

Министерство образования Республики Беларусь

Учреждение образования
«Гомельский государственный технический
университет имени П. О. Сухого»

Институт повышения квалификации
и переподготовки кадров

Кафедра «Информатика»

Д. П. Андреева, М. А. Старовойтова

АНИМАЦИОННАЯ ГРАФИКА

**ЛАБОРАТОРНЫЙ ПРАКТИКУМ
по одноименной дисциплине
для слушателей специальности 1-40 01 74
«Web-дизайн и компьютерная графика»
заочной формы обучения**

Гомель 2013

УДК 004.928(075.8)
ББК 32.973.202-018.2я73
А65

*Рекомендовано кафедрой «Информатика» ГГТУ им. П. О. Сухого
(протокол № 3 от 25.09.2012 г.)*

Рецензент: зав. каф. «Инженерная графика» ГГТУ им. П. О. Сухого
канд. техн. наук, доц. *О. М. Остриков*

Андреева, Д. П.

А65 Анимационная графика : лаборатор. практикум по одноим. дисциплине для слушателей специальности 1-40 01 74 «Web-дизайн и компьютерная графика» заоч. формы обучения / Д. П. Андреева, М. А. Старовойтова. – Гомель : ГГТУ им. П. О. Сухого, 2013. – 47 с. – Систем. требования: PC не ниже Intel Celeron 300 МГц ; 32 Mb RAM ; свободное место на HDD 16 Mb ; Windows 98 и выше ; Adobe Acrobat Reader. – Режим доступа: <http://library.gstu.by/StartEK/>. – Загл. с титул. экрана.

Содержится подробное описание всех возможностей, которые могут быть реализованы с помощью программы Adobe Flash CS5. В частности, рассматриваются такие важные разделы анимационной графики, как покадровая анимация, различные виды автоматической анимации, в том числе и работа с текстом, а также интерактивные возможности Flash.

Для слушателей ИПК и ПК специальности 1-40 01 74 «Web-дизайн и компьютерная графика» заочной формы обучения.

**УДК 004.928(075.8)
ББК 32.973.202-018.2я73**

© Учреждение образования «Гомельский
государственный технический университет
имени П. О. Сухого», 2013

СОДЕРЖАНИЕ

Лабораторная работа №1. Создание и редактирование иллюстраций	4
Лабораторная работа №2. Покадровая анимация	7
Лабораторная работа №3. Анимация формы	11
Лабораторная работа №4. Анимация движения	14
Лабораторная работа №5. Текст.....	20
Лабораторная работа №6. Маска	24
Лабораторная работа №7. Интерактивность во Flash.....	30

Лабораторная работа №1. Создание и редактирование иллюстраций

Цель работы: с помощью различных инструментов Flash научиться создавать иллюстрации.

Теоретические сведения

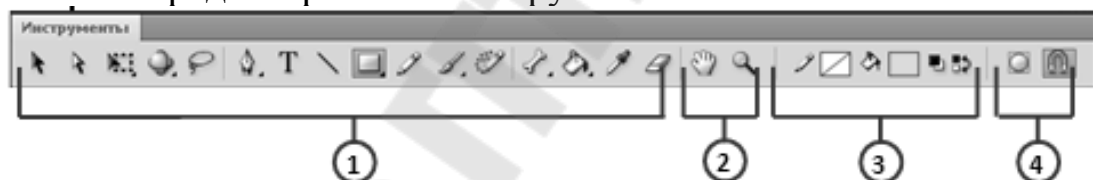
Инструменты, доступные на панели «Инструменты», предназначены для рисования, закрашки, выделения и изменения объектов, а также изменения вида рабочей области. Панель «Инструменты» делится на четыре области.

1. Область инструментов содержит инструменты рисования, закрашки и выделения.

2. Область просмотра содержит инструменты масштабирования и панорамирования в окне приложения.

3. Область цветов содержит модификаторы для обводки и заливки.

4. Область параметров содержит модификаторы текущего выбранного инструмента. Модификаторы влияют на операции рисования и редактирования инструмента.



Если в одной позиции доступны несколько инструментов, то у верхнего инструмента группы (последних примененных) в правом нижнем углу значка имеется стрелка. Она указывает, что в раскрывающемся меню присутствуют дополнительные инструменты. Если нажать и удерживать кнопку мыши на значке, в раскрывающемся меню будут отображены остальные инструменты этой группы.

На панели инструментов, в области «Инструменты рисования»



собраны инструменты для рисования простых линий и фигур, для нанесения узоров, для декоративного рисования.

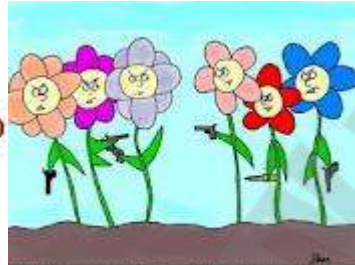
Задание. Используя различные инструменты, нарисовать следующие картинки.



Zalmen Park



www.wallpaperbase.com



And before anybody really knew what was happening, another gang war had erupted between the croissants and donuts.



МОИ
ЛЮБИМЫЕ
ПЕСНИ





Ход выполнения работы

1. Откройте Adobe Flash CS5, создайте новый документ и сохраните его в своей папке под именем lab_rabota1.
2. Выберите один из рисунков, представленных в задании.
3. С помощью различных инструментов рисования создайте контур рисунка (для сложных объектов можно использовать слои).
4. Примените, если необходимо, инструменты выделения, копирования и группировки объектов.
5. Используя различные способы заливки, раскрасьте полученный контур.
6. Сохраните изменения в созданном документе.

Контрольные вопросы

1. Что такое панель инструментов и панель свойств?
2. Что такое контур, обводка, заливка?
3. Какие бывают режимы рисования и чем они отличаются?
4. Назовите основные инструменты рисования.
5. Какие операции можно производить над объектами с помощью инструмента Свободное преобразование?
6. Каким образом можно изменить цвет объекта (цвет обводки и цвет заливки)? (См. электронный курс лекций по дисциплине «Анимационная графика», раздел 2, создание и редактирование иллюстраций).

Лабораторная работа №2. Покадровая анимация

Цель работы: получить навыки создания покадровой анимации во Flash.

Теоретические сведения

Любая анимация состоит из кадров. **Кадры** — это изображения последовательных фаз движения объектов или их частей. При просмотре последовательности кадров возникает иллюзия оживления изображенных на них статичных объектов.

Кадры во Flash делятся на ключевые кадры, ключевые кадры свойств и промежуточные. **Ключевые кадры** играют особую роль: в них фиксируются новые объекты или существенные изменения в изображаемых объектах. **Ключевые кадры свойств** — это кадры, в которых определяется изменение свойств анимации объекта. В **промежуточных кадрах** изображение не меняется (дублирует предыдущий кадр) или строится программой автоматически. Ключевые кадры и ключевые кадры свойств управляют последовательностью событий и анимацией.

Еще одним важнейшим элементом анимации являются **слои**. Обычно анимация содержит сразу несколько объектов, как анимированных так и статичных. Для упрощения работы с объектами и для создания независимой анимации отдельных элементов изображения в технологии Flash предусмотрена возможность размещения их на разных слоях. Слои можно представить как стопку

прозрачных пленок с изображениями. Рассматривают слои сверху, поэтому объект из верхнего слоя может перекрывать объекты или части объектов в слоях, расположенных ниже. Взаимное расположение элементов изображения можно менять, меняя порядок слоев. Фоновый рисунок размещается в нижнем слое.


Покадровая анимация – это традиционный вид анимации, при котором все фазы движения прорисовываются вручную в ключевых кадрах. Ролик с покадровой анимацией может содержать и промежуточные кадры, в которых изображение не изменяется.

Порядок выполнения работы

I. Покадровая анимация увеличения листа.

1. Создайте новый ключевой кадр в кадре 1.


2. Выберите темно-зеленый цвет контура, толщину линии 3.


С помощью инструмента  **Перо** нарисуйте контур листа с жилкой посередине. Залейте его светло-зеленым цветом.




3. Выделите кадр 1 и вставьте 4 новых ключевых кадра, нажимая **F6**. Сейчас во всех кадрах — одинаковое изображение. Мы изменим фильм так, чтобы на первом кадре был самый маленький лист, а потом — все больше и больше.

4. Включите инструмент  **Стрелка** и выберите кадр 1. При этом все его содержимое будет также выделено.

5. Для уменьшения листа на 1-ом кадре примените инструмент **Свободные преобразования**, который включается кнопкой  на панели инструментов, клавишей **Q**, или командой **Свободное преобразование** из контекстного меню (после нажатия правой кнопки мыши). Вокруг выделенного объекта появляется рамка с маркерами, такая же, как, например, в редакторе *Word*. Теперь можно:

- изменить **размеры**, перетаскивая за маркеры (при нажатой клавише **Shift** пропорции не меняются, при нажатии **Alt** точка вращения остается неподвижной);
- **повернуть** объект вокруг центра, обозначенного белым кружком (для этого нужно подвести мышь к углу рамки снаружи, чтобы курсор принял вид ) , центр вращения можно перетаскивать мышкой;





- **перекосить** объект, схватив и потянув мышкой за рамку между маркерами (курсор );
- **перетащить углы** рамки при нажатой клавише **Ctrl**.

6. Уменьшите размеры листа примерно в 3 раза. Перетащите центр вращения в левый нижний угол листа и поверните лист немного более вертикально. Таким же способом постройте промежуточные рисунки в кадрах 2-4. Просмотрите фильм.

Другие варианты преобразования применяются с помощью меню **Модификация—Преобразовать**. В частности можно

- повернуть объект на 90 градусов;
- построить отражение объекта по вертикали;
- выполнить тонкую настройку контура с помощью команды **Огибающая**.

Когда включен инструмент **Свободное преобразование**, в нижней части панели инструментов появляются кнопки, позволяющие включать какой-то один вид преобразований:

-  **Вращение и наклон;**
-  **Масштаб;**
-  **Искажение;**
-  **Огибающая.**

При нажатии клавиш **Ctrl+T** открывается панель **Преобразовать**, на которой можно точно задать угол поворота и скоса. При всех трансформациях объект не «портится». Если надо отменить все преобразования и вернуть его к исходному виду, выберите пункт меню **Модификация—Преобразовать—Удалить преобразование**.

Скорее всего, анимация с первого раза получится не очень равномерная. Для настройки покадровой анимации есть специальное средство, которое называется **Многослойная структура**. Если включить этот режим, при обработке очередного кадра вы будете видеть полупрозрачные изображения соседних кадров, что поможет улучшить переходы. Для включения этого режима надо щелкнуть по кнопке **Многослойная структура** в нижней части временной шкалы. Если нужно оставить **только контуры** фигур на соседних кадрах (чтобы заливка не мешала), можно щелкнуть по кнопке **Контуры многослойной структуры**. Область видимости регулируется движками над временной шкалой.

Щелкнув по кнопке **Изменить маркеры многослойной структуры**, вы увидите меню, с помощью которого можно изменить ее настройки

- **Всегда показывать маркеры;**
- **Привязка шлейфа** (иначе он смещается вслед за читающей головкой);
- выделять 2 или 5 кадров слева от текущего — **Шлейф 2** или **Шлейф 5;**
- **Поместить все кадры в шлейф.**

7. Включите режим **Контур** многослойной структуры, установите видимость только двух соседних кадров. Попробуйте сделать более ровные переходы между кадрами. Проиграйте фильм.

Справа от кнопки **Контур** многослойной структуры расположена кнопка **Редактировать несколько кадров**. Она позволяет заменять положение объекта сразу во всех кадрах, ограниченных движками зоны видимости (над временной шкалой).

8. Включите режим **Редактировать несколько кадров** и раздвиньте движки видимости так, чтобы они охватывали все кадры с 1-ого по 5-ый. Включите инструмент **Стрелка**, выделите рамкой изображение листа на всех кадрах и перетащите вправо, чтобы слева можно было поставить еще один лист. В этом режиме можно выполнять все действия также, как и обычно, разница в том, что захватываются несколько кадров.

9. Выделите снова все изображения листа и скопируйте их влево, перетащив при нажатой клавише **Alt**. Постройте отражение нового листа (**Модификация — Преобразовать — Отразить слева направо**) и переместите его так, чтобы нижние точки (точки вращения) первого и второго листов совпали. Просмотрите фильм.

II. Оживите картинку, нарисованную в лабораторной работе №1, с помощью покадровой анимации.

Контрольные вопросы

1. Что такое ключевой кадр?
2. Как вставить ключевой кадр на временную шкалу?
3. В чем особенность покадровой анимации?
4. Что является основным недостатком покадровой анимации? (См. электронный курс лекций по дисциплине «Анимационная графика», раздел 4.3, покадровая анимация).

Лабораторная работа №3. Анимация формы

Цель работы: получить навыки создания анимации формы, а также использования контрольных точек.

Теоретические сведения

Форма — это контур, заливка или несколько контуров и заливок. Анимация формы позволяет

- плавно преобразовать одну **форму** в другую (*морфинг*);
- изменить **цвет** формы;
- **переместить** форму.

При анимации формы в начальном ключевом кадре рисуется исходная форма, а в конечном ключевом кадре эта форма изменяется, либо рисуется новая форма. Flash затем генерирует формы для промежуточных кадров, создавая анимацию трансформации одной формы в другую. К анимации относятся также параметры **Замедлить** и **Морфинг**. Обычно анимация выполняется с постоянной скоростью. Если надо постепенно замедлять анимацию, используют положительные значения параметра **Замедлить** (от 1 до 100), для ускорения — отрицательные (от -1 до -100). Параметр **Морфинг** имеет два возможных значения: **Дистрибутивный** (плавные контуры) и **Угловой** (сохранение углов).

Для управления более сложными или неожиданными изменениями формы можно использовать контрольные точки. Контрольные точки определяют точки, которые должны соответствовать в начальной и конечной форме. Контрольные точки создаются на первом кадре анимации и устанавливаются мышью в характерные точки формы. Если после этого перейти на последний кадр интервала анимации, мы увидим те же точки, которые можно перетащить мышкой в нужные места.

Контрольные точки обозначаются латинскими буквами (*a, b, c, ...*). Точка *a* на первом кадре «приходит» туда, куда установлена точка *a* на последнем кадре.

Контрольные точки бывают трех цветов:

- **красный** — контрольная точка создана, но еще не перемещалась;
- **желтый** — контрольная точка на первом кадре анимации, установленная на контуре фигуры;

- **зеленый** — установленная контрольная точка на последнем кадре анимации.


В программе предусмотрены следующие операции с контрольными точками:

- **добавить** точку — нажать клавиши **Ctrl+Shift+N** или выбрать пункт меню **Модификация—Фигура—Добавить хинт кривой**;
- **удалить** точку — щелкнуть правой кнопкой мыши на точке и выбрать пункт **Удалить хинт** из контекстного меню; удалить все точки можно с помощью пункта меню **Модификация — Фигура — Удалить все хинты**;
- **показать** (скрыть) контрольные точки можно с помощью пункта **Просмотр — Показать хинты кривых** главного меню.



Анимация формы — сложный процесс, который иногда дает сбой. Например, при установке нескольких контрольных точек изображение в промежуточных кадрах может вообще исчезнуть. Поэтому рекомендуется использовать наименьшее количество точек и расставлять их в порядке обхода контура против часовой стрелки.

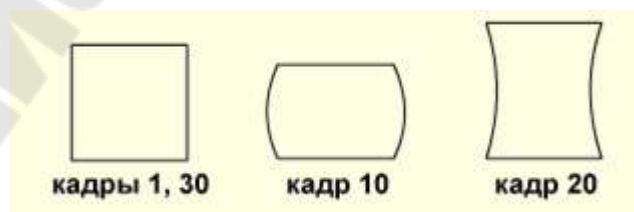
Порядок выполнения работы

I. Создание анимации формы.

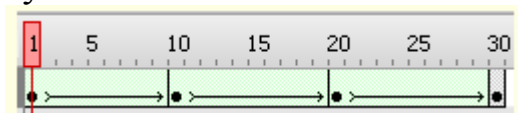
1. Выберите инструмент **Прямоугольник**, установите черный цвет контура, без заливки. Обязательно **отключите режим рисования объектов** (кнопка  в нижней части панели инструментов не должна быть нажата).

2. Нарисуйте в центре поля квадрат размером примерно 300 на 300. Вставьте новые ключевые кадры в кадры 10, 20 и 30 (клавиша **F6**).

3. Используя инструмент  **Стрелка** и  **Свободное преобразование**, измените контуры в кадрах 10 и 20 так, как показано на рисунках ниже. Кадр 30 оставьте без изменений (он совпадает с кадром 1).



4. Теперь мы создадим плавные переходы между этими формами, причем рисовать вручную ничего не придется. Чтобы включить анимацию формы для первого перехода, щелкните правой кнопкой мыши по кадру 10 и выберите из контекстного меню пункт **Создать анимацию формы**. Повторите то же самое для кадров 10 и 20. Если вы сделали все правильно, промежуточные кадры имеют зеленоватый цвет и через них проходит черная стрелка от начального кадра к конечному:



5. Проверьте ролик, нажав клавиши **Ctrl+Enter**.

II. Установка контрольных точек.

1. Создайте новый документ, вытащите на экран линейки, выбрав пункт меню **Просмотр — Линейки**. Установите две горизонтальных и две вертикальных направляющих, так чтобы они ограничили квадрат размером примерно 300 на 300.

2. Нарисуйте синий круг без границы, вписав его в квадрат.

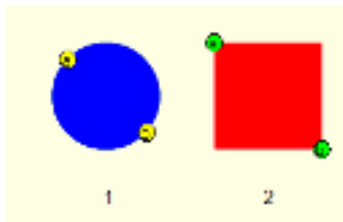
3. Добавьте пустой ключевой кадр в кадр 10 (клавиша **F7**).

4. Нарисуйте красный квадрат без контура, который выровнен по направляющим.

5. Добавьте анимацию формы к кадру 1.

6. Просмотрите результат.

7. Когда круг переходит в квадрат, видно некоторое вращение. Чтобы сделать анимацию более красивой, мы поможем программе, установив **контрольные точки**. Перейдите на кадр 1 и установите две контрольные точки так, как показано на рисунке 1. Затем перейдите на кадр 10 и переместите контрольные точки в противоположные углы квадрата (рисунок 2). Просмотрите результат.



III. С помощью анимации формы

- превратить круг в квадрат;
- превратить 2 в 1;
- превратить круг в 1;

- d) превратить 2 в квадрат;
- e) превратить яблоко в грушу;
- f) превратить «Слон» в «Цирк».

IV. Выполнить пункт III, используя контрольные точки.

Контрольные вопросы

1. Что такое форма?
2. Какую анимацию позволяет создавать анимация формы?
3. Что такое контрольные точки?
4. Для чего нужны контрольные точки?
5. Какое правило следует помнить при использовании контрольных точек? (См. электронный курс лекций по дисциплине «Анимационная графика», раздел 4.6, анимация формы).

Лабораторная работа №4. Анимация движения

Цель работы: получить основные навыки создания анимации движения.

Теоретические сведения

Анимации движения применяется в тех случаях, когда анимация состоит из непрерывного движения или преобразования одного экземпляра символа.

Элементы этой анимации позволяют задать свойства экземпляра символа, такие как

- изменение положения символа;
- преобразование символа (масштабирование, поворот, скос);
- применение цветовых эффектов;
- различные комбинации вышеназванных преобразований.

При анимации движения используется один ключевой кадр, с расположенным на нем экземпляром символа (целевым объектом). Для фиксации его движений и преобразований используются ключевые кадры свойств. Flash затем генерирует значения свойств экземпляра для промежуточных кадров.

Классическая анимация движения анимация похожа на анимацию движения, но более сложна в создании. Если анимация движения может быть связана только с одним экземпляром символа, то классическая анимация позволяет выполнять движение и преобразование нескольких экземпляров.

При классической анимации движения используются начальный и конечный ключевые кадры, в которых появляются новые экземпляры символов и определяются их свойства, промежуточные кадры генерируются путем преобразований экземпляров символов.


Классическая анимация движения позволяет создавать некоторые анимированные эффекты, которых невозможно добиться с использованием анимации движения.

Порядок выполнения работы


I. Создание анимации движения по траектории.

1. Создайте новый документ с размером сцены 550 на 400 пикселей. Сохраните его под именем `ball fla`.


2. Переименуйте слой *Слой 1* в *Фон* и добавьте на него фоновое изображение `back.jpg` (используйте команду меню **Файл — Импорт — Импортировать в рабочую область**). Чтобы защитить этот слой от случайных изменений, щелкните по черной точке ниже символа напротив названия слоя.

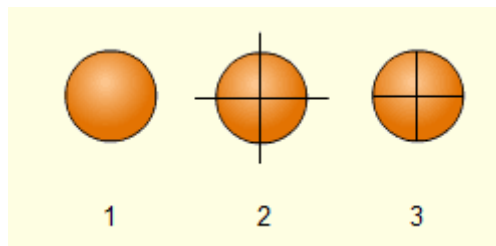
3. Создайте новый слой с именем *Мяч*. Чтобы фон не мешал нам, отключите видимость слоя *Фон*, щелкнув на точке под значком  напротив его названия.

4. Теперь будем рисовать баскетбольный мяч. Выберите инструмент **Овал** и отключите режим **Рисование объектов**. Установите черный контур толщиной 1 пиксель и оранжевый цвет заливки. Нарисуйте круг (мяч), удерживая клавишу **Shift**.

5. Выделите заливку и замените ее (с помощью панели **Цвет**) на **Радиальный градиент** от светло-оранжевого к темно-оранжевому. С помощью инструмента  (**Преобразование градиента**, клавиша **F**) переместите центр градиента немного выше и левее центра мяча (рисунок 1).

6. Проведите вертикальную и горизонтальную линии через центр мяча (рисунок 2).

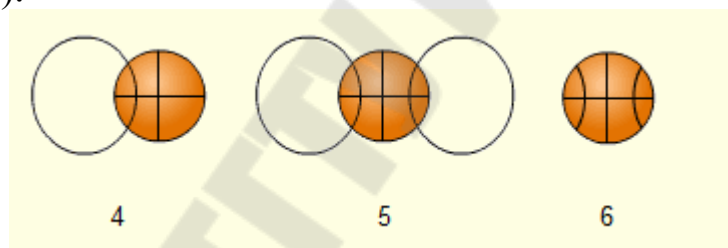
7. При нажатой клавише **Ctrl** (которая временно включает инструмент ) выделите «хвостики», выступающие за мяч, и удалите их (рисунок 3).



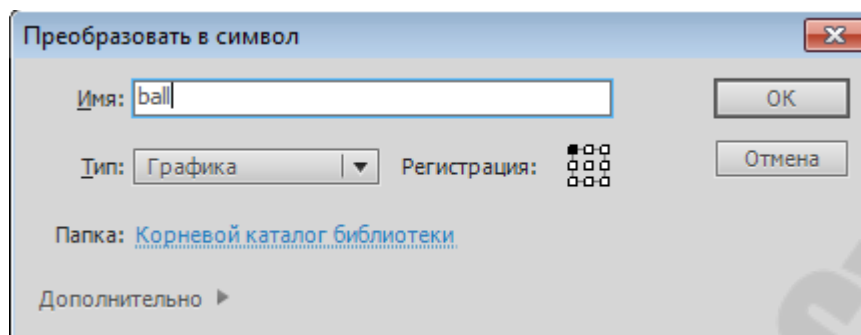
8. Включите снова инструмент **Овал** и режим **Рисование объектов**. Отключите заливку и нарисуйте овал (рисунок 4).

9. Включив инструмент **Стрелка**, перетащите копию этого овал вправо (при нажатой клавише **Alt**). Выровняйте оба овала (рисунок 5).

10. Преобразуйте оба овала в простые контуры с помощью команды **Распределить по слоям** из контекстного меню (клавиши **Ctrl+B**). Затем выделите и удалите все части, выступающие за контур мяча (рисунок 6).



11. Преобразуйте нарисованный мяч в символ: выделите его инструментом **Стрелка** и нажмите клавишу **F8** (меню **Модификация—Преобразовать в символ**). На экране появится окно, в котором нужно ввести название символа (мы введем *ball*), выбрать тип символа **Графика** и определить, где находится так называемая **точка регистрации (Регистрация)**. Эта точка, которая определяет координаты символа на экране (они используются при управлении объектом из программы). По умолчанию точка регистрации располагается в левом верхнем углу, но это можно изменить как при создании символа (щелкнув мышкой по другому квадратику в поле **Регистрация**), так и потом при редактировании.



Теперь символ появился в библиотеке и его можно использовать много раз, не опасаясь увеличения объема файла. На сцене новый символ обведен голубой рамкой, крестик показывает точку регистрации. Для создания символа «с нуля» можно использовать и другой способ: нажать клавиши **Ctrl+F8** (меню **Вставка—Создать символ**). После этого в библиотеке создается пустой символ и сразу открывается в режиме редактирования.

12. Войдите в режим редактирования символа и сделайте так, чтобы точка регистрации оказалась в центре мяча. Если символ уже есть на сцене, для перехода в режим редактирования можно сделать на нем двойной щелчок. Это так называемое **редактирование на месте**. Помните, что все сделанные изменения применяются ко всем копиям символа, которые есть в ролике.

Форму символа, расположенного на сцене, можно изменять с помощью инструмента **Свободное преобразование**. Эти изменения влияют только на тот символ, с которым вы работаете. Отрегулируйте размер мяча так, чтобы он соответствовал размеру площадки.

13. Сделайте видимым фоновый слой и расположите мяч в верхней части поля.

14. В слое *Мяч* вставьте два ключевых кадра в кадры 10 и 20 (клавиша **F6**). Заметим, что при этом фоновое изображение в кадрах 2-20 исчезло, поскольку рабочая часть временной шкалы для слоя *Фон* содержит только 1 кадр. Выберите кадр 20 слоя *Фон* и добавьте в него новый простой кадр (клавиша **F5**).

15. Перейдите в кадр 10 слоя *Мяч*, выделите мяч и передвиньте его вертикально вниз почти до границы сцены. Чтобы случайно не сдвинуть его влево или вправо, лучше использовать для перемещения стрелку «вниз». При нажатой клавише **Shift** скорость перемещения увеличивается. Теперь программа сможет автоматически построить все промежуточные кадры анимации между

кадрами 1—10 и 10—20. Это **анимация движения**, поскольку форма объекта не меняется.

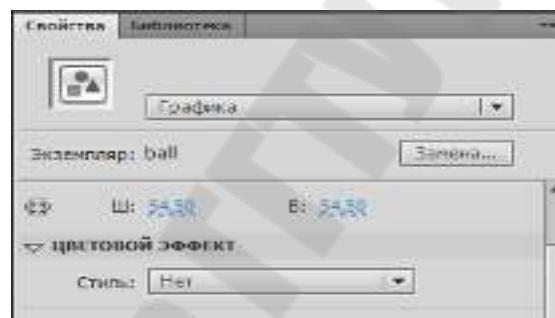
16. Щелкните правой кнопкой мыши по кадру 1 слоя *Мяч* и выберите пункт меню **Создать классическую анимацию движения**. Сделайте то же самое для кадра 10. Просмотрите ролик.

Видим, что мяч летит неестественно — с постоянной скоростью и как будто отталкивается от верхней точки.

По законам физики, мяч летит вниз ускоренно, а вверх — замедленно. В программе *Flash* это можно сделать с помощью параметра на панели **Свойства**. Кнопка справа от поля **Замедлить** позволяет точно настраивать кривую изменения скорости анимации.

17. Перейдите в кадр 1 слоя **Мяч** и установите значение **Замедлить** равное **-100**. В кадре 10 установите этот параметр равным **100**. Просмотрите результат.

Теперь посмотрим, какие свойства символа можно менять. Если выделить мяч и посмотреть на панель **Свойства**, мы увидим, что выделенный объект — копия символа *ball* из библиотеки.



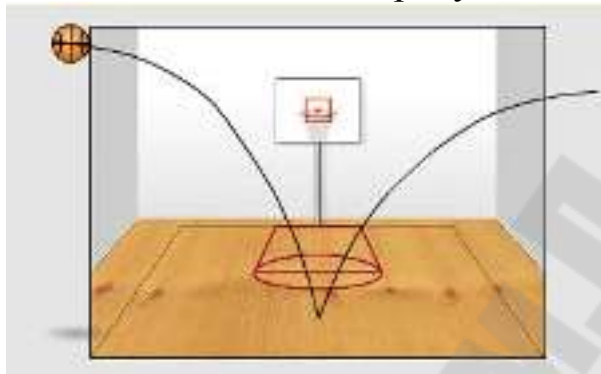
Список **Стиль** позволяет изменять цвет данного экземпляра (эти изменения не отражаются на символе в библиотеке и на других копиях):

- **Нет**;
- **Яркость**;
- **Тон**;
- **Альфа-канал** - прозрачность;
- **Дополнительно**.

18. В кадре 10 выделите мяч, выберите в списке **Стиль** вариант **Тон** и установите для смешивания зеленый цвет в соотношении 50%. Просмотрите ролик.

19. Теперь изменим ролик так, чтобы мяч вылетал слева, ударялся о пол и улетал вправо за экран. В программе есть возможность задать нестандартную траекторию полета с помощью слоя направляющих. Выделите слой *Мяч* и щелкните **Добавить направляющую классической анимации** в контекстном меню.

20. Перейдите в кадр 1 слоя *Направляющая: Мяч*. Нарисуйте кривую примерно так, как показано на рисунке ниже.



Обратите внимание, что мяч в ключевом кадре сам перескакивает на направляющую (на ней должен находиться центр вращения).

21. В 10 кадре установите мяч так, чтобы центр вращения совпал с самой нижней точкой траектории.

22. В кадре 20 также переместите мяч на траекторию. Просмотрите ролик.

Реальный баскетбольный мяч всегда вращается в полете. В программе *Flash* есть возможность добавить вращение объекта на целое число оборотов на каждом интервале анимации движения. На панели **Свойства** есть список **Поворот**, в котором можно выбрать

- **Нет** — нет вращения;
- **Автоматически** — строится автоматически, учитывая угол поворота в начальном и конечном положениях;
- **По часовой стрелке**;
- **Против часовой стрелки**.

Также рядом с этим списком можно выбрать количество оборотов.

23. Выделите кадр 1 слоя *Мяч* и добавьте к анимации вращение — 2 оборота по часовой стрелке. То же самое сделайте для кадра 14. Чтобы лучше увидеть результат, уменьшите частоту кадров до 5. Просмотрите ролик.

II. Используя анимацию движения по траектории, нарисовать

- a) пчелу, летящую к сотам;
- b) бабочку, летящую к цветку;
- c) букашку, ползающую по листу;
- d) грузовик, движущийся по дороге;
- e) самолет, летящий по небу.

Контрольные вопросы

1. Какие виды анимации движения во Flash вы знаете?
2. Чем отличается анимация движения от классической анимации движения?
3. Как создать анимацию движения по траектории?
4. Чем отличается ключевой кадр от обычного?
5. Как ускорить (замедлить) движение объекта?
6. Как осуществить поворот объекта при анимации? (См. электронный курс лекций по дисциплине «Анимационная графика», раздел 4.4, анимация движения и раздел 4.5, классическая анимация движения).

Лабораторная работа №5. Текст

Цель работы: получить навыки работы с текстом, научиться создавать различные текстовые эффекты.

Теоретические сведения

Классический текст — это один из механизмов визуализации текста во Flash. Выбрать этот механизм можно в Инспекторе свойств текста.

Классический текст можно включить в приложения Flash различными способами. Можно создать текстовые поля, содержащие *статический* текст, *динамический* текст и текстовые поля *ввода*.

Программа Flash реализует множество способов работы с классическим текстом. Она позволяет ориентировать текст горизонтально или вертикально, задавать его атрибуты (шрифт, размер, стиль, цвет, межстрочный интервал и т.д.), проверять орфографию, выполнять преобразование текста (поворот, наклон и зеркальное отображение), анимировать, выделять, связывать с

объектами, управлять подстановкой шрифтов и использовать шрифты из общих библиотек.

Можно создать поля классического текста трех типов: статические, динамические и вводимые.

- В статических текстовых полях отображается текст, содержимое которого динамически не изменяется.

- В динамических текстовых полях отображается динамически обновляемый текст, например биржевые котировки, сводки погоды и т. п.

- Текстовые поля ввода предназначены для того, чтобы пользователи вводили текст в формах или опросах.

При создании статического текста поле может быть размещено в одной строке, расширяющейся по мере ввода, либо являться полем фиксированной ширины (для горизонтального текста) или фиксированной высоты (для вертикального текста), которое автоматически расширяется с переносом текста на следующую строку. При создании динамического или вводимого поля текст может быть размещен как в одной строке, так и иметь фиксированную ширину и высоту.

Порядок выполнения работы

I. Создание текстового эффекта тающих букв.

1. Создайте новый документ размером 500 на 200 пикселей и сохраните его.

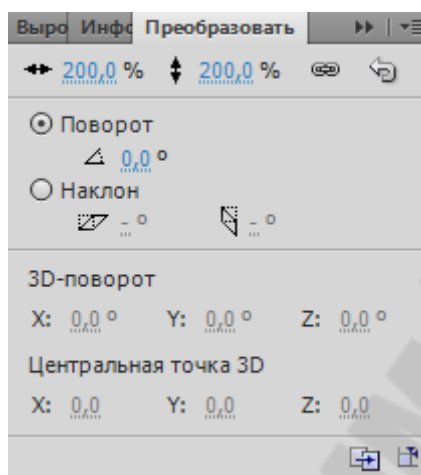
2. Нарисуйте прямоугольник без рамки, закрывающий всю сцену и залейте его градиентом (снизу — черный, сверху — темно-синий).

3. Переименуйте *Слой 1* в *Фон* и заблокируйте его.

4. Вставьте новый слой *Буквы* выше фона. С помощью инструмента **Текст** введите на этом слое в центре сцены текст *УРА* белого цвета размером 120 (этот размер нельзя установить с помощью движка, но можно ввести с клавиатуры) и расположите текст по центру.

5. Разбейте текст по команде **Модификация — Разделить** и примените команду **Модификация — Временная шкала — Распределить по слоям**, после чего каждая буква слова будет помещена в отдельный слой, при этом каждый слой будет назван соответствующей буквой.

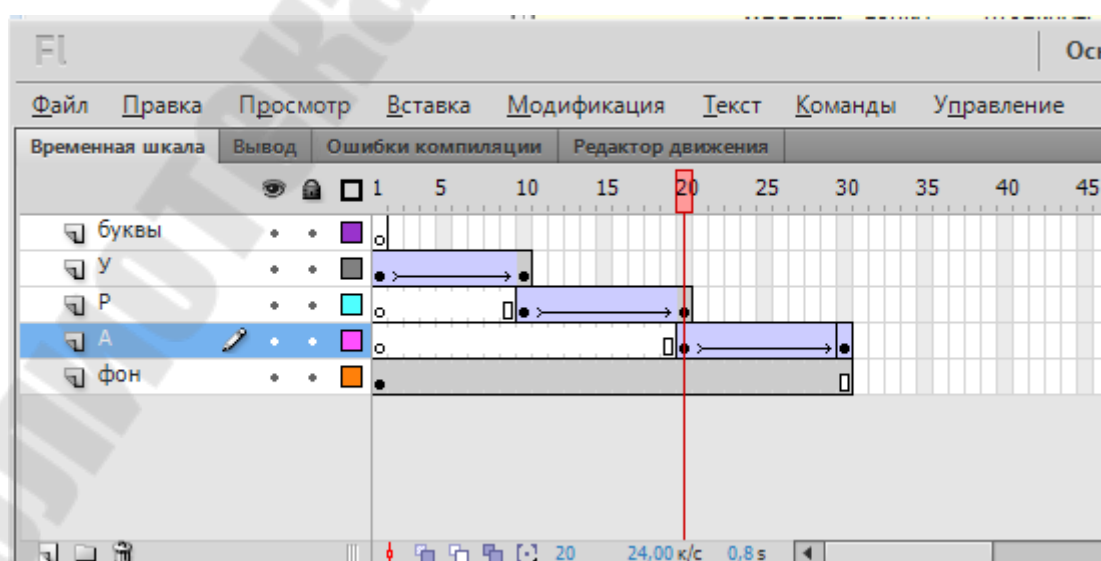
6. Перейдите в слой с первой буквой, в сделаем 10 кадр ключевым, перейдем в первый кадр и создадим **Классическую анимацию движения**. В 10 кадре увеличьте масштаб буквы до 200% — для этого вызовем панель **Преобразование** по команде **Окно — Преобразование** и введите нужное значение масштаба в процентах.



7. Выделите экземпляр символа в 1-м кадре и в панели **Цвет** увеличьте прозрачность экземпляра символа. При установке параметра **Alpha** на уровне 0% буква станет прозрачной.

8. Переместите первый фрейм в слое «р» в 10-й кадр (для этого необходимо щелкнуть по нему мышью и потом в режиме drag&drop перетащить в нужное место) и повторите ту же процедуру, что и с буквой «У». Аналогично поступите со всеми буквами.

9. В слое *Фон* в 30 кадре вставьте обычный кадр. Текстовый эффект готов.



II. Создание текстового эффекта появления букв.

1. Создайте новый документ, установите размеры поля 400 на 300 пикселей. Загрузите фоновый рисунок `dawn.jpg` (**Файл — Импортировать — Импортировать в рабочую область**) и заблокируйте слой. Сохраните файл под именем `valaam fla`.

2. Создайте новый слой *Текст*, введите в нижней части сцены текст *Валаам* (шрифт *Arial*, черный, жирный, размер 70) и выровняйте его по середине сцены (панель **Выравнивание Ctrl+K**).

3. Разбейте текст на отдельные буквы и разместите каждую из них на отдельном слое. Выделите текст и примените команды **Разделить** и **Распределить по слоям** из контекстного меню. Удалите слой *Текст* (он теперь пустой).

Программа создала несколько новых слоев, их имена совпадают с буквами текста.

4. Выделите кадр 10 во всех слоях с буквами и добавьте новый ключевой кадр (**F6**).

5. Затем выделите кадр 1 во всех слоях-буквах (выделятся все буквы) и клавишей «вправо» переместите буквы за правую границу сцены (нажатие клавиши **Shift** ускоряет процесс).

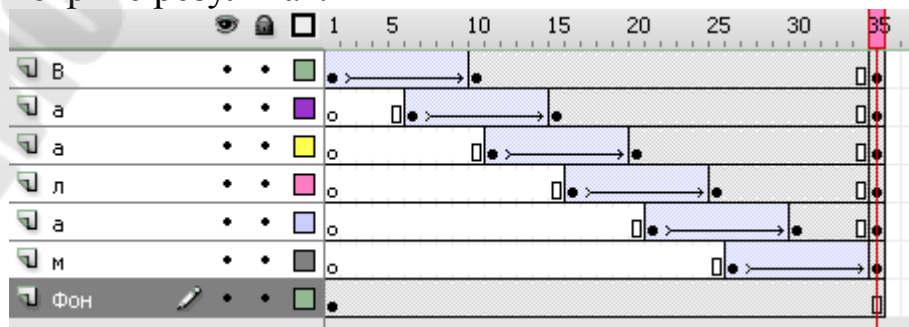
6. Включите анимацию движения для всех слоев-букв, вставьте новый промежуточный кадр в кадр 10 слоя *Фон* и просмотрите результат.

Если посмотреть на библиотеку, можно заметить, что в нее добавлены новые символы, созданные автоматически из букв. Это необходимо программе для построения анимации движения.

7. Выделите кадры 1-10 слоя, где находится первая буква *а*, и перетащите вправо, так чтобы анимация начиналась с кадра 6.

8. Сдвиньте и другие слои с буквами так, как показано на рисунке.

9. Выделите кадр 35 всех слоев-букв и вставьте новый ключевой кадр, в тот же кадр фонового слоя вставьте промежуточный кадр. Просмотрите результат.



10. Далее, используя фильтр **Тень**, сделаем так, чтобы буквы постепенно становились золотистыми. Однако применить фильтры можно только к тексту или символам типа **Фрагмент ролика**. Сейчас буквы имеют тип **Графика**, это можно увидеть на панели **Свойства**. Поэтому измените тип символов на **Фрагмент ролика**.

11. Сейчас все готово для создания анимации, которая сводится к постепенному применению фильтра. Выделите кадр 50 во всех слоях-буквах и вставьте новый ключевой кадр. Выделяя каждую букву, на панели свойства перейдите на панель **Фильтры** и примените фильтр **Тень**.

12. Добавьте анимацию движения ко всем слоям с буквами в кадре 35.

13. Остается сделать небольшую паузу перед началом нового цикла. Вставьте промежуточные кадры в кадр 70 во всех слоях и просмотрите результат. Сохраните файл.

III. Используя пункт I, выполнить рассмотренный эффект для своего имени.

IV. Используя пункт II, выполнить рассмотренный эффект для своей фамилии.

Контрольные вопросы

1. Что позволяет делать инструмент Текст?
2. Какие свойства имеет инструмент Текст?
3. Как производится форматирование текста во Flash?
4. Чтобы применить к тексту эффект, что необходимо сделать?
5. Можно ли применять к тексту анимацию формы, анимацию движения? (См. электронный курс лекций по дисциплине «Анимационная графика», раздел 5, текст).

Лабораторная работа №6. Маска


Цель работы: познакомиться с понятием маски во Flash и научиться использовать ее для создания различных эффектов анимации.


Теоретические сведения

Слой-маска — это такой слой, который скрывает изображения на расположенном под ним (маскируемом) слое, но с помощью маскирующего элемента создается прокол, через который видны нижележащие слои. Маскирующий элемент может быть заполненной фигурой, текстовым объектом, экземпляром графического символа или фрагмента ролика.

Т.е. если на слое-маске поместить объект с заливкой, то через него, как через окно, будет просматриваться участок маскируемого слоя.



Другими словами, заливка в маске вырезает часть рисунка, а все остальное делает прозрачным. Причем цвет заливки в маске не имеет значения, он может быть даже полупрозрачным. Более того, в режиме объединения (когда отключена кнопка  в нижней части панели инструментов) на маске можно рисовать разными цветами, все области будут выделять части изображения.

Если же работать в режиме рисования объектов (кнопка  нажата), маской становится только самый нижний объект.

Сгруппировав несколько слоев под одним маскирующим слоем, можно создать сложные эффекты.

Чтобы создать динамические эффекты, можно анимировать маскирующий слой. Для заполненной формы, используемой в качестве маски, используется анимация формы, для текстового объекта, графического экземпляра или фрагмента ролика используется анимация движения. При использовании фрагмента ролика в качестве маски анимируется маска вдоль контура движения.

Порядок выполнения работы

I. Разноцветный текст.

Вообразим себе такую ситуацию: кто-то взял несколько различных красок и смешал на листе бумаги. В другом листе прорезал ряд отверстий по форме букв текста и положил этот лист поверх первого. То, что будет после этого видно в вырезе, — и есть результат, который мы с вами сейчас попробуем получить во Flash.

Для этого:

1. С помощью кисточки создайте разноцветное пятно.
2. Создайте новый слой над существующим (выбрать **Вставить слой** из контекстного меню **Слой**) и разместите на нем надпись с помощью инструмента **Текст** «Привет» так, чтобы она была поверх пятна.
3. Превратите верхний слой в маску, выбрав из контекстного меню слоя **Маска**. Эффект готов.



II. Текст, наблюдаемый сквозь движущееся отверстие.

Представьте себе такую ситуацию. На листе написан какой-то текст. Во втором листе прорезано круглое отверстие, и этот лист, не спеша, перемещается над первым листом. Текст, изображенный на нижнем листе, просматривается сквозь отверстие во втором. Вот такую ситуацию и модулируем во Flash.

Анимацию перемещения отверстия будем размещать на первых двадцати кадрах. Для ее создания понадобится два слоя. На нижнем слое должна находиться надпись, а на верхнем — движущееся отверстие, сквозь которое будет проводиться наблюдение нижнего слоя.

А теперь по шагам:

1. Создайте текстовую надпись «Привет всем».
2. Продлите слой до 20-го кадра (щелчок левой клавишей мыши по 20-му кадру и нажать на клавишу **F5**).
3. Создайте новый слой над существующим.
4. На созданном слое нарисуйте круг с заливкой (все равно, какого цвета),двигающийся от начала надписи к ее концу. Движение должно занимать 20 кадров.
5. Превратите верхний слой в маску. Эффект готов.

III. Надпись, освещаемая прожекторами.

Для создания такого эффекта понадобится три слоя. На нижнем слое будет неосвещенный вариант надписи. На слое выше — освещенный вариант. На верхнем слое будет маска, сквозь которую должна просматриваться освещенная надпись.

1. Установите серый цвет сцены, выбрав из контекстного меню **Свойства документа** и изменив фон на серый.
2. Создайте текстовую надпись «Привет всем!!!» черного цвета.
3. Продлите слой до 20-го кадра (щелкнуть по нужному кадру и нажать клавишу **F5**).
4. Создайте новый слой над существующим, он автоматически должен продлиться до 20-го кадра.
5. На новом слое нарисуйте светло-серый прямоугольник (светлее чем сцена), который полностью, с запасом закрывает надпись, и на нем напишите «Привет всем!!!», но буквами белого цвета. Надпись должна располагаться точно над предыдущей. Такой точности можно добиться, скопировав нижнюю надпись и вставив ее с помощью команд **Правка — Вставить на место**.
6. Создайте третий слой над существующими.
7. На новом слое нарисуйте заполненный круг,двигающийся от начала надписи к ее концу. Движение должно занимать двадцать кадров.
8. Превратите самый верхний слой в маску, а последний слой в маскируемый с помощью контекстного меню **Свойства— Маскируемый**. Просмотреть созданный эффект (**Ctrl+Enter**).

IV. Двухтоновое инвертирование надписи.

Для создания подобного эффекта нужно иметь два варианта изображения. Один из другого должен получаться инвертированием. С помощью маски нужно прорезать отверстие в верхнем изображении, чтобы из под него было видно нижнее. Вот и весь секрет фокуса.

1. Создайте надпись «Привет» черного цвета.
2. На новом слое закройте сцену черным прямоугольником и поместите надпись «Привет» белого цвета на том же месте, где старая.
3. Добавьте третий слой. На нем нарисуйте фигуру, которая будет выступать в качестве маски. Нарисованная фигура должна закрывать часть надписи.

4. Превратить последний слой в маску. Эффект готов.

V. Маскирование фотографии.

Иногда интересный эффект можно получить применяя маски к фотографиям.

Для создания подобного эффекта понадобится три слоя. На нижнем нужно будет разместить картинку, на среднем — маску в виде надписи, на верхнем — ту же надпись с тенью и рельефом.

1. На нижний слой импортируйте фотографию.
2. На новом слое поместите надпись «Привет!!!».
3. Создайте еще один слой и на него скопируйте уже существующий текст с предыдущего слоя так, чтобы буквы располагались точно над оригиналом.
4. Выделите скопированный текст и примените к нему два фильтра: **Тень** и **Фаска** с настройками по умолчанию.
5. В настройках фильтра **Фаска** отметьте флажок **Удаление фона**, чтобы сами буквы скрыть и оставить только эффекты применения фильтров.
6. Превратите средний слой в маску.
7. Просмотрите результат.

VI. Создать анимацию движения лупы над панорамой города.

1. В первый кадр слоя “вид” импортируйте изображение панорамы города **minsk1.jpg**. и преобразуйте в символ. С помощью панели **Цвет** установите прозрачность $\alpha = 40\%$.

2. Выше добавьте слой с именем “увеличение”. Из библиотеки поместите на него изображение панорамы и увеличьте его.

3. Еще выше добавьте третий слой и назовите его “лупа”. Нарисуйте на нем круг с заливкой и преобразуйте его в символ с именем “лупа”.

4. В этом слое создайте анимацию движения лупы из левого нижнего угла (1-й кадр) по диагонали в правый верхний (24 кадр).

5. Щелкните правой клавишей мыши по слою “лупа” и установите тип **Маска**. Через него будет просматриваться участок маскируемого слоя с увеличенным изображением.



6. Сохраним работу под именем **panorama.fla**. Протестируем анимацию.

VII. Создать анимацию, в которой сверху вниз движутся вращающиеся увеличивающиеся прямоугольники. Внутри прямоугольников просматривается фотография. В конце анимации просматривается фотография целиком.

1. Создайте новое изображение.
2. Импортируйте растровое изображение природа.jpg. Дайте слою имя Природа.
3. Создайте новый слой, дайте имя Маска.
4. На слое Маска создайте изображение прямоугольника, вытянутого по высоте. Сгруппируйте прямоугольник.
5. Создайте на основе прямоугольника символ типа **Фрагмент ролика** и назовите Прямоугольник.
6. Настройте для символа анимацию, которая должна изменять положение, размер и вращать его. Для этого:
 - войдите в режим редактирования символов, дважды щелкнув кнопкой мыши по символу из библиотеки;
 - выделите 20 кадр, сделайте его ключевым (F6);
 - переместите прямоугольник за нижнюю границу кадра, увеличьте размер прямоугольника примерно в два раза;
 - создайте анимацию движения с 1 по 20 кадр с поворотом по часовой стрелке;
 - выйдите из режима редактирования группы, нажав на **Монтажный кадр**.
7. На слой Маска поместите несколько символов Прямоугольник. Объедините их в группу.
8. Создайте на основе этой группы символ Прямоугольники.
9. Измените тип слоя на слой-маска.

10. Для слоя Маска сделайте 20-й кадр промежуточным. Для слоя Природа сделайте промежуточным 34 кадр (F5).
11. Посмотрите полученную анимацию в окне проигрывателя.
12. Сохраните файл.

Контрольные вопросы

1. Что такое слой-маска?
2. Для чего используется слой-маска?
3. Как создать слой-маску во Flash?
4. Приведите примеры использования слой-маски.
5. Назовите фильтры, используемые во Flash. Какие эффекты они позволяют получить?
6. Какие существуют режимы наложения во Flash? Для чего они используются? (См. электронный курс лекций по дисциплине «Анимационная графика», раздел 4.8, маскирующий слой и раздел 4.9, фильтры и режимы наложения).

Лабораторная работа №7. Интерактивность во Flash

Цель работы: получить навыки создания анимации, управления анимацией с помощью простейшего программного кода.

Теоретические сведения

В простом анимированном фильме Flash воспроизводит кадры временной шкалы в определенной последовательности, которая остается неизменной, сколько бы раз вы не просматривали фильм. В интерактивном фильме зритель имеет возможность использовать клавиатуру, мышь или то и другое, чтобы перейти к некоторому фрагменту или сцене фильма, переместить объекты, ввести информацию, а также выполнить многие другие интерактивные операции.

Интерактивность Flash-фильма обеспечивается за счет включения в него так называемых сценариев, которые представляют собой набор инструкций на языке ActionScript. Каждая инструкция инициируется при наступлении определенного связанного с ней события. События могут быть самые разнообразные: достижение

считывающей головкой определенного кадра, нажатие пользователем клавиши на клавиатуре, щелчок кнопкой мыши и т.д.

Использовать ActionScript в среде разработчика можно с помощью панели «Действия», в окне «Сценарий». Панель «Действия» содержат полнофункциональный редактор программного кода, включающий подсказки кода и выделение цветом, форматирование, цветовую подсветку и проверку синтаксиса, отладку, номера строк, перенос слов и поддержку Юникода.

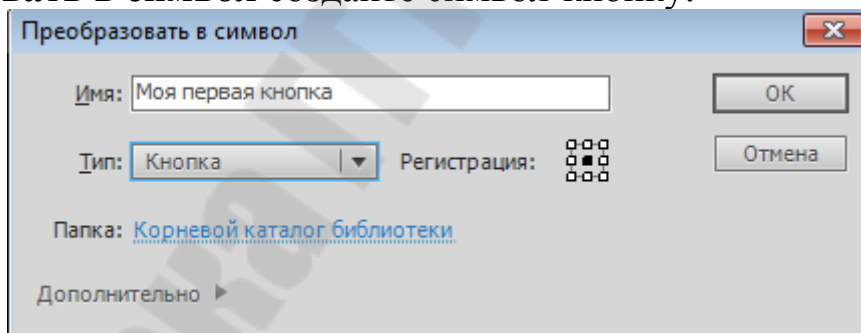
Порядок выполнения работы

I. Создание символа-кнопки.

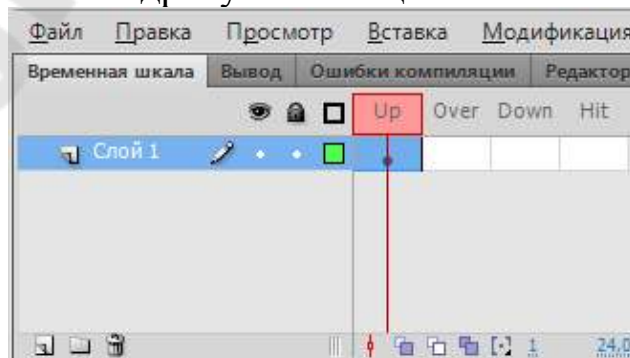
1. Создайте заготовку для будущей кнопки, например:



2. Выделите заготовку кнопки и командой **Модификация—Преобразовать в символ** создайте символ-кнопку:



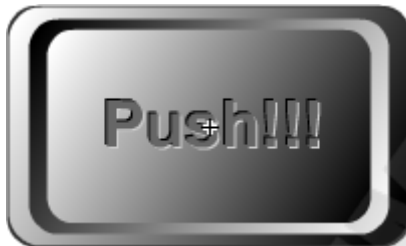
3. Войдите в режим редактирования символа, щелкнув на нём дважды. Обратите внимание на то, что кнопка во Flash представляет собой 4-кадровую анимацию:



4. В кадре **Up** изобразите кнопку в отжатом (нормальном) состоянии, например:



5. В кадре **Over** изобразите кнопку в том виде, который она будет принимать при наведении на неё указателя мыши. Например на кнопке будет появляться надпись Push!!!:



6. В кадре **Down** изобразите кнопку состояния, когда на неё нажали, например:



7. В кадре **Hit** с помощью заливки создайте активную область. Активная область – это пространство, щелчок внутри которого воспринимается Flash как щелчок на кнопке:



8. Готовое изображение сохраните в своей папке в формате *.fla. Если результат не устраивает, внесите изменения в соответствующие ключевые кадры.

9. Просмотрите готовый фильм, нажав сочетание клавиш **Ctrl+Enter**.

II. Управление воспроизведением фильма.

1. Откройте фильм, созданный в одной из предыдущих работ.

2. Добавьте в фильм новый слой, который будет использоваться в качестве дополнительного фона; именно в фоновый слой следует добавить кнопки, чтобы они присутствовали на сцене на всем протяжении фильма.

3. Щелкните мышью в ячейке первого ключевого кадра в «кнопочном» слое.

4. Перетащите поочередно из окна библиотеки на слой две кнопки, которые вы решили использовать в фильме (**Окно – Общие библиотеки – Кнопки**);



5. Выберите первую из кнопок, с помощью которой фильм должен приостанавливаться.



6. Выберите **Окно – Действия** (клавиша **F9**) и введите текст в окно кода

```
on (release)
{
stop();
}
```

On (release) обеспечивает инициализацию действия stop() при наступлении события release (то есть при щелчке на кнопке); пустые

скобки после имени процедуры означают, что данная процедура используется без параметров.

7. Не закрывая палитру Действия(Actions), выберите вторую кнопку



и назначьте ей действие `play()`, повторив описанную выше процедуру:

```
on (release)
{
play();
}
```

8. Протестируйте фильм, нажав сочетание клавиш **Ctrl+Enter**.

III. Адреса: родители, дети и корень.

1. Откройте файл `car.flx`.
2. Создайте два новых слоя: *Кнопка* и *Программа*.
3. На слой *Кнопка* поместите кнопку из библиотеки и просмотрите ролик.



Обратите внимание, что *Кнопка* — это символ типа **Кнопка**, а *Машина*, так же как и два ее колеса — символы типа **Фрагмент ролика**.

4. В первый кадр слоя *Программа* добавьте код `stop ()`

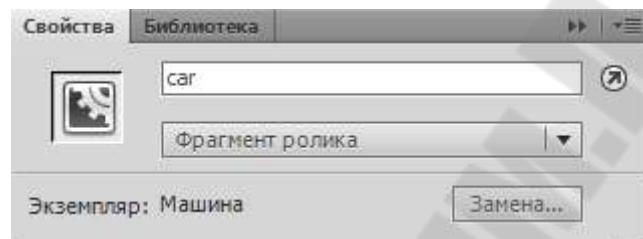
Если теперь просмотреть ролик, выяснится, что машина стоит на месте, но ее колеса вращаются. Это происходит потому, что команда `stop()` остановила только проигрывание главной

временной шкалы, а внутренние клипы (колеса) продолжают работать.

Чтобы ответить на вопрос «Как остановить колеса?» нужно разобраться с адресами внутри флэш-ролика. Но прежде всего, дадим имена объектам на сцене, к которым мы будем обращаться: машине и двум колесам внутри нее.

5. Выделите машину и на панели **Свойства** введите имя этого элемента `car`.

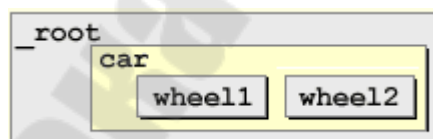
6. Откройте клип *Машина* для редактирования и таким же образом дайте колесам имена `wheel1` и `wheel2`.



Итак, в этом фильме 4 монтажных стола со своими временными шкалами:

- **главный** (корневой) монтажный стол, он называется `_root`;
- монтажный стол клипа **Машина**;
- два монтажных стола **для каждого колеса**.

Связи этих монтажных столов показаны на схеме:



Клип `car` находится внутри главного монтажного стола `_root`, а клипы `wheel1` и `wheel2` — внутри клипа `car`.

Главный монтажный стол `_root` является «родителем» (*parent*) для клипа `car`.

Клип `car` является «сыном» (*child*) для `_root` и родителем для клипов `wheel1` и `wheel2`.

Клипы `wheel1` и `wheel2` — сыновья для `car`, у них самих нет потомков.

Абсолютные адреса. Каждый объект имеет свой абсолютный адрес, по которому его можно найти из любого места. Для объектов

Flash-фильма абсолютный адрес начинается с `_root`, так что абсолютные адреса всех клипов, показанных на схеме, выглядят так:

```
_root
_root.car
_root.car.wheel1
_root.car.wheel2
```

Как видим, для обращения к сыновьям используется точка.

Абсолютные адреса не зависят от того, внутри какого монтажного стола они используются.

Чтобы остановить проигрывание для какого-то монтажного стола, надо перед командой `stop()` поставить его адрес и точку:

```
_root.stop();
_root.car.stop();
_root.car.wheel1.stop();
_root.car.wheel2.stop();
```

Если мы применяем абсолютные адреса, нужно помнить, что они станут неверными, если машина будет находиться уже не на главном монтажном столе, а станет частью другого клипа.

Относительные адреса. Относительные адреса зависят от того, где находится объект, использующий этот адрес. Для обращения к родителю используют слово `_parent`. Например, чтобы из клипа `wheel1` остановить проигрывание для монтажного стола клипа `car`, надо применить команду

```
_parent.stop();
```

Эта команда работает из клипа `wheel2`, поскольку его родителем также является `car`.

Если два раза использовать `_parent`, мы обращаемся к «деду». Например, команда

```
_parent._parent.stop();
```

останавливает проигрывание на главном монтажном столе из клипа `wheel1` или `wheel2`.

К сыновьям можно обращаться просто по имени. Например, команда

```
car.stop();
```

допустима внутри `_root`. Кроме того, здесь можно было использовать и слово `this` (текущий монтажный стол):

```
this.car.stop();
```

Для остановки `wheel1` из `_root` можно применить одну из команд

```
car.wheel1.stop();  
this.car.wheel1.stop();
```

Наконец, самый сложный случай. Обратиться к «брату» можно только через родителя. Чтобы остановить второе колесо из `wheel1`, надо написать

```
_parent.wheel2.stop();
```

Добавьте в код кад 1 слоя *Программа* команды, останавливающие вращение колес, а в код обработчика нажатия кнопки — команды для запуска вращения колес. Сохраните файл и просмотрите результат.

IV. Автомобиль.

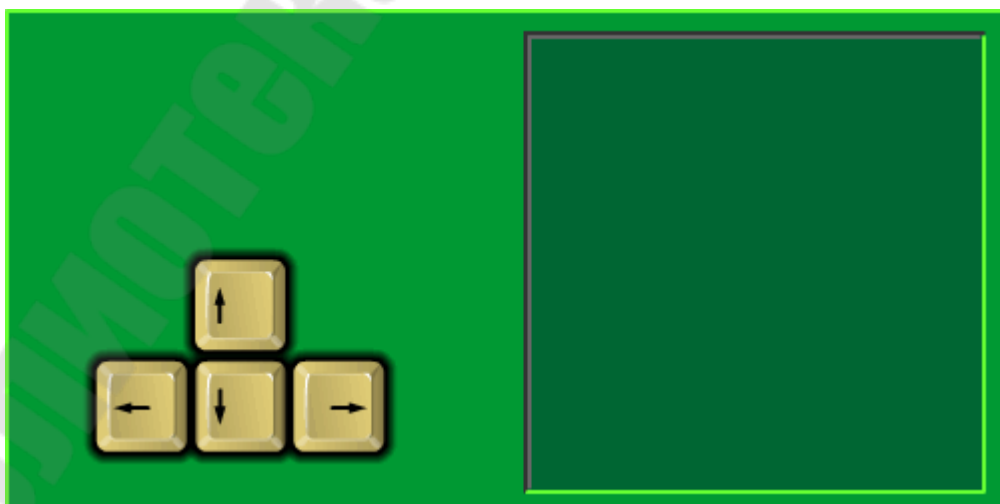
1. Откройте приложение Adobe Flash CS5 (создать ActionScript 2.0).
2. Переименуйте слой в **Background**. Создайте в этом слое два прямоугольника, как показано на рисунке.



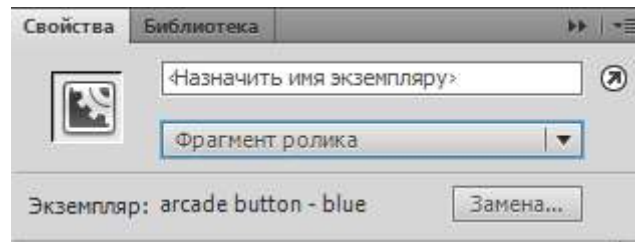
3. Создайте новый слой и переименуйте его в Кнопки.
4. Вставьте в слой Кнопки кнопки из общей библиотеки Кнопки/Classic Buttons/Key Buttons.



5. Расставьте кнопки соответствующим образом:



6. Преобразуйте каждую из кнопок (Кнопка) в фильм (Фрагмент ролика):



7. Для каждого из клипов-«кнопок» напишите код, обрабатывающий событие enterFrame обработчика onClipEvent:



```
onClipEvent (enterFrame)
{
if (Key.isDown(Key.LEFT))
{
gotoAndStop(3);
}
else
{
gotoAndStop(1);
}
}
```



```
onClipEvent (enterFrame)
{
if (Key.isDown(Key.DOWN))
{
gotoAndStop(3);
}
else
{
gotoAndStop(1);
}
}
```



```
onClipEvent (enterFrame)
{
    if (Key.isDown(Key.UP))
    {
        gotoAndStop(3);
    }
    else
    {
        gotoAndStop(1);
    }
}
```



```
onClipEvent (enterFrame)
{
    if (Key.isDown(Key.RIGHT))
    {
        gotoAndStop(3);
    }
    else
    {
        gotoAndStop(1);
    }
}
```

8. Создайте новый слой между слоями Background и Keys и переименуйте его в Avto.
9. Создайте в слое Avto символ-клип в виде автомобиля, присвоив ему(символу-клипу) имя Car



10. Напишите код, обрабатывающий события Load и enterFrame обработчика onClipEvent клипа Car. В данном примере это будет выглядеть так:

```
onClipEvent (load)
{
// Объявление и установка переменной скорости
    speed = 5;
}
onClipEvent (enterFrame)
{
// Движение влево, вправо, вперёд, назад
    if (Key.isDown(Key.LEFT) && !Key.isDown(Key.RIGHT))
    {
        _x -= speed;
        _rotation = 270;
    }
    if (Key.isDown(Key.RIGHT) && !Key.isDown(Key.LEFT))
    {
        _x += speed;
        _rotation = 90;
    }
    if (Key.isDown(Key.UP) && !Key.isDown(Key.DOWN))
    {
        _y -= speed;
        _rotation = 0;
    }
    if (Key.isDown(Key.DOWN) && !Key.isDown(Key.UP))
```

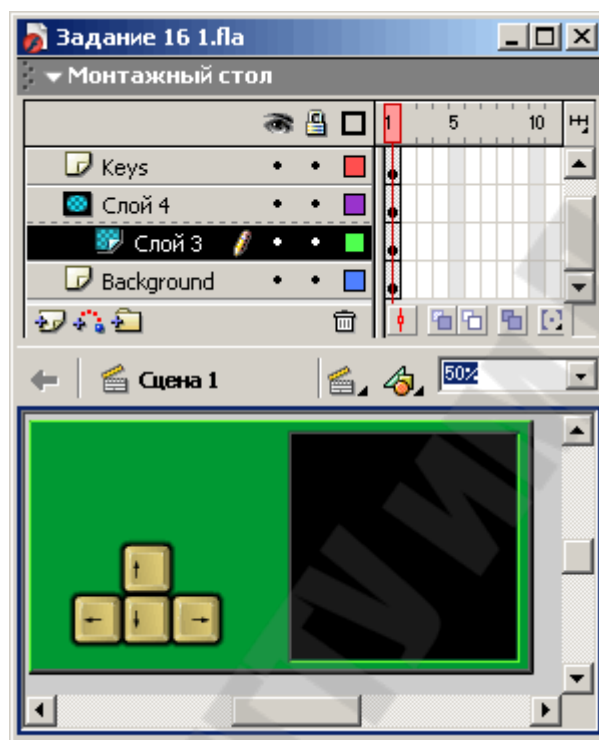
```

    {
        _y += speed;
        _rotation = 180;
    }
//
// Движение по диагонали
    if (Key.isDown(Key.LEFT) && Key.isDown(Key.UP) &&
        !Key.isDown(Key.RIGHT) && !Key.isDown(Key.DOWN))
    {
        _rotation = 315;
    }
    if (Key.isDown(Key.RIGHT) && Key.isDown(Key.UP) &&
        !Key.isDown(Key.LEFT) && !Key.isDown(Key.DOWN))
    {
        _rotation = 45;
    }
    if (Key.isDown(Key.LEFT) && Key.isDown(Key.DOWN) &&
        !Key.isDown(Key.RIGHT) && !Key.isDown(Key.UP))
    {
        _rotation = 225;
    }
    if (Key.isDown(Key.RIGHT) && Key.isDown(Key.DOWN) &&
        !Key.isDown(Key.LEFT) && !Key.isDown(Key.UP))
    {
        _rotation = 135;
    }
//
// возвращение на противоположную сторону замаскированной
// области
// когда машинка уходит с экрана
    if (_y < 0) { _y = 231; }
    if (_y > 231) { _y = 0; }
    if (_x < 231) { _x = 465; }
    if (_x > 465) { _x = 231; }
}

```

11. Теперь необходимо «замаскировать» машинку так, чтобы она появлялась в строго определённой зоне. Для этого нужно для слоя с машинкой создать слой маску:

- Выбрав слой с машинкой, создайте выше него новый слой. Переименуйте слой в Mask;
- Заливкой в слое Mask обозначьте квадратную область. Именно в ней и будет видна машинка;
- Преобразуйте слой Mask в слой-маску.



12. Готовое изображение сохраните в своей папке в формате *.fla.

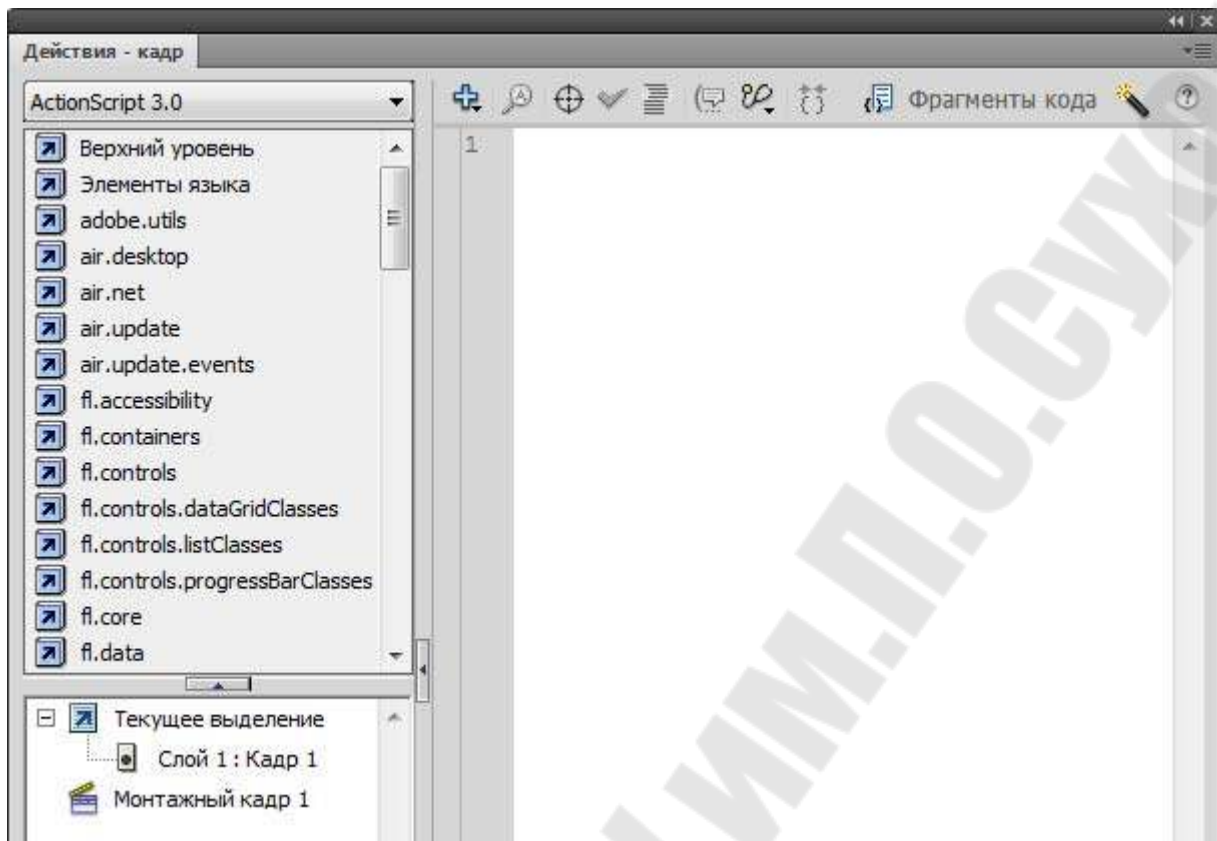
13. Протестируйте фильм, нажав сочетание клавиш **Ctrl+Enter**.

V. Создание интерактивного альбома.

Создадим набор фотографий и добавим кнопки, которые будут листать фото.

1. Поместите на основной монтажной линейке первое фото и добавьте кнопки из стандартного набора (например, кнопки похожие на клавиатурные клавиши «Влево» и «Вправо» с помощью команды **Окно – Общие библиотеки – Кнопки**).

2. Вызовите редактор Action Script по команде **Окно - Действия** или нажав клавишу **F9**. В результате появится окно редактора Action Script.



3. Для кнопки со стрелкой влево нам нужно формализовать следующий сценарий: «Если кнопка отпускается на некотором кадре, то с этого кадра необходимо перейти на предыдущий кадр». В соответствии с синтаксисом языка Action Script это будет выглядеть следующим образом:

```
on (release)
{
prevFrame();
}
```

В первой строчке записан обработчик событий кнопки `on()`, имеющий формат:

```
on (event)
{
текст сценария
}
```

Здесь `event` — это название события, в рассмотренном случае `release` (отпускание кнопки); может также отслеживаться такое событие, как `press` (нажатие кнопки), и другие события, которые мы рассмотрим позже.

Функция `prevFrame` — это функция безусловного перехода, которая перемещает воспроизводящую головку в предыдущий кадр. Она находится внутри обработчика событий, то есть может исполняться только в случае наступления описанного события.

4. Аналогично на вторую кнопку повесим код, обеспечивающий переход к следующему кадру:

```
on (release)
{
nextFrame();
}
```

5. Добавим несколько ключевых кадров так, что в них скопируются созданные в первом кадре кнопки, и поместим во вновь созданные кадры нужные фотографии.

Если запустить на исполнение созданный фильм, то кадры будут непрерывно проигрываться один за другим, а следовательно, прежде всего нам нужно дать команду «Стоп» на первом кадре. Для этого первому кадру добавим соответствующую команду.

6. Добавим еще две кнопки в наш интерактивный альбом «В первый кадр» и «В последний кадр». Выбрать кнопки подходящей мнемоники можно из стандартной библиотеки. Для этих кнопок нам понадобится команда `gotoAndStop()`, которая позволяет перейти к нужному кадру с последующей остановкой.

7. На кнопку перехода к первому кадру повесим следующий код:

```
on (release)
{
gotoAndStop(1);
}
```

8. На кнопку перехода к последнему (в данном случае пятому) кадру повесим следующий код:

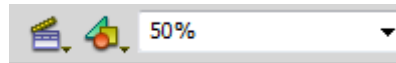
```
on (release)
{
gotoAndStop(5);
}
```

VI. Добавление пароля ролику.

В следующем примере мы научимся защищать данные с помощью пароля.

Для перехода к основной сцене нужно сначала ввести пароль. На основной сцене будет находиться побережье Ладожского озера

1. Откройте файл **panoram.fla**. Переименуйте слой *Layer 1* в *Фон*. Установите масштаб 50% (окно выбора масштаба в правом верхнем углу сцены) и перетащите на сцену клип *Панорама* из библиотеки.



2. На панели **Свойства** установите координаты клипа $X=-725$ и $Y=0$.

3. Выделите кадр 1 и перетащите его мышкой на кадр 2. В результате появился пустой кадр 1, в котором мы сделаем ввод пароля.

4. Выделите кадр 1 и перетащите на него рисунок sphinx.jpg из библиотеки. С помощью панели **Свойства** установите координаты рисунка $X=0$ и $Y=0$.

5. Включите инструмент **Текст**. Установите шрифт **Arial** светлого цвета, размер 20. Добавьте новое текстовое поле в углублении под лапами сфинкса. Выберите тип поля: **Вводимый**, в параметрах абзаца выберите поведение **Пароль**, дайте полю имя *pass*, по которому мы будем обращаться к нему из программы, в свойстве **Сглаживание** введите Шрифты устройства. Стиль текста **Пароль** говорит о том, что вместо вводимых символов на экране видны только звездочки, чтобы посторонние не смогли увидеть верный пароль.

6. Теперь сделаем, чтобы при вводе пароля «123» ролик переходил на следующий слайд с просмотром панорамы. Выделите кадр 1 и добавьте код:

```
stop();
pass.onChanged = function ()
{
if ( pass.text == "123" ) nextFrame();
}
```

В первой строчке мы остановили проигрывание, чтобы не было автоматического перехода на второй кадр. Дальше текстовому полю *pass* назначается обработчик события *changed* — оно возникает

каждый раз, когда текст в поле изменяется. При этом будет вызываться написанная нами функция-обработчик.

В условном операторе содержимое текстового поля, которое имеет адрес `pass.text`, сравнивается с правильным паролем «123». Если получено совпадение, происходит переход к следующему кадру по команде `nextFrame`.

7. Добавьте слева от поля ввода надпись *пароль* (статический текст, тип **Статический текст**). Проверьте работу фильма.

VII. Нарисуйте две кнопки, которые можно использовать в качестве воспроизведения и остановки ролика (пункт I).

VIII. Откройте один любой ролик из предыдущих заданий. Используя кнопки, созданные в задании 1, сделайте воспроизведение ролика управляемым (пункт II).

IX. Создайте интерактивный альбом с четырьмя кнопками (пункт V).

X. На основании примера 6 сделайте свой ролик, для запуска которого необходимо вводить пароль (пункт VI).

Контрольные вопросы

1. Что такое символ?
 2. Какие бывают символы и чем они отличаются?
 3. Какие ключевые кадры имеет символ-кнопка?
 4. Что такое панель Действия?
 5. Какие команды используются для управления воспроизведением ролика?
 6. Что такое «родитель» и «сын»?
 7. В чем различие относительных и абсолютных адресов?
 8. Как остановить анимацию на первом кадре, на последнем?
- (См. электронный курс лекций по дисциплине «Анимационная графика», раздел 3, символы, экземпляры и ресурсы библиотеки и раздел 6, создание интерактивных фильмов).

Учебное издание

**Андреева Дина Петровна
Старовойтова Марина Александровна**

АНИМАЦИОННАЯ ГРАФИКА

**Лабораторный практикум
по одноименной дисциплине
для слушателей специальности 1-40 01 74
«Web-дизайн и компьютерная графика»
заочной формы обучения**

Подписано к размещению в электронную библиотеку
ГГТУ им. П. О. Сухого в качестве электронного
учебно-методического документа 10.04.13.

Рег. № 60Е.

<http://www.gstu.by>