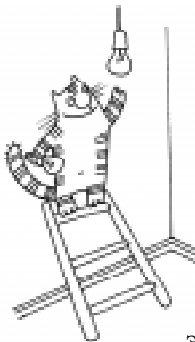


ШКОЛА МОДЕЛИРОВАНИЯ-2003. ЗАНЯТИЕ 2. МОДЕЛИРОВАНИЕ ДВИЖЕНИЯ ОБЪЕКТА В СРЕДЕ MVS

При объектном подходе к исследованию окружающего мира очень важно уметь выделять некоторый объект и решать вопрос о том, рассматривать ли его как нечто целое или как систему. При рассмотрении объекта как системы необходимо выделить в ней составляющие элементы, описать их свойства и поведение, а также установить связи между ними.

В задачнике [2] предложены задачи, в которых исследуется один объект, а также задачи на исследование системы объектов.

Рассмотрим, как решается задача о падении лестницы [2, стр. 79] с позиций выполнения этапов моделирования, предложенных в учебнике [1].



1 ЭТАП. ПОСТАНОВКА ЗАДАЧИ

ОПИСАНИЕ ЗАДАЧИ

Электрик Петров поставил к стене лестницу и, поднявшись вверх, остановился на одной из ступенек. В

это время концы лестницы начали скользить вдоль стены и пола. Провести исследование, по какой кривой будет падать вниз электрик Петров.

Комментарий. Как видите, в реальной жизни задачи формулируются не так, как в учебнике. Описывается ситуация в самом общем виде. Такие задачи называются «непоставленные»



(или слабо формализованные). В отличие от предметных задач по физике, математике или химии, при формулировке задачи намеренно не выделяется исследуемый объект, не указываются параметры объекта, которые будут использоваться при исследовании, и не задаются их исходные значения. Отобрать эти параметры предлагается самому ученику после определения целей моделирования. Общая направленность исследования формулируется в виде проблемного вопроса «по какой кривой будет падать электрик?»

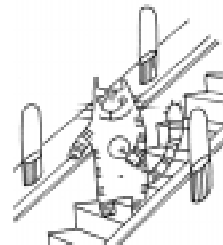
Однако задача возникает не на пустом месте, а на основе первоначальных наблюдений.

Почему здесь возникает необходимость исследования? Да потому, что, на первый взгляд, может показаться, что он будет двигаться вниз по вертикали. Однако первоначальное наблюдение позволяет сделать вывод: нет, не по вертикали. Потому что, если предположить, что в начале движения лестница имела вертикальное положение, то электрик находился у самой стены, на какой бы ступеньке он ни стоял. А когда лестница упала, то он явно уже был «далеко» от стены, на расстоянии ступеньки от верхнего конца.

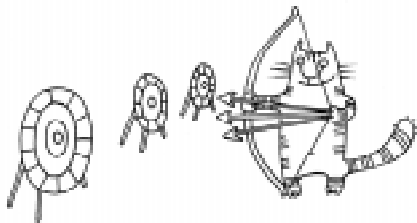
ЦЕЛИ МОДЕЛИРОВАНИЯ

Исследовать движение ступеньки лестницы, на которой стоит электрик.

Комментарий. Основой осмысленного поведения является целенаправленная деятель-



ность. Именно поэтому во всяком исследовании должна быть сформулирована цель. Цель может быть одна, а может быть и несколько. Может быть главная цель и несколько подцелей.



ФОРМАЛИЗАЦИЯ ЗАДАЧИ

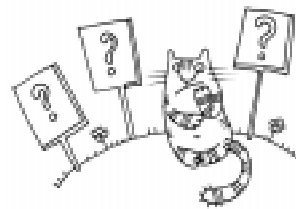
Комментарий. Формализация задачи – необходимый шаг, который проходит исследователь на пути разработки и исследования модели (а на самом деле, объекта). Результатом формализации является построение информационной модели, по которой можно составить формальное описание задачи.

Поскольку информационная модель – это целенаправленно отобранная информация, представленная в некоторой форме, то следует рассматривать формализацию как отбор этой информации в соответствии с выдвинутой целью и выбор формы ее представления. При формализации задачи прежде всего необходимо уяснить, что дано, что надо найти и каковы правила преобразования исходных данных в результат. А для этого, отталкиваясь от общего описания задачи, надо выделить прототип моделирования и, опираясь на цели моделирования, решить вопрос, рассматривать его как целостный объект или как систему. Если моделируется система, то производится ее анализ: выявляются составляющие системы (элементарные объекты) и определяются связи между ними. При анализе необходимо также решить вопрос о степени детализации системы.

После определения прототипа необходимо определить, какие характеристики объекта будут учитываться при моделировании. Это вопрос непростой. Если отбросить существенные факторы, то модель будет неверно отражать оригинал (прототип). Если оставить слишком много, модель будет сложна для построения и исследования.

Методически формализацию проводят в виде поиска ответов на вопросы, уточняющие общее описание задачи:

- Что моделируется?
- Какие параметры моделируются?



- Какие параметры известны и какие надо определить?
- Возможный диапазон значений.
- Отношения и связи (для систем).
- Установить правила (формулы) преобразования исходных данных в результат.

Проведем формализацию задачи в виде поиска ответов на следующие вопросы:

- Что моделируется?

Объект «лестница», который представляет систему, состоящую из ступенек.

- Какие параметры лестницы известны?

Длина L , количество ступенек N , угол φ , образуемый лестницей и стеной.

- Как расположены ступеньки на лестнице?

На одинаковом расстоянии. Верхняя ступенька совпадает с верхним концом лестницы.

По точкам.

- Каково начальное положение лестницы?

Реально она должна быть отклонена на некоторый угол φ_0 . Но для простоты можно считать, что она имеет вертикальное положение ($\varphi_0 = 0$).

- Как движется лестница?

Концы скользят вдоль стены и пола, угол φ изменяется от 0 до 90° .

- Что надо определить?

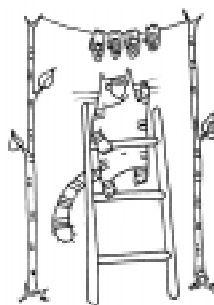
Кривую, по которой движется ступенька (и вместе с ней электрик Петров).

- Какие параметры ступеньки известны?

Номер ступеньки k , на которой стоит электрик Петров. Для определенности будем считать, что ступеньки перенумерованы снизу вверх.

- Как строится кривая?

По точкам.





• Чем характеризуется точка кривой?

Координатами x и y , которые определяются в некоторой заданной системе координат и связаны с углом отклонения лестницы.

• Как определяются координаты?

Через равные промежутки изменения угла φ ($\Delta\varphi$), h_φ – шаг изменения угла

должен быть задан; по сути, это параметр, характеризующий отдельные моменты процесса падения лестницы (дискретизацию процесса).

В процессе ответов на поставленные вопросы выяснилось, что:

– в задаче рассматривается объект «Лестница» как система и объект «Ступенька» как элемент системы «Лестница»;

– объект «Электрик» не требует рассмотрения;

– в соответствии с поставленной целью моделирования, информационную модель надо разработать для объекта «Ступенька».

По итогам формализации можно описать задачу в формальной постановке:

Верхний конец A лестницы длиной L прислонен к вертикальной стене, а ее нижний конец B расположен на горизонтальной

поверхности. На лестнице N ступенек, расположенных на одинаковом расстоянии d . Первая ступенька расположена на расстоянии d от точки B , последняя – на верхнем конце A . Трение отсутствует. В начальный момент времени лестница имела угол наклона к стене равный φ_0 . Далее она начинает падать так, что верхний конец скользит по стене, а нижний – по горизонтальной поверхности. Построить траекторию движения точки (x, y) , соответствующей k -той ступеньке в координатах (X, Y) , где X – горизонтальная ось, направленная вдоль поверхности земли, а Y – вертикальная ось, направленная вдоль стены вверх.

2 ЭТАП. РАЗРАБОТКА МОДЕЛИ

ИНФОРМАЦИОННАЯ МОДЕЛЬ

Результаты формализации задачи сведены в таблицу 1 ([2], с. 79).

В этой таблице отдельно представлены параметры, характеризующие лестницу как единый целостный объект, и отдельно параметры ступеньки как элемента системы. Информационная модель ступеньки как

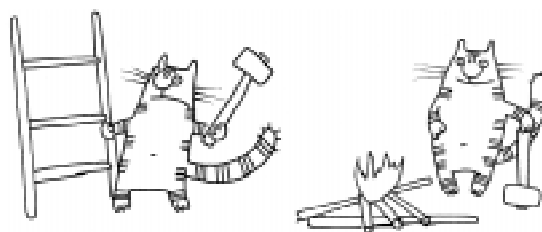


Таблица 1.

Объект	Параметры		Действия
	Название	Значения	
Лестница (система)	Длина L	Исходные данные	Скольжение концов вдоль стены и пола
	Количество ступенек N	Исходные данные	
	Угол наклона φ	Расчетные данные	
	Шаг изменения угла $\Delta\varphi$	Исходные данные	
	Расстояние между ступеньками d	Расчетные данные	
Ступенька (элемент системы)	Номер ступеньки на которой стоит электрик k	Исходные данные	Изменение положения
	Расстояние до нижнего конца	Расчетные данные	
	Расстояние до верхнего конца	Расчетные данные	
	Координата x	Результат	
	Координата y	Результат	

объекта, являющегося частью целого, автоматически включает и параметры лестницы в целом.

Данная задача предполагает построение еще одной формы информационной модели, а именно математической модели – совокупности формул, связывающих параметры процесса и описывающих события.

МАТЕМАТИЧЕСКАЯ МОДЕЛЬ

Для построения математической модели воспользуемся чертежом.

Так как верхний конец А скользит вдоль стены, то у него изменяется только координата y от L до 0 , а координата x всегда равна 0 , а для конца В наоборот – x изменяется от 0 до L , $y = 0$. У промежуточных точек изменяются обе координаты.

Вычислим координаты ступеньки с номером k (рисунок 1), которая располагается в точке С.

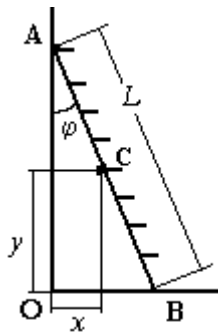


Рисунок 1.

Очевидно, что если менять угол от начального значения до 90 градусов, то мы получим искомую траекторию. Это так называемое параметрическое задание кривой. Для построения кривой необходимо задавать значение угла, вычислять координаты x и y и отмечать полученную точку на плоскости. Чем больше вычислено точек траектории, тем легче будет ее нарисовать. Если кривая строится руками, то вводят еще один вспомогательный параметр M – число расчетных точек траектории. Опорные точки (x_i, y_i) , вычисленные при следующих значениях углов

$$\varphi_i = \varphi_{i-1} + h_\varphi,$$

где $i = 1, 2, 3, \dots$; $\varphi_0 = 0$; $h_\varphi = \frac{90}{M}$, затем соединяются, например, ломаной.

Угол начального (вертикального) положения лестницы обозначим $\varphi_0 = 0$.

КОМПЬЮТЕРНАЯ МОДЕЛЬ

При построении компьютерной модели одним из первых шагов является выбор программной среды. В задачнике [2] предлагается использовать среду табличного процессора. Рассмотрим, как эта модель строится и исследуется в среде MVS. На прошлом занятии [3] мы, описывая объект, выделяли не просто всю совокупность параметров, характеризующую состояние объекта в некоторый момент, а разбивали их в группы по следующим соображениям:

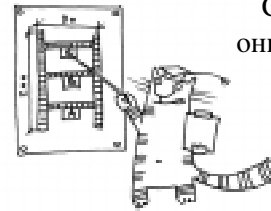
Константы – это параметры, не меняющиеся для данного класса.

Параметры – это параметры, не меняющиеся для данного экземпляра класса.

Вспомогательные расчетные переменные – это параметры, также не меняющиеся для данного экземпляра класса, но получаемые на основании расчета по формулам.

Переменные – это параметры, меняющиеся для данного экземпляра класса и характеризующие поведение объекта.

Составим информационную модель класса Ступенька в терминах языка MVL для разработки и исследования модели среды MVS (таблица 2).



Таким образом, здесь более детально описано поведение объекта, которое характеризуется двумя длительными действиями – падение и покой после приземления, а также событие – условие перехода из одного длительного действия в другое.

Замечание. В работе [2] и языке MVL существуют несущественные терминологические различия при описании классов. Класс в MVL это элемент языка моделирования, и, соответственно, при его описании используются термины, принятые в языках программирования. Характеризуя объект и его свойства, мы используем переменные, которые

Таблица 2.

Объект (класс) – Ступенька (элемент системы Лестница).

Параметры	
Название	Значения
Константы	нет
Параметры	
Длина L	Исходные данные
Количество ступенек N	Исходные данные
Шаг изменения угла h_ϕ	Исходные данные
Номер ступеньки	Исходные данные
Вспомогательные (внутренние)	
Расстояние между ступеньками d	Расчетные данные
Расстояние до верхнего конца	Расчетные данные
Расстояние до нижнего конца	Расчетные данные
Поведение	
1. Изменение положения	
Угол наклона ϕ	Расчетные данные
Координата x	Расчетные данные
Координата y	Расчетные данные
2. Неподвижное положение (покой) лестницы после приземления	Окончание процесса
Событие	
Приземление ступеньки (и лестницы в целом) – условие перехода из одного длительного действия в другое	Угол больше 90°

могут принимать значения различного типа. В данной задаче использованы только вещественные и целочисленные переменные. Сами переменные, в свою очередь, могут быть разделены на:

- константы (никогда не меняют своего значения);
- параметры (могут менять свои значения от экземпляра класса к экземпляру, оставаясь постоянными для данного экземпляра);
- вспомогательные переменные, позволяющие избегать сложных математических выражений, выделяя из них небольшие, связанные с сутью задачи, подвыражения;
- и, наконец, основные переменные, с помощью которых мы описываем конкретные свойства объекта.

Вспомогательные и основные переменные в MVL называются внутренними пере-

менными. Как мы видим, переменные могут сохранять свои значения (статические переменные или неменяющиеся свойства объекта) – константы, параметры, отдельные вспомогательные или основные переменные – или менять их во времени – вспомогательные и основные переменные. В данном случае – ВС и АС это внутренние переменные, сохраняющие свои значения для конкретного экземпляра объекта, а x и y – переменные, меняющие свое значение во времени.

Построение компьютерной модели можно представить в виде нескольких последовательных шагов.

Шаг 1. Создание нового проекта.

Запустите среду MVS, создайте новый проект Лестница 79¹ и в нем класс «Ступенька» (рисунок 2).

Шаг 2. Вводим и описываем все необходимые параметры (рисунок 3).

В разделе Параметры задаем начальные значения параметров, которые не меняются в течение процесса падения конкретного экземпляра лестницы.

В разделе Внутренние переменные описываем формулы вычисления значений вспомогательных параметров, также не ме-

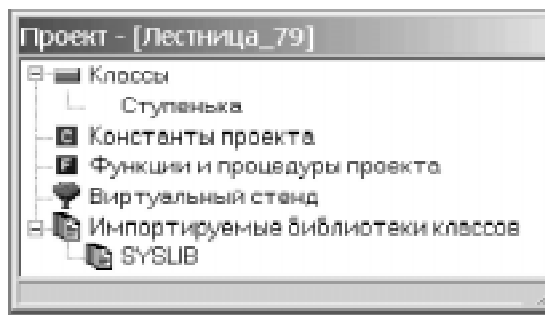


Рисунок 2. Новый проект.

¹ Название проекта связано с задачей из задачника [2], приведенной на странице 79.

няющиеся в течение всего процесса. Здесь же определены начальные значения переменных x и y , описывающих поведение объекта. Им присвоено значение 0.

Обратите внимание. Среда не требует расположения формул расчета вспомогательных переменных в логической последовательности, что отражено на рисунке.

Шаг 3. Построение карты поведения (рисунок 4).

На рисунке изображены два узла длительного поведения. В первом узле, названном INIT, реализуется длительное действие – плавное изменение угла в заданных пределах и вычисление координат ступеньки. Длительные действия выполняются (DO) в соответствии с уравнениями, названными Система_уравнений_1. Этот узел помечен как начальный (в него ведет маленькая стрелка). Во втором узле, в разделе входные действия (Entry actions), описана команда останова процесса. Обратите внимание на то, что все длительные действия описаны с помощью уравнений, а входные (мгновенно выполняемые) – это алгоритмические операторы. В данном случае использована только одна команда – Stop. Стрелка, соединяющая два узла или длительных состояния, показывает, что при наступлении события (угол превысил значение 90 градусов) необходимо прекратить длительные действия первого узла и перейти во второй узел.

Карту поведения необходимо создавать самому, как и вводить нужные уравнения. Первоначально карта содержит только один узел Init с предписанными ему по умолчанию уравнениями – Система_уравнений_1. Добавление новых узлов и стрелок (переходов) осуществляется с помощью кнопок, расположенных слева. Кнопки, позволяющие описывать события, мгновенные, входные и выходные действия, расположены на верхней панели.

Шаг 4. Выписываем формулы, необходимые для вычисления (рисунок 5).

Так как MVS умеет плавно менять только одну независимую переменную – время, то мы полагаем $\varphi = \text{rad}(\text{Time})$. Здесь

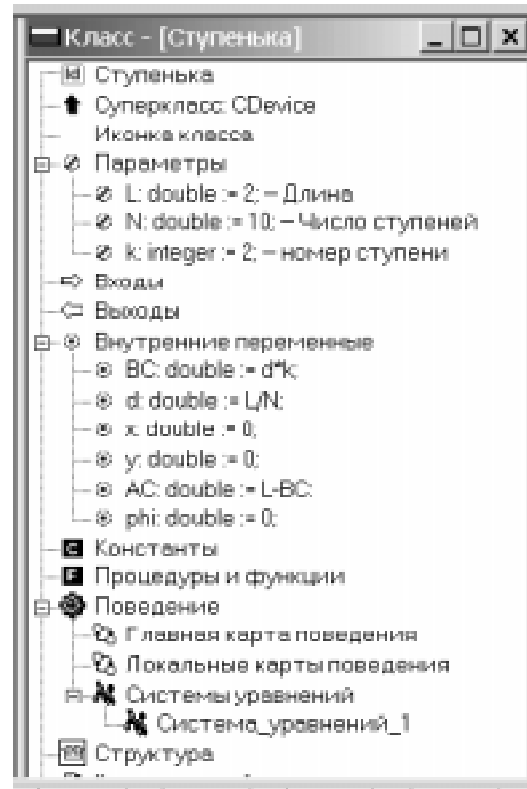


Рисунок 3. Описание переменных класса.

Time – имя специальной переменной типа double, которую мы трактуем как время. Эта внутримашинная переменная с очень маленьким шагом изменения, таким маленьким, что пользователю будет казаться, что процесс непрерывный. Использование переменной *Time* избавляет от введения в качестве исходного параметра моде-

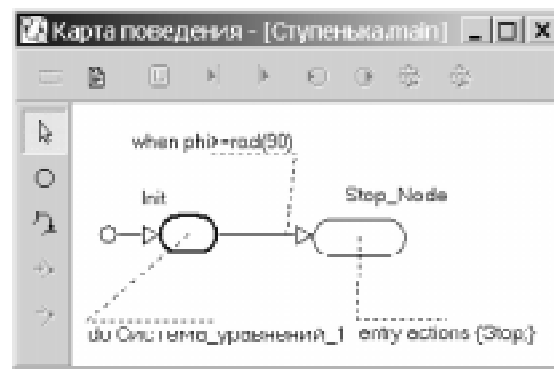
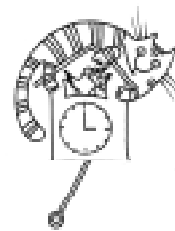


Рисунок 4. Карта поведения.

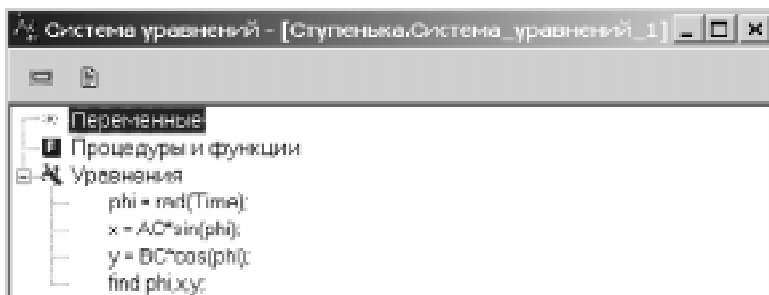


Рисунок 5. Формулы, определяющие координаты точки.

лирования шага изменения угла h_φ (или числа расчетных точек M). Функция $Rad(Time)$ возвращает текущее значение времени в радианах. Во втором узле движение (течение времени) прекращается, и мы можем проанализировать построенные траектории. Прекращение течения времени связано с событием – достижением значения $Time \geq 90$, или $\varphi \geq rad(90) = \pi/2$.



Шаг 5. Планирование будущего вычислительного эксперимента.

В редакторе моделей есть еще одно окно – прообраз испытательного стенда. На этом макете испытательного стенда можно одновременно разместить несколько экземпляров классов с различными исходными данными. Проведем испытания с двумя экземплярами. Размещаем на Испытательном стенде два экземпляра класса с различными значениями параметров. В начале работы в окне «Испытательный стенд» был автоматически размещен экземпляр класса «Ступенька_1». Для того чтобы поместить туда второй экземпляр, достаточно с помощью операции **Drag_and_Drop** «перенести» класс «Ступенька» из окна «Проект» в окно «Испытательный стенд». Параметры экземпляра «Ступенька_1» совпадают с параметрами, заданными по умолчанию. Изменить их во втором классе «Ступенька_2» можно с помощью диалога, открывающегося при нажатии правой кнопки мыши на образе экземпляра класса. Активизируйте диалог. Логично было бы рассмотреть две ступеньки одной лестницы. Одну взять ближе в нижнем конце

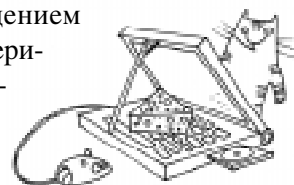


($k = 2$), а другую – ближе к верхнему концу ($k = 8$), а значения длины лестницы и количества ступенек не менять. Найдите нужный параметр. Введите новое значение. Нажмите клавишу **Enter** (введенное значение будет отображено более ярко). Закройте диалог.

3 ЭТАП.

КОМПЬЮТЕРНЫЙ ЭКСПЕРИМЕНТ.

Перед проведением компьютерного эксперимента необходимо откомпилировать модель. Для этого в открывшемся окне «Испытательный



стенд» выполните команду **СТАРТ**. Начнется первый длительный процесс, и в двух окнах будут строиться диаграммы (рисунок 5). Когда наступит событие, соответствующее падению лестницы, построение графиков закончится. Именно для автоматической остановки эксперимента и был предусмотрен второй узел с единственной выполняемой командой **STOP**. Если бы мы этого не предусмотрели, нам бы пришлось останавливать эксперимент самим, используя команду **STOP** (кнопка с двумя коричневыми вертикальными чертами).



На рисунке 6 показан фрагмент окна «Испытательный стенд» с тремя окнами: «Ступенька_1» («живая таблица»), «Временная диаграмма» и «Фазовая диаграмма» для экземпляра класса второй ступеньки $k = 2$.

Окно «живая таблица» содержит значения всех переменных на момент окончания эксперимента. На временной диаграмме строятся значения переменных x и y как функций от времени. Горизонтальная ось – время, вертикальная – координаты x и y . На фазовой диаграмме строится зависимость $y(x)$. Построение этой кривой и является целью моделирования.

ТЕСТИРОВАНИЕ МОДЕЛИ

Анализ временной диаграммы показывает, что координата y в начальный момент была ненулевая, а x равна 0, что соответствует ситуации, когда лестница стоит вертикально у стены. И наоборот, в конечный момент $x \neq 0$, а $y = 0$, что соответствует ситуации, когда лестница лежит на полу.

Чтобы убедиться, что расчеты выполнены верно, надо использовать тестовый пример с заранее известными результатами. Так, например, ступенька, расположенная на верхнем конце лестницы A , будет двигаться по вертикали, то есть для нее в течение процесса координата x равна 0, а y уменьшается от L до 0.

Установите параметр $k = 10$ для одного из экземпляров на испытательном стенде, откомпилируйте модель заново и убедитесь, что модель правильно отражает процесс.

ПРОВЕДЕНИЕ ЭКСПЕРИМЕНТОВ

Проведите несколько компьютерных экспериментов с различными значениями исходных параметров. Например, посмотрите, какой вид будет иметь траектория движения ступеньки, расположенной на середине лестницы.



Чтобы более четко представлять себе траекторию движения, желательно установить одинаковые шкалы для вертикальной и горизонтальной осей на фазовой диаграмме.

ДОМАШНЕЕ ЗАДАНИЕ

А) Спроектируйте «Испытательный стенд» так, чтобы можно было менять номер ступеньки непосредственно перед началом эксперимента.

Б) Измените модель так, чтобы лестница в начальный момент времени стояла не вертикально, а под некоторым заданным углом к стене. (Подсказка. Вместо изменения угла, мы меняем равномерно и

непрерывно время. Время всегда начинается с нулевого отсчета. Поэтому мы должны изменить формулу вычисления угла $\varphi = rad(\varphi_0 - Time)$ и условие для события, соответствующего приземлению лестницы ($\varphi_0 + Time \geq 90$).

Рассмотренный пример уже содержит многие черты событийно-зависимого поведения. Так, наш алгоритм может быть записан в алгоритмической форме

До тех пор пока (While)

угол не достиг 90 градусов (Условие)

Выполняй

Вычисляй значения координат и рисуй график (Длительное действие)

Иначе

Прекрати вычисления (Длительное действие «ничего не делай»).

Если назвать выполнение условия «угол не достиг 90 градусов» – событием, то наш алгоритм может быть записан еще короче: смени длительное действие «Вычисляю координаты точки» на новое действие «Ничего не делай», как только наступит событие «Угол стал больше либо равным 90 градусов». Именно такой стиль описания и присущ языку MVL.

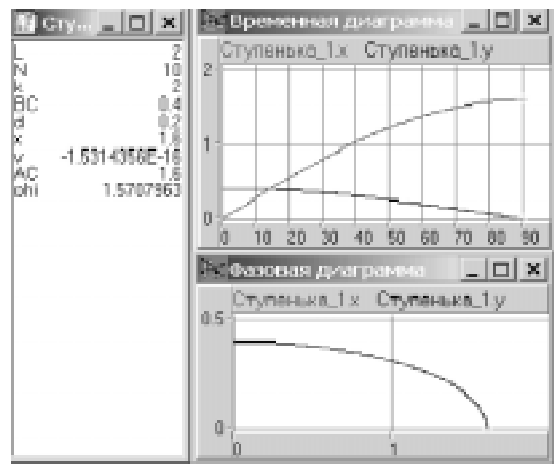


Рисунок 6. Временная и фазовая диаграммы движения точки.

Литература.

1. Информатика. 7–9 класс. Базовый курс. Теория. / Под ред. *Н.В. Макаровой*. СПб.: Питер, 2001.
2. Информатика. 7–9 класс. Базовый курс. Практикум-задачник по моделированию. / Под ред. *Н.В. Макаровой*. СПб.: Питер, 2000–2003.
3. *Ю.Б. Колесов, Ю.Б. Сениченков*. Шкомод-2003. Занятие 1. Компьютерные инструменты в образовании, № 1, 2003.



*Макарова Наталья Владимировна,
доктор педагогических наук,
кандидат технических наук,
проректор Международного
банковского института
(Санкт-Петербург), заведующий
кафедрой информационных систем
и технологий,*

*Сениченков Юрий Борисович,
доцент кафедры Распределенных
Вычислений и Компьютерных
Сетей Санкт-Петербургского
Политехнического Университета,*

*Титова Юлияна Францевна,
канд. пед. наук, доцент кафедры
информационных систем и
технологий МБИ.*



Наши авторы, 2003.
Our authors, 2003.