

ЛОГО МИРЫ – СРЕДА ДЛЯ МОДЕЛИРОВАНИЯ

ВИЗУАЛИЗАЦИЯ ОБЪЕКТОВ, ПРОЦЕССОВ И СОБЫТИЙ

На этом занятии мы познакомимся с новой версией системы Лого Миры – MicroWorld Pro.

Мы будем работать с англоязычной демоверсией (<http://www.microworlds.com/solutions/mwpro.html>), так как к настоящему времени в России не существует локализованного варианта.

Система MicroWorld Pro представляет собой более совершенную с точки зрения объектно-ориентированного программирования (ООП) среду программирования. В нее включены инструменты для анализа созданных объектов и процессов. Общий вид интерфейса системы MicroWorld Pro представлен на рисунке 1.

Помимо привычных областей интерфейса – меню, панели инструментов, рабочего поля и командного центра, здесь присутствуют так называемые Инспектора (справа). Инспектора – это окна для анализа и изменения состояния проекта. Имеется 4 окна:

- Окно процедур (Procedures);
- Окно проекта (Project);
- Окно процессов (Processes);
- Окно графического редактора (Graphics)

Вы можете открывать один из этих Инспекторов.

Окно процедур так же, как и в системе Лого Миры, содержит программы.

Окно графического редактора содержит инструменты для создания на Рабочем поле графических объектов и инструменты для программирования и измерения цвета. По сравнению с графическим редактором системы Лого Миры, здесь добавлены коллекция видов кисти, возможность изменения прозрачности цвета, инструмент для измерения цвета.

Окно проекта показывает, какие экземпляры объектов созданы в вашем проекте. В виде иерархического дерева представлены объекты-родители и

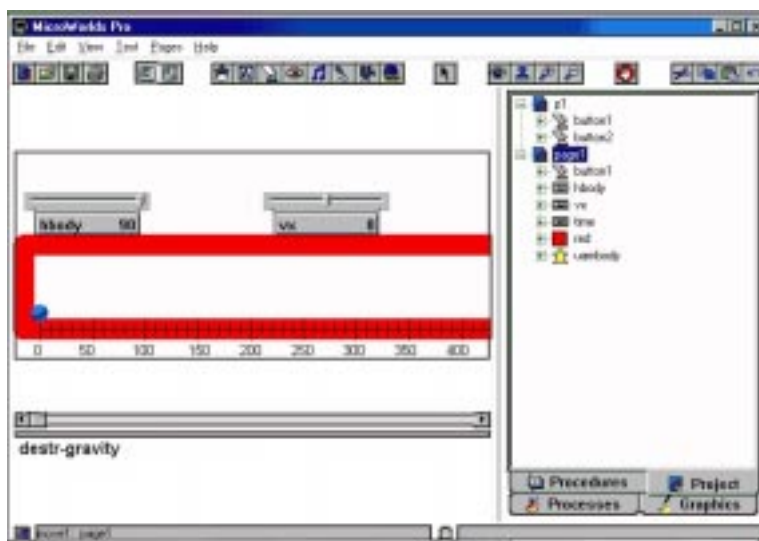
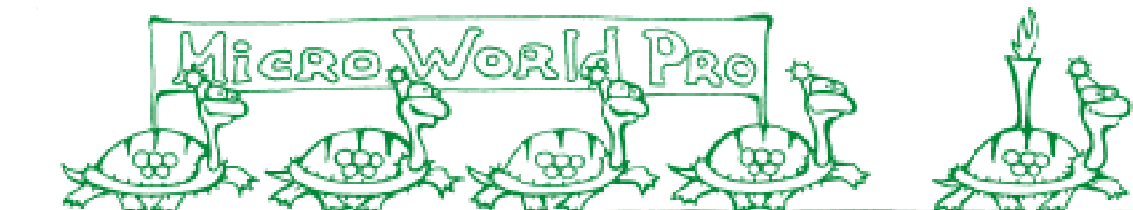


Рисунок 1



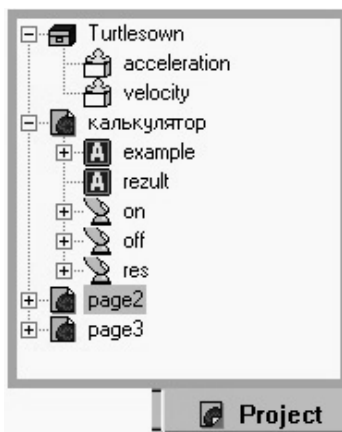


Рисунок 2

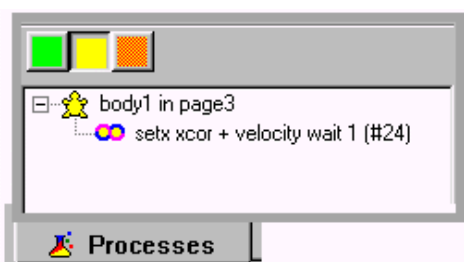


Рисунок 3

объекты-наследники (рисунок 2). Если объектам назначены методы, то показаны события, при которых эти методы активизируются. При нажатии правой клавиши мыши вызывается контекстное меню, с помощью которого вы можете посмотреть или изменить свойства экземпляров объектов, удалить экземпляры объектов или их методы, обеспечивать трассировку выполнения методов.

На прошлом занятии мы создавали объект КАЛЬКУЛЯТОР. В окне Project представлен экземпляр объекта Page – калькулятор, в котором созданы экземп-

ляры объектов ТЕКСТОВОЕ ОКНО (example, rezult) и КНОПОК (on, off, res).

Щелчком левой клавиши мыши открывается следующий уровень иерархии. Здесь показаны методы, которые назначены для перечисленных объектов, и события, по которым методы начинают выполняться.

Окно процессов содержит описание активных процессов (рисунок 3).

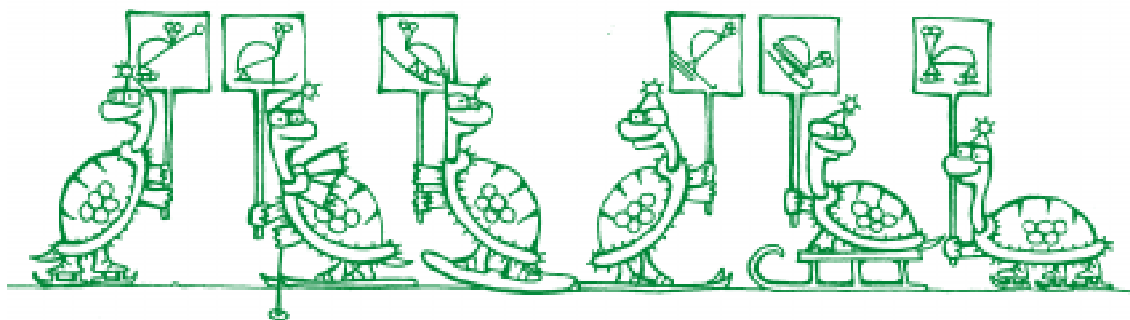
Каждое семейство процессов привязано к конкретному экземпляру объекта. В виде иерархии показываются процессы-родители и процессы-наследники. Инспектор процессов позволяет изменять скорость выполнения процессов (щелчком мыши по цветным кнопкам наверху окна) и останавливать их.

В окне процессов видны только активные процессы. После завершения процесса информация о нем в этом окне исчезает.

Поясним еще раз понятия *процесс* и *событие*.

Процессом называют последовательную смену состояний объекта под влиянием выполнения им или над ним действий. Эта смена состояний происходит в течение определенного времени. Можно сказать, что *процесс* – это выполнение назначенных объекту методов.

Выполнение метода начинается при возникновении определенного *события*. Как в любой среде ООП, в среде Лого Миры (так же, как и в MicroWorld Pro) все программы управляются *событиями*. *Событием* назовем небольшой пакет информации, который создается в ответ на то или иное действие пользователя. Этот пакет информации передается от объек-



та к объекту (по иерархии) до тех пор, пока не найдется объект, который сможет обработать его (у которого есть соответствующий метод). *События* сами по себе не производят никаких действий. Но в ответ на *событие* могут запускаться *процессы*, создаваться новые экземпляры объектов...

Мы уже перечисляли встроенные в систему Лого Миры (и MicroWorld Pro) события: от мыши, от клавиатуры, от Черепашки, от цвета, от кнопки, от завершения процесса. Кроме того, программист может описать свои события, наследующие свойства встроенных. При назначении метода объекту указывается явно (визуально или программно) событие, по которому этот метод начинает работать. Например, в КАЛЬКУЛЯТОРЕ метод ВКЛЮЧИТЬ начинает работать при нажатии на кнопку ON. Внутренняя процедура **answ** начинает работать при нажатии клавиши ENTER на клавиатуре и т.д.

Пользовательские события оформляются как логические выражения, значением которых может быть логическая константа – **true** или **false**. Тогда обнаружение определенного события и запуск соответствующего метода осуществляется с помощью команды **waituntil**.

Запуск процессов осуществляется командами **launch** и **forever**. Если один процесс порождает другие процессы, то их называют процессами – наследниками.

Такая мощная объектно ориентированная среда программирования может служить для создания моделей, предназначенных для изучения вопросов физики, химии, математики, языков и других областей знания. Каждая такая работа представляет собственный интерес и должна

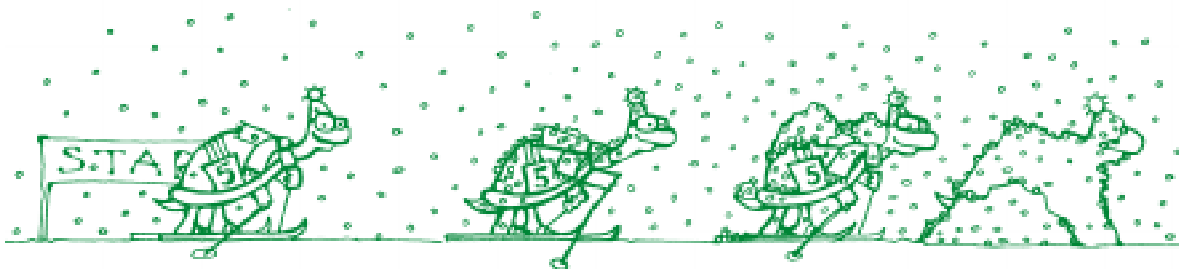
быть опубликована (например, смотри www.microworlds.com/library/). Летом 2001 года состоялась конференция EUROLOGO 2001, на которой были представлены такие серьезные модели, как функционирование системы (вопросы синергетики), поведение тела в поле всемирного тяготения (физика) и др. Среда Лого Миры рассматривалась как философская среда для общения Учителя и Ученика в процессе создания нового Знания.

В качестве примера попытаемся моделировать законы движения тела. Идея этого проекта заимствована из учебного пособия А.Г. Юдиной «Практикум по информатике для общеобразовательной школы» и материалов, представленных ею на конференцию EUROLOGO 2001.

МОДЕЛИРОВАНИЕ ДВИЖЕНИЯ

Обычно для исследования явления создают специальную лабораторную установку. Также и мы на Рабочем поле построим своего рода экспериментальную установку – ограниченное поле, внутри которого движется объект – Body. Направление движения объекта Body задается направлением головы. Направление головы меняется под влиянием каких-то внешних по отношению к Body сил. Граница поля наблюдения оказывает воздействие на движущееся внутри поля тело – как только тело достигает этой границы, движение останавливается.

Объект Body является наследником Черепашки. Поэтому Body наследует все свойства и методы Черепашки. Кроме того, Body обладает своими особыми свойствами – скоростью, ускорением и своим особым методом – двигаться с заданной скоростью.



Объект	Свойства (возможные значения)	Методы
Поле движения	Цвет границы, толщина границы, начальное положение поля, высота, ширина	Обнулять скорость движущегося тела
Показатель времени движения	Имя, положение на Рабочем поле, минимальное, максимальное и текущее значения (в условных единицах времени)	Изменять значение
Показатель направления	Имя, положение на Рабочем поле, минимальное, максимальное и текущее значения (в градусах)	Изменять значение
Показатель скорости	Имя, положение на Рабочем поле, минимальное, максимальное и текущее значения (в пикселях)	Изменять значение
Движущийся объект	Скорость по оси X	Двигаться с заданной скоростью в течение заданного времени,
	Состояние пера (возможные значения pu или pd)	Поднимать и опускать перо
	Начальное положение тела	Менять направление

Таблица 1

В зависимости от подготовки участников экспериментов, среду исследования можно создать как визуально, так и программно. Программное создание среды обычно осуществляется на этапе публикации и внедрения эксперимента в учебный процесс, так как программное описание занимает гораздо меньше места, легко может быть установлено на сайт. В этой статье мы приведем программы для конструктора экспериментальной установки, так как нас интересуют также вопросы программирования.

ПРЯМОЛИНЕЙНОЕ РАВНОМЕРНОЕ ДВИЖЕНИЕ

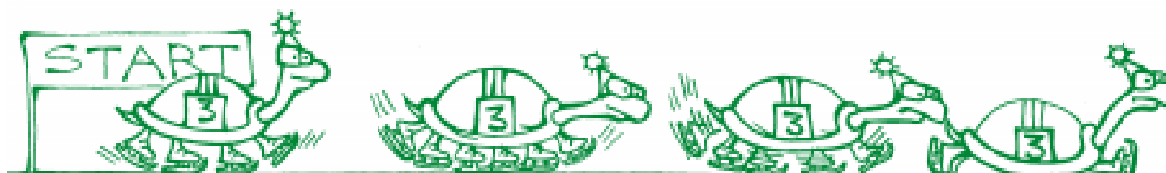
При таком движении телу придана постоянная скорость, направленная, например, только вдоль оси X.

Информационная модель установки для моделирования равномерного прямолинейного движения приведена в таблице 1.

Скоростью называют расстояние (мы будем его измерять в пикселях), которое тело проходит за некоторую условную единицу времени.

Как моделировать движение тела? Повторять перемещение тела на заданное количество пикселей в течение заданного времени! При этом условной единицей времени будет один проход цикла.

Направление головы может изменяться внешними силами, которые представлены глобальной переменной **hbody**, скорость также задается извне глобальной переменной **vx**.



Метод равномерного движения
в течение времени `time`.

```
to moveua
repeat time [fd vx]
end
```

Конструктор движущегося тела.

```
to init-body :namet :pen :xbody :ybody
newturtle :namet st
make "penbody :pen
pu setx :xbody sety :ybody seth hbody setsh 3
set :namet "rule [launch [moveua]]
end
```

На основании информационной модели (таблица 1) опишем необходимые Конструкторы и Деструктор.

Конструктор установки.

```
to init-gravity :bx :by :h :w :c
newslider "hbody list :bx :by + :h
+ 50 [-90 90 90]
newslider "vx list :bx + :w / 3 :by
+ :h + 50 [-10 10 0]
newslider "time list :bx + :w * 2 /
3 :by + :h + 50 [0 99 0]
make "cbrd :c ;цвет ограды
рамка :bx :by :h :w [cancel [moveua]]
;рисуем ограду
init-body "uambody "pu :bx + 15 :by
+ 15
end
```

Конструктор для ограничивающей рамки, которая начинается с указанного места, имеет заданные высоту и ширину:

```
to рамка :bx :by :h :w :act
newturtle "w st setpensize 20
pu setx :bx sety :by
setc :cbrd pd
repeat 2[fd :h rt 90 fd :w rt 90]
set :cbrd "turtlerule :act
setpensize 1
```

```
axelx :bx + 15 :by :w + 10 ; ось с делениями
remove "w
end
```

Здесь объявлен виртуальный метод `act`, реализация которого зависит от вида движения. В этой задаче должен завершаться процесс равномерного прямолинейного движения, как только тело достигнет границы. Это описано параметром процедуры `рамка` в конструкторе `init-gravity`:

```
[cancel [moveua]].
```

Деструктор установки.

```
to destr-gravity
cg
everyone [remove who]
remove "vx
remove "hbody
set :cbrd "turtlerule []
end
```

Для измерения пройденного пути неплохо иметь измерительную линейку. Ее нарисуем с помощью вспомогательной процедуры `axelx`.

Программа рисования
оси с делениями.

```
to axelx x0 y0 long
make "dx 10 ;цена деления
pu setx :x0 sety :y0 pd setc 9
repeat :long / (:dx * 5) [sety ycor
+ 10 sety ycor - 20 sety ycor + 10
setx xcor + :dx
repeat 4[sety ycor + 5 sety
ycor - 10 sety ycor + 5
setx xcor + :dx]
]
pu setpos [-1,4 -90] setsh 7 pd stamp
end
```



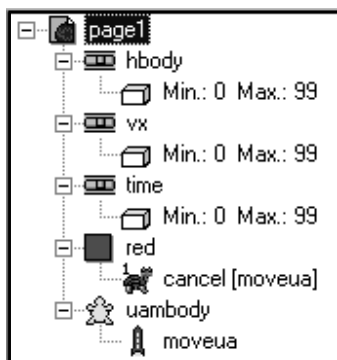


Рисунок 4

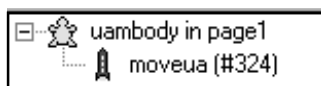


Рисунок 5

После выполнения следующей команды вы получите на текущей странице установку для проведения экспериментов:

```
init-gravity -340 -70 80 680 "red
```

Иерархию объектов нашей системы можно видеть в окне Project системы MicroWorld Pro (рисунок 4):

- на странице Page1 установка имеет границу красного цвета; при попадании любой Черепашки (событие Черепашки) на эту границу останавливается процесс движения **moveua**;
- внутри поля размещен экземпляр **uambody** объекта **Body**, который имеет собственный метод – «движение вперед» **uambody**;
- изменение свойств объекта **Body** осуществляется с помощью бегунков **vx** и **hbody**.

Щелкнув левой клавишей мыши на объект **uambody** (событие мыши), вы активизируете установку и сможете наблю-

дать процесс прямолинейного равномерного движения.

Используя окно Process, можно замедлить процесс и видеть, какие действия осуществляются в данный момент времени.

На рисунке 5 показан процесс движения тела **uambody**.

На полученной установке можно решать разнообразные задачи. Бегунки (глобальные переменные) позволяют менять скорость и время движения. Вы можете вычислить путь, который пройдет тело с заданной скоростью за определенное время, построить графики зависимости пути от времени.

Задания.

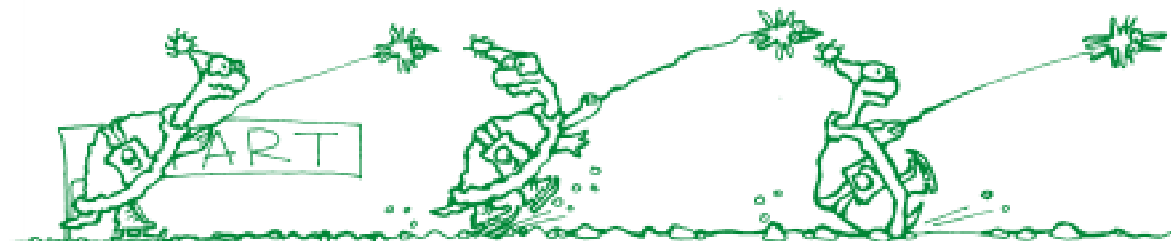
Устанавливая разные значения скорости, постройте в тетради графики зависимости пути от времени.

Рассчитайте, какой путь пройдет и где окажется тело, если оно в течение 20 условных единиц идет со скоростью 20 п/уе, затем 10 уе движется со скоростью 30 п/уе, далее движется со скоростью –10 п/уе в течение 10 уе.

Придумайте еще задачи, которые можно моделировать на нашей установке.

ПРЯМОЛИНЕЙНОЕ РАВНОУСКОРЕННОЕ ДВИЖЕНИЕ

В этом случае скорость движения тела изменяется со временем, причем за каждую условную единицу времени (уе) скорость изменяется на одно и то же число. Скорость изменения скорости называют ускорением. Для исследования прямолинейного равноускоренного движения модифицируем нашу экспериментальную



Объект	Свойства (поля)	Методы
Движущийся объект	Ускорение (av)	Изменять скорость
Показатель ускорения	Имя, положение на Рабочем поле, текущее значения (в пикселях)	Изменять значение
Кнопка для уменьшения ускорения	Имя, положение на Рабочем поле, имя метода, запускаемого щелчком мыши на кнопке	Вычитать 0.01 из текущего значения ускорения
Кнопка для увеличения ускорения	Имя, положение на Рабочем поле, имя метода, запускаемого щелчком мыши на кнопке	Прибавлять 0.01 к текущему значению ускорения
Кнопка для сброса показателей	Имя, положение на Рабочем поле, имя метода, запускаемого щелчком мыши на кнопке	Устанавливать значение скорости, ускорения и времени равным 0

Таблица 2

установку, добавив в нее еще одно свойство тела – ускорение и объект – показатель ускорения.

Для наглядности показатели времени, скорости, ускорения заменим текстовыми окнами. Для управления ускорением добавим кнопки. Это не обязательно, но полезно, так как с помощью кнопок можно задавать значение ускорения в разумных пределах. Установим также кнопку для сброса показателей в нулевое состояние.

Дополнение к информационной модели установки для моделирования движения приведено в таблице 2.

Метод равноускоренного движения.

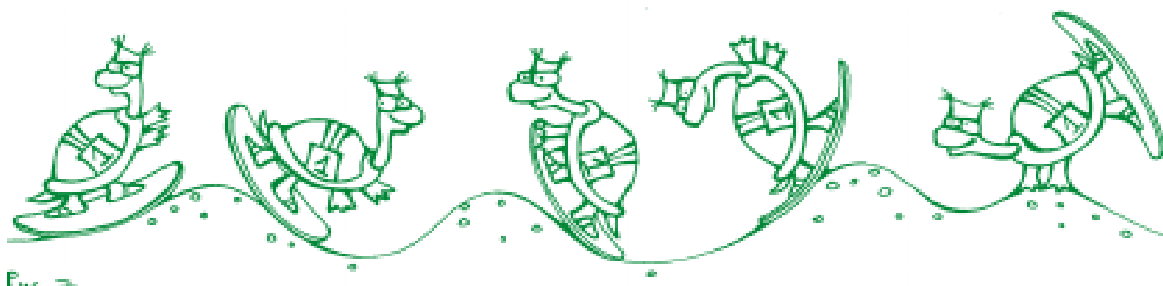
```
to movea
fd vx
setvx vx + ax
settime time + 1
movea
end
```

В этой процедуре движение тела моделируется рекурсивным алгоритмом, благодаря чему время движения не устанавливается, а вычисляется. Однако можно выбрать и другой способ моделирования, в зависимости от опыта.

Конструктор движущегося равноускоренно тела.

```
to init-abody :namet :pen :xbody :ybody
newturtle :namet st
make "penbody :pen
pu setx :xbody sety :ybody seth hbody
setsh 2
set :namet "rule [launch [movea]]
end
```

Эта процедура отличается от `init-body` только именем метода движения – `movea`.



Конструктор установки.

```
to init-agravity :bx :by :h :w :c
make "cbrd :c ;цвет ограды
newslider "hbody list :bx :by + :h
+ 50 [-90 90 90]
newtext "time list :bx + :w / 5 :by
+ :h + 60 [40 25]
settime 0
newtext "vx list :bx + :w * 4 / 10
:by + :h + 60 [60 25]
setvx 0
newbutton "ax- list :bx + :w * 5 /
10 :by + :h + 40 [ax-0.01]
set "ax- "size [50 25]
newtext "ax list :bx + :w * 5.8 / 10
:by + :h + 60 [40 25] ct insert 0
newbutton "ax+ list :bx + :w * 6.4 /
10 :by + :h + 40 [ax+0.01]
set "ax+ "size [50 25]
newbutton "resva list :bx + :w
* 8 / 10 :by + :h + 40 [reset]
set "ax+ "size [60 25]
рамка :bx :by :h :w [cancel
[movea]] ;рисуем ограду
init-abody "ambody "pu :bx +
25 :by + 25
end
```

Деструктор установки.

```
to destr-agravity
cg
everyone [remove who]
remove "ax
remove "ax-
remove "ax+
remove "vx
remove "resva
remove "time
remove "hbody
set :cbrd "turtlerule []
end
```

Кроме того, необходимы процедуры для методов кнопок для увеличения, уменьшения и установки в начальное положение свойств объекта.

Изменение значения ускорения в текстовом окне.

```
to ax-0.01
setax ax - 0.01
end
```

```
to ax+0.01
setax ax + 0.01
end
```

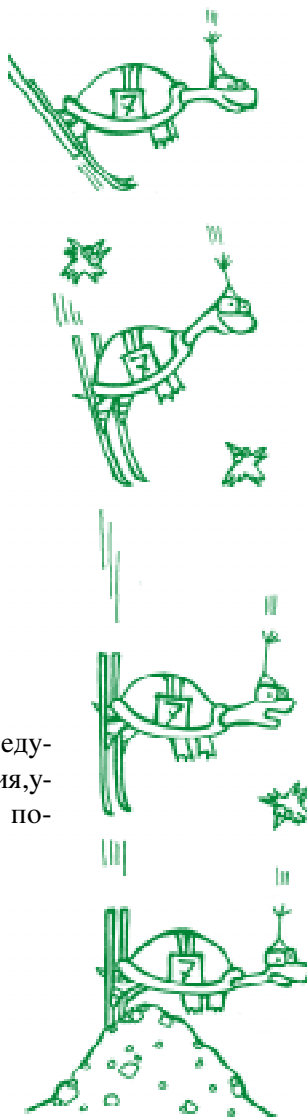
Сброс значений скорости и ускорения.

```
to reset
setvx 0
setax 0
settime 0
end
```

После выполнения следующей команды вы получите на текущей странице установку для проведения опытов:

```
init-agravity -340 -70 100 680 "blue
```

Задания.



Исследуйте, как зависит путь, пройденный телом, от времени при положительном ускорении, при отрицательном ускорении.

Придумайте еще задачи, которые можно моделировать на нашей установке.

В сборнике А.Г. Юдиной рассматриваются интересные задачи, связанные со сложением движений по координатным осям, с упругостью и так далее. Все их можно рассматривать в среде Лого Миры.

ПУБЛИКАЦИЯ НА WEB САЙТЕ

Если созданную обучающую среду опубликовать на сайте, то ее можно использовать в любом классе, имеющим выход на этот сайт.

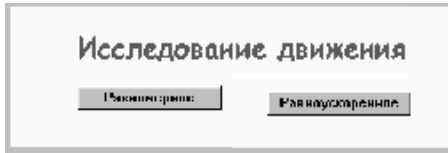


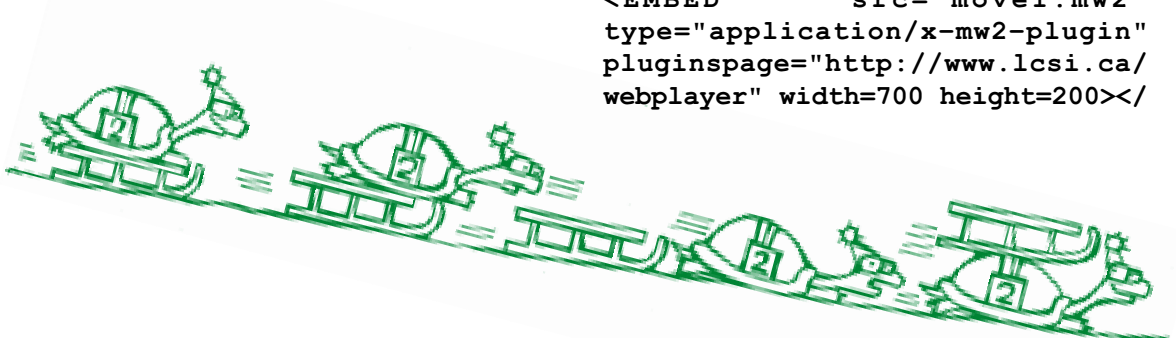
Рисунок 6

Однако перед публикацией следует провести подготовительную работу для осуществления навигации по нашей установке.

Мы создадим в нашем проекте еще одну страницу и разместим на ней титульный лист с двумя кнопками, позволяющими выбирать вид движения (рисунок 6). На листе с самой установкой разместим кнопку для возврата на титульную страницу.

Для запуска проекта и поддержки навигации на машине клиента подготовим следующие процедуры:

```
to startup
presentationmode p1
end
```



Метод для запуска установки равномерного движения.

```
to Равномерное
page1
init-gravity -340 -70 80 680 "red
make "finish "destr-gravity
end
```

Метод для запуска установки равноускоренного движения.

```
to Равноускоренное
page1
init-agravity -340 -70 100 680 "blue
make "finish "destr-agravity
end
```

Метод для возврата на титульную страницу.

```
to назад_
run :finish
p1
end
```

Перед созданием HTML документа на основной странице удалим все объекты установки (запустим деструктор) и сохраним в таком виде проект.

В системе MicroWorldPro создание HTML документа осуществляется просто выбором пункта меню File → Create HTML Template. При этом сохраняется с указанным именем файл следующего содержания:

```
<HTML>
<HEAD>
<META NAME="keywords" CONTENT="MW
Web Player">
<TITLE>move1</TITLE></HEAD>
<BODY>
<CENTER>
<EMBED src="move1.mw2"
type="application/x-mw2-plugin"
pluginspage="http://www.lcsi.ca/
webplayer" width=700 height=200></
```

```
CENTER>
</BODY>
</HTML>
```

ПОЯСНЕНИЯ К СЛОВАМ LOGO

В рассмотренном проекте не использовались какие-либо неизвестные встроенные команды или датчики LOGO. Однако в текстах процедур вы встретили слова, которые не входят в базовый язык и не описаны программами на листе процедур.

Речь идет о словах, включаемых в язык при создании новых объектов ко-

мандами типа `newtext`, `newbutton`, `newturtle`.

Рассмотрим команду

```
newslider "time list :bx + :w * 2 /  
3 :by + :h + 50 [0 99 0]
```

Создается бегунок (глобальная числовая переменная) с именем `"time`.

При этом появляются команда `settime <число>` для присваивания указанного значения переменной и датчик `time` для получения значения этой переменной.

Если указанный объект создать другой командой, например:

```
newtext "time list :bx + :w / 5 :by  
+ :h + 60 [40 25]
```

в виде текстового окна (это тоже глобальная переменная – текстовая или числовая, в зависимости от содержания), то так же, как в предыдущем случае, появятся команда `settime` и датчик `time`.

Так проявляется идея полиморфизма – метод называется и действует одинаково для объектов разных типов.

В данной статье мы хотели показать, что в среде Лого Миры учебные задачи и задачи моделирования ставятся и решаются на уровне знаний как учителя, так и ученика или любого другого исследователя. Именно поэтому можно предложить эту среду как инструмент творческого подхода к обучению и самообразованию.

Литература.

1. А.Г. Юдина, Практикум по информатике в среде LogoWriter. Учебное пособие для общеобразовательной школы.
2. Документация на системы Лого Миры 2.04 и MicroWorld Pro.

Эксклюзивным поставщиком русской версии программы ЛогоМиры 2.0 является Институт новых технологий образования: тел. (095) 915-6121; int@mtu-net.ru; www.school.edu.ru/int.



Наши авторы, 2001.
Our authors, 2001.

*Кузнецова Ирина Николаевна,
учитель высшей категории,
директор НОУ ДО
«Папертовский центр»,
Санкт-Петербург*