

МУНИЦИПАЛЬНОЕ БЮДЖЕТНОЕ ОБЩЕОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ  
«АЛФИМОВСКАЯ СРЕДНЯЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ ШКОЛА»  
СТУПИНСКОГО МУНИЦИПАЛЬНОГО РАЙОНА  
142860 Московская область Ступинский район д. Алфимово улица Новосёлов владение 20  
Тел.\ факс 8(49664) 5-93-10 [Alfimovo@rambler.ru](mailto:Alfimovo@rambler.ru)

---

Районный конкурс исследовательских работ обучающихся  
Ступинского муниципального района

Предметная секция: ИНФОРМАТИКА

ПРОЕКТ

Тема: «Создание программой анимации»

Авторы работы: Луговских Венера Алексеевна,  
7 класс

Масин Вадим Александрович,  
7 класс

Руководитель: Авилова Светлана Владимировна,  
учитель информатики

Ступино

## Оглавление

Аннотация .....	3
Введение.....	4
Глава I. Учимся управлять черепашкой.....	6
1.1. Программная среда ЛогоМиры и ее возможности.....	6
1.2. Программное управление Черепашкой .....	6
1.3. Мультимедийные возможности программы.....	6
1.4. Еще немного о программном управлении .....	7
Глава II. Создание ролевого мультфильма .....	8
2.1. Принцип модульности .....	8
2.2. Модуль 1: движение машины и мальчика .....	8
2.3. Модуль 2: работа светофоров .....	9
2.4. Модуль 3: заключительный.....	10
2.5. Основной модуль .....	10
Заключение .....	12
Список использованных источников и литературы .....	13
Приложение 1 .....	14
Приложение 2 .....	15

### **Аннотация**

Данная работа выполнена в форме проекта, имеет четко сформулированную проблему: создание программной анимации, этапы реализации и получения готового продукта – анимационного ролика. В ходе работы описана технология создания программной анимации с использованием языка программирования ЛогоМиры 2.0.

Работа состоит из введения, двух глав, заключения, списка используемых источников и литературы, приложения.

В введении описывается актуальность выбранной проблемы, различные способы создания анимации, бесплатные программы для создания анимации, обоснованность выбора программной среды ЛогоМиры. А также введение содержит цель и задачи проекта, методы, используемые в работе, краткий литературный обзор по теме.

В первой главе описывается технология работы в программной среде ЛогоМиры 2.0: ее интерфейс, графические и мультимедийные возможности, способы управления графическим исполнителем – Черепашкой.

Во второй главе представлена технология создания анимационного ролика с подробным описанием каждого этапа.

Заключение содержит выводы о практическом применении проектного продукта, а также перспективы дальнейшего роста авторов проекта.

В конце работы представлена библиография использованных источников и приложение, в котором представлены материалы, используемые в проекте.

## Введение

На первом занятии внеурочной деятельности мы посмотрели видеоролик «Как делали Незнайку на Луне» ([www.youtube.com](http://www.youtube.com) «Как делали Незнайку на Луне»), познакомились с профессиями: сценарист, художник-мультипликатор, звукорежиссер и многими другими. А также с этапами создания мультфильмов. Конечно, нам захотелось создать свой мультфильм. Для этого необходимо выбрать способ создания мультфильма, изучить литературу, придумать сценарий, героев, озвучить их и найти практическое применение нашему ролику. Но сначала разберемся в терминах мультфильм и анимация.

Мы заглянули в Школьный словарь иностранных слов:

**АНИМАЦИЯ**, -и; ж. [от лат. *anima* - душа] – создание иллюзии движения на основе большого числа последовательных рисунков и их съемки на киноплёнку.

**МУЛЬТИПЛИКАЦИЯ**, -и; ж. [от лат. *multiplicatio* - умножение] – киносъемка последовательных фаз движений рисованных или объемных фигур, в результате чего у зрителей создается впечатление движения.

Получается, что эти слова являются синонимами.

Существуют различные способы создания мультфильмов: рисованная анимация, кукольная анимация, пластилиновая анимация, песочная анимация, компьютерная анимация, но нас интересовала программная анимация.

Сейчас существует множество компьютерных программ для создания мультфильмов. В сети Интернет мы нашли бесплатные программы для создания мультфильмов - Plastic animation paper, Synfig Studio, Creatoon. С их помощью можно создать профессиональные анимационные ролики, используя технологию, создания промежуточных кадров, в которых изображения плавно перетекают из одной формы в другую. И многие другие... Но все они рассчитаны на создание рисованной анимации, а нас интересовала анимация, созданная с использованием языка программирования. Поэтому мы остановили свой выбор на имеющейся в школе программе ЛогоМиры.

Цель данной работы: создание анимации с помощью языка программирования ЛогоМиры.

Так как мы выбрали способ - создание программной анимации, то наша проблема делится на следующие задачи:

1. Придумать идею для анимационного ролика, разбить ее на этапы реализации, определить характеристики каждого объекта для создания процессов движения, выбора героев, озвучивания;
2. Изучить программную среду ЛогоМиры и ее возможности: графические для создания собственного микро-мира, процессы управления, мультимедийные возможности для озвучивания героев;
3. Научиться создавать программы, процедуры и программные модули: познакомиться с понятием алгоритма и его видами, познакомиться с системой команд исполнителя, научиться использовать ручной и программный процессы управления;
4. Изучить процессы движения разных объектов, создать графические модели движения для главных героев анимации;
5. Создать декорации, определить передний и задние планы, нарисовать новые костюмы для черепашки;
6. Записать звуковые фрагменты, музыкальные заставки;
7. Написать процедуры для анимации каждого объекта, объединить их в программные модули и создать основную программу (т.е. используем программирование «снизу вверх»)
8. Создать кнопку для одновременного запуска всех героев, переключения между листами программ.
9. Провести компьютерный эксперимент, проанализировать полученный результат, при возникновении ошибок отладить программу, рассчитать временные промежутки для синхронного движения главных героев.

В своей работе мы использовали следующие методы: сбор информации из различных источников, сравнение, анализ, создание программы и ее отладка, синхронизация различных процессов.

Справиться с поставленными задачами нам помогли различные литературные и интернет-источники:

- книга «Программирование в среде Лого. Первые шаги» автор Белова Г. В. познакомила нас со средой программирования, научила управлять черепашкой, создавать различные мини-проекты, которые мы использовали как программные модули в нашем анимационном ролике;
- в качестве дополнительной информации мы также использовали учебную литературу: «Информатика {начальный курс}», «Информатика практикум по информационным технологиям {базовый курс}» под редакцией профессора Н. В. Макаровой;
- сайт <http://prologo.звуконлайн.рф> – сайт любителей программ ЛогоМиры и ПервоЛого и многие другие.

## Глава I. Учимся управлять черепашкой

### 1.1. Программная среда ЛогоМиры и ее возможности

Программная среда Лого состоит из трех окон: Рабочее поле; Поле команд; Лист программ, имеется панель с инструментами. На Рабочем поле отображается графический исполнитель и главный герой - Черепашка. Она может передвигаться по рабочему полю, рисовать. При открытии программы Черепашка находится в центре Рабочего и смотрит вверх (см Приложение 2).

С помощью панели рисования мы создали пейзаж – микро-мир для нашей анимации, а также нарисовали новые формы, которых нам не хватало для создания мультфильма. Это светофор для машин – три формы с горящим красным светом, желтым и зеленым. Две формы для пешехода, и две формы для пешеходного светофора.

Дальше мы начали изучать команды движения и формы записи. Самая простая команда **вперед** (краткая форма записи **вп**) – она заставляет Черепашку двигаться вперед туда, куда смотрит ее голова. У всех команд Черепашки есть параметры, они уточняют ее действия. У команды **вперед** тоже есть параметр, он показывает сколько шагов должна сделать Черепашка. Значит, по команде **вп 100** Черепашка пройдет 100 маленьких шагов вверх (один шаг – один пиксель). Далее мы создали таблицу команд Черепашки, которая пополнялась с появлением новых команд (см. Приложение 1).

С самого начала мы командовали Черепашкой из Командного центра. Черепашка выполняла команды по очереди, одну за другой. Такая организация действий называется последовательной, а, запись команд последовательным алгоритмом. Управление Черепашкой из Командного центра является примером ручного управления и не всегда удобно. Поэтому в программе ЛогоМиры есть два вида программного управления: диалоговое окно Черепашки и Лист программ.

### 1.2. Программное управление Черепашкой

Чтобы управлять Черепашкой, живущей в проекте нужно использовать Диалоговое окно. Это окно содержит изменяемое поле имени Черепашки, инструкцию для Черепашки и переключатель, заставляющий выполнять эту инструкцию либо один, либо много раз. Так мы познакомились с организацией циклических действий Черепашки или циклическим алгоритмом. Наконец мы смогли создать свой первый проект, который назывался «Движение»: мы заставили двигаться машину, самолет, облака, солнышко. Чтобы они двигались с разной скоростью, необходимо было замедлить движение Черепашки с помощью новой команды **жди**.

Когда мы попытались заставить двигаться человека таким же образом, мы столкнулись с проблемой - его движения выглядели странно, как движения оловянных солдатиков, у которых не двигались руки и ноги. Нам пришлось вспомнить наши первые занятия, когда мы знакомились с разными способами создания анимации. Оказывается способов много, но у них у всех один принцип: объект будет двигаться, если меняется его поза. Значит, движения человека будут выглядеть естественными, если мы будем менять его позу (смена рук и ног: сначала одна нога впереди, остановка, затем другая рука и нога, остановка и т. д.). В меню форм есть заготовки для таких движений, а если их нет, то надо нарисовать, используя графический редактор. Также мы познакомились с новой командой, которая позволяла менять форму: **новая форма**. Теперь и человек, и собака, и рыбка, и птичка двигались естественно. Мы сделали множество разных проектов, оживляя этих героев. Казалось, мы научились всему, чтобы реализовать нашу идею.

### 1.3. Мультимедийные возможности программы

В нашей заготовке приходилось часто использовать компьютерную мышь, чтобы заставлять двигаться разные объекты, это было очень неудобно, чуть запоздаешь и процесс движения нарушался. Необходимо было синхронизировать процессы. Также научить наших героев разговаривать, или вставить музыкальную заставку, чтобы мультфильм не

выглядел скучным. Поэтому мы познакомились с мультимедийными возможностями нашей программы (см Приложение 2).

#### **1.4. Еще немного о программном управлении**

При выполнении предыдущих проектов мы пользовались диалоговым окном Черепашек. Для записи и отладки простеньких программ им пользоваться было удобно, но если команд было много и они все были не видны, приходилось пролистывать для нахождения нужной команды, что было не очень удобно и совсем запутанно при отладке программы и поиске ошибки. Поэтому настоящие многомодульные программы лучше всего писать на листе программ. Текст программы записывается по строгим правилам оформления (см. Приложение 2)

Для вызова программы в Командном центре или диалоговом окне Черепашки используется только имя программы.

## Глава II. Создание ролевого мультфильма

### 2.1. Принцип модульности

При решении сложной задачи, например в математике, мы разбиваем ее условие на более простые подзадачи (отдельные модули), которые решая поочередно, мы решаем исходную задачу. Такой принцип решения и называется принципом модульности. Он позволяет упростить и улучшить общую структуру программы.

Итак, наш сюжет будет таким: проезжую часть с двигающимися по ней машинами, светофором и пешеходным переходом переходит мальчик, соблюдая правила дорожного движения. Этот сюжет мы взяли из книги Беловой Г. В. Программирование в среде ЛОГО. Первые шаги. – М.: СОЛОН-ПРЕСС, 2007 (стр. 42), добавили работу с текстом, музыкой. Анимационный ролик обучает дошкольников и учеников начальной школы правильно переходить проезжую часть. Разобьем условие задачи на более простые этапы.

Этап 1: создание декораций, привязка дороги к системе координат, так будет легче программировать движение машины и пешехода. Для декорации будем использовать уже созданный проект «Мой микро-мир» изменив в нем расположение дороги.

Этап 2 (модуль 1): движение мальчика и машины. Сначала установим наши объекты в начальное положение, определим направления и траектории движения.

Этап 3 (модуль 2): работа светофора. Изначально, светофор для машины горит зеленым светом, затем должен переключиться на желтый, через какое-то время – на красный. Машина в этот момент останавливается, а пешеход переходит дорогу. Воспользуемся уже созданными формами светофора в проекте «Мой микро-мир». Создадим еще две новые формы светофора для пешехода, который будет гореть красным светом изначально, зеленым при переходе пешеходом улицы.

Этап 4 (модуль 3 - заключительный): светофор для пешехода переключается на красный свет - пешеход движется по другой стороне дороги, светофор для машины переключается на зеленый свет - машина едет дальше.

Этап 5 (основной модуль): сборка готовых сцен и создание кнопки для синхронного начала движения, отладка всех модулей.

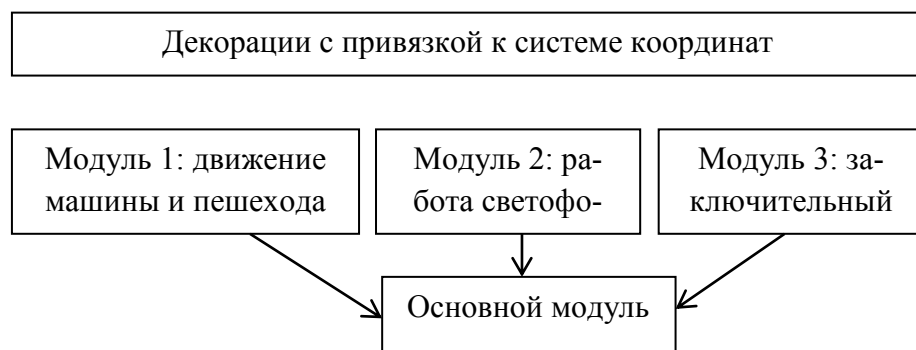


Рис. 2 Структура программы

### 2.2. Модуль 1: движение машины и мальчика

Создадим траекторию движения машины.



Рис. 3

Согласно этому рисунку составляем алгоритм:

Начало

Установить машину в начальное положение (-372, -100)

Направление движения 90

Повторить 9 раз (вперед 50 жди 2)

Конец



Переводим на язык программирования:

Это машина  
 нм [-372 -100]  
 нк 90  
 нф 28  
 повтори 9 [вп 50 жди 2]  
 конец

Количество повторений необходимо рассчитать при запуске всего модуля для синхронизации движения мальчика и машины.

Теперь создаем траекторию движения пешехода.

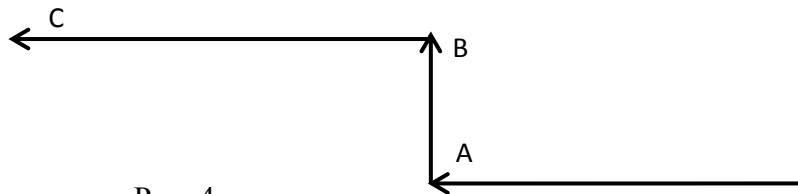


Рис. 4

На первом этапе движение пешехода длится до точки А.

Составляем алгоритм:

Начало

Установить пешехода в начальное положение (372, -140)

Направление движения 270

Повторить 11 раз (новая форма 42 вперед 5 жди 2 новая форма 43 вперед 5 жди 2 новая форма 44 вперед 5 жди 2)

Конец

Переводим на язык программирования:

Это пеш1  
 нм [372 -140]  
 нк 270  
 повтори 10 [нф 42 вп 5 жди 2 нф 43 вп 5 жди 2 нф 44 вп 5 жди 2]  
 нф 45  
 конец

Собираем модуль1

Это модуль1

для [авто] машина

для [чел] пеш1

конец

Запускаем модуль с помощью Командного центра, производим отладку процедур, рассчитываем количество повторений. Первый модуль готов, но процедуры Машина и Пешеход будут дополнены при создании Модуля 2 и Модуля 3.

### 2.3. Модуль 2: работа светофоров

Для работы этого модуля составляем алгоритмы светофоров и вносим коррективы в алгоритм и процедуру пешехода.

Алгоритм работы светофора для машины:

Начало

новая форма 15 жди 30 новая форма 14 жди 5 новая форма 13

Конец

Переводим на язык программирования:

Это светофор1  
 нф 15 жди 30 нф 14 жди 5 нф 13  
 конец

Алгоритм работы светофора для пешехода:

Начало  
 новая форма 29 жди 15 новая форма 30 жди 5  
 Конец

Переводим на язык программирования:

Это светофор2  
 нф 29 жди 15 нф 30 жди 5  
 конец

Вносим дополнение в процедуру Пешеход. На этом этапе пешеход переходит дорогу (траектория от А до В), значит меняется направление его движения, также используем нарисованную форму для пешехода.

Фрагмент алгоритма движения пешехода по траектории АВ

Поворот направо

Повторить 16 раз (новая форма 45 вперед 5 жди 2 новая форма 46 вперед 5 жди 2)

Дополнение в процедуру Пешеход

Это пеш2  
 нк 0  
 Повтори 16 [нф 45 вп 5 жди 2 нф 46 вп 5 жди 2]  
 Собираем модуль2.  
 Это модуль2  
 для [свет2]светофор2  
 для [чел] пеш2  
 конец

Используя Командный центр отлаживаем этот модуль.

#### 2.4. Модуль 3: заключительный

На этом этапе продолжается движение машины, пешеход движется по траектории ВС.

Дополнение в процедуру движения машины  
 повтори 21 [вп 50 жди 2]

Дополнение в процедуру движения пешехода  
 нк 270

повтори 5 [нф 42 вп 5 жди2 нф 43 вп 5 жди 2 нф 44 вп 5 жди 2]

Собираем модуль3

это модуль3  
 для [свет1] светофор1  
 для [свет2] нф 29  
 для [авто] повтори 21 [вп 50 жди 2]  
 для [чел] пеш3  
 конец

#### 2.5. Основной модуль

Используя написанные ранее модули, собираем основной. Назовем его ролик.

Это ролик  
 модуль1  
 модуль2  
 модуль3  
 конец

Добавим текстовое окно с надписью: «Уважаемые пешеходы! Переходите дорогу только в разрешенных местах и на зеленый свет светофора»

Для этого изучим новую команду – **новый текст** имя [х у] [длина ширина]. Она создает текстовое окно, чтобы его можно было использовать в программе должен быть параметр «имя», координаты х и у задают положение текста на рабочем поле, а параметр

[длина ширина] определяют размер текстового окна. Теперь заполняем текстовое окно, наша процедура будет выглядеть так:

```

Это текст
нт "правило [-372 213] [744 426]
нрш 45 нцт 15 нс "жирный
вставь [Уважаемые пешеходы!]
нрш 35
вставь [Переходите дорогу только в разрешенных местах и на зеленый свет свето-
фора!]
```

конец

Добавим музыкальную заставку, чтобы было не так скучно смотреть наш ролик.

Для этого используем команду **Возьми – Музыка** в меню **Файл, запусти [музыка]**.

Корректируем основной модуль:

```

Это ролик
запусти [музыка]
модуль1
модуль2
модуль3
текст
конец
```

Создадим кнопку, чтобы все процессы запускались синхронно. В диалоговом окне кнопки напомним имя основного модуля **ролик**.

## Заключение

Итак, наш ролик готов. Мы изучили секреты создания программой анимации, научились создавать процедуры, объединять их в модули, синхронизировать процессы. Конечно, этот вид анимации не станет популярным, но он помогает в интересной форме научиться программировать. Мы не остановимся на достигнутом, будем создавать новые проекты: «Физкультминутка» для учащихся начальной школы, модель «Планеты Солнечной системы» и игра «Мины».

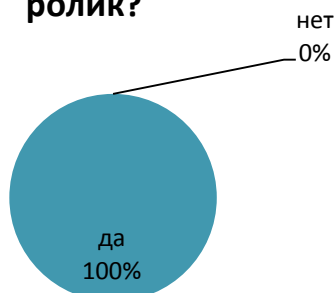
А наш ролик демонстрируем учащимся начальных классов и проводим опрос. Вот его результаты:

Всего было опрошено: 18 учащихся

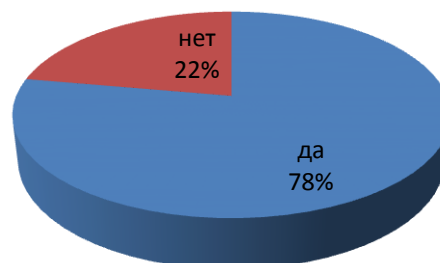
1. Понравился ли вам этот ролик?  
а) да – 18 чел – 100%                      б) нет – 0 чел – 0%
2. Считаете ли вы, что этот ролик выполнен профессионально?  
а) да – 14 чел – 78%                      б) нет – 4 чел – 22%
3. Чему учит этот ролик? (напишите сами)  
Правильно переходить дорогу – 12 чел – 67%  
Соблюдать правила дорожного движения - 5 чел – 28%  
Безопасности – 1 чел – 5 %

Наша цель достигнута, практически всем опрошенным наш ролик понравился, многие посчитали нашу работу профессиональной, все ребята правильно поняли, чему мы хотели научить их.

**Понравился ли вам этот ролик?**



**Считаете ли вы, что этот ролик выполнен профессионально?**



**Чему учит этот ролик?**



### **Список использованных источников и литературы**

1. Белова Г. В. Программирование в среде ЛОГО. Первые шаги. – М.: СОЛОН-ПРЕСС, 2007 (стр. 7 – 46)
2. Информатика. 5-6 класс. Начальный курс/Под ред. Н. В. Макаровой. – СПб.: Питер, 2005 (стр. 89-131)
3. Информатика. 7-9 класс. Базовый курс. Практикум по информационным технологиям/Под ред. Н. В. Макаровой. – СПб.: Питер, 2007 (стр. 225-258)
4. Сайт <http://int-edu.ru/logo/products.html>
5. Сайт <http://prologo.звуконлайн.рф>
6. Сайт <http://www.youtube.com> «Как делали Незнайку на Луне»
7. Сайт <https://ru.wikipedia.org/wiki/Мультфильм>
8. Сопрунов С.Ф. Непростое программирование на Лого М., 2009. Электронное издание
9. Школьный словарь иностранных слов под редакцией В. В. Иванова. – М.: Просвещение, 2006 (стр 28, стр 198)
10. Яковлев И.Н., Яковлева Е.И. ЛогоМиры 3.0: Сборник методических материалов. Электронное издание

## Приложение 1

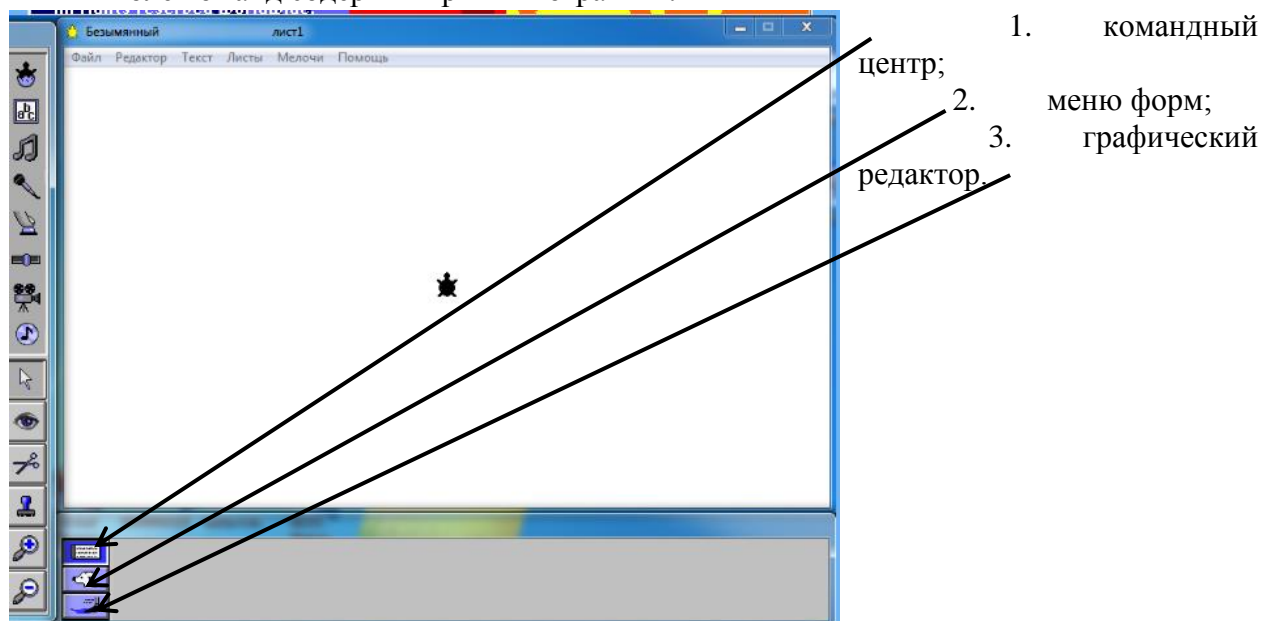
Таблица команд Черепашки по окончании проекта

Команда	Параметр	Сокращенная форма	Примечание
Вперед	Количество шагов	вп 100	Черепашка идет вперед на 100 пикселей
Назад	Количество шагов	нд 100	Черепашка идет назад на 100 пикселей
Направо	Количество градусов	пр 90	Поворот направо
Налево	Количество градусов	лв 90	Поворот налево
Жди	Число	жди 2	Черепашка замирает
Перо опусти	Нет параметра	по	Начинает рисовать
Перо подними	Нет параметра	пп	Не рисует
Новый курс	Количество градусов 	нк 45	Поворачивает Черепашку в строго определенном направлении
Новое место	Координаты x и y	нм [50 50]	Перемещает Черепашку в точку с координатами x и y
Новая форма	Номер формы из меню форм	нф 12	Одевает Черепашку в форму с данным номером
Новый текст	имя [x y] [длина ширина].	нт "текст" [50 50] [100 100]	Создает текстовое окно с именем текст, левый верхний угол расположен в точке с координатами (50, 50) размерами (100, 100)
Новый размер шрифта	Число (максимальный размер шрифта 1638 ст)	нрш 45	Устанавливает размер шрифта
Новый цвет текста	Слово-или-число Слово "красный" Число (графический редактор)	нцт "красный" нцт 12	Изменяет цвет текста
Новый стиль	Слово-или-список	нс "курсив"	Устанавливает стиль шрифта
Пиши	Слово-или-список	-	Печатает слово или список в текстовом окне

## Приложение 2

### Интерфейс программы ЛогоМиры

Поле команд содержит три пиктограммы:



### Мультимедийные возможности программы

Первый объект – бегунок. Он позволяет изменять скорость объекта, направление, цвет и многие другие параметры. Так с помощью бегунка **скорость** смогли создать три этапа движения машины: начало движения (медленно набирает скорость), основное движение (движется быстро и равномерно), торможение (замедление движения до полной остановки). С помощью бегунка **направление** самолет начинал взлетать и приземляться, бегунка **цвет** наша черепашка меняла цвет пера.

Второй объект – кнопка. Ее предназначение что-то включать или выключать. Инструкция написания в кнопке может быть выполнена разными черепашками. Нажав кнопку, мы запускаем процесс, отжав кнопку, мы его завершаем и не надо больше «гоняться» за Черепашкой, чтобы ее остановить. А еще кнопки позволяют синхронизировать процессы, используя инструкцию **каждая [включить]** и **каждая [выключить]**. Для синхронизации двух процессов используется еще один способ: первый процесс запускается командой **всегда**, затем запускается второй процесс, и как только второй процесс заканчивается, первый прекращается командой **отмени**.

Третий объект – музыкальный редактор. В ЛогоМирах можно создавать свои простенькие мелодии, если вы понимаете нотную грамоту, а можно проиграть мелодию, используя ее имя в качестве команды для Черепашки или в окне кнопки.

### Правила оформления программы на языке ЛогоМиры

1. Каждая программа имеет заголовок, в котором указано ее имя. Заголовок начинается со слова **это (служебное слово)**, после которого через пробел надо написать имя программы. Имя программы не должно содержать пробелов и не совпадать со служебными словами ЛогоМиров.
2. Программа содержит тело, в котором на языке программирования записан алгоритм. Тело программы, обязательно начинается с новой строки. Алгоритм записывается так же, как последовательности команд, вводимые в диалоговом окне Черепашки или командном центре.
3. Программа имеет конец, который является признаком завершения программы. Слово **конец** пишется на отдельной строке и отмечает окончание каждой программы.