображения:

destBmp.applyFilter(srcBmp, sourceRect, destPoint, filter);

где:

• srcBmp - растровое изображение, к которому применяется фильтр.

• sourceRect- определенная прямоугольная область исходного изображения, к которой применяется фильтр.

• Параметр destPoint определяет верхнюю левую точку в целевом изображении, от которой будут изменяться пикселы.

• filter - это, конечно, экземпляр класса BitmapFilter, который будет применен.

Другой способ применения фильтра к растровому изображению - добавить его в свойство filters объекта Bitmap, заключающего в себе объект BitmapData. Это еще один недеструктивный способ применения фильтра, поскольку данные пикселов BitmapData остаются нетронутыми.