

Лабораторная работа № 5

Тема: Математическая модель физического явления. Анализ физического опыта.

Цель работы:

- научиться исследовать и описывать предметную область моделирования;
- научиться решать и описывать математическую модель поставленной задачи;
- научиться выбирать вид визуализации данных в зависимости от предметной области;
- научиться использовать процедуры и функции для решения задачи математического моделирования физического опыта.

Краткие теоретические сведения

Математическая модель - математическое представление реальности, один из вариантов модели как системы, исследование которой позволяет получать информацию о некоторой другой системе, это формулы, системы, неравенства.

Математическая модель — это «эквивалент» объекта, отражающий в математической форме важнейшие его свойства — законы, которым он подчиняется, связи, присущие составляющим его частям.

Все современные разделы физики посвящены построению и исследованию математических моделей различных физических объектов и явлений.

Так физики - «ядерщики» до проведения экспериментов выполняют серьезные исследования с применением математических моделей.

Идут активные работы по созданию математических моделей в экологии.

Предвидеть ответные реакции системы на действие конкретных факторов можно лишь через сложный анализ существующих в ней количественных взаимоотношений и закономерностей. В экологии поэтому широкое распространение получил метод **математического моделирования** как средство изучения и прогнозирования природных процессов.

Суть метода заключается в том, что с помощью математических символов строится абстрактное упрощенное подобие изучаемой системы. Затем, меняя значение отдельных параметров, исследуют, как поведет себя данная искусственная система, т. е. как изменится конечный результат.

Огромную роль играет использование математических моделей в медицине.

Медицинская кибернетика - научное направление, связанное с проникновением идей, методов и технических средств кибернетики в медицину. Развитие идей и методов кибернетики в медицине осуществляется в основном в направлениях создания диагностических систем для различных классов заболеваний с использованием универсальных или специализированных ЭВМ; создания автоматизированного электронного медицинского архива; разработки математических методов анализа данных обследования больного; разработки метода математического моделирования на ЭВМ деятельности различных функциональных систем; использования математических машин для оценки состояния больного.

Основные этапы математического моделирования

Построение модели.

На этом этапе задается некоторый «нематематический» объект — явление природы, конструкция, экономический план, производственный процесс и т. д. Сначала выявляются основные особенности явления и связи между ними на качественном уровне. Затем найденные качественные зависимости формулируются на языке математики, то есть строится математическая модель.

Решение математической задачи, к которой приводит модель.

На этом этапе большое внимание уделяется разработке алгоритмов и численных методов решения задачи на ЭВМ, при помощи которых результат может быть найден с необходимой точностью и за допустимое время.

Проверка адекватности модели.

На этом этапе выясняется, согласуются ли результаты эксперимента с теоретическими следствиями из модели в пределах определенной точности.

Визуализация полученных данных.

Построение графиков, диаграмм или создание других видов графических представлений на основе решенной математической задачи.

Интерпретация полученных следствий из математической модели.

Следствия, выведенные из модели на языке математики, интерпретируются на языке, принятом в данной области.

Модификация модели.

На этом этапе происходит либо усложнение модели, чтобы она была более адекватной действительности, либо ее упрощение ради достижения практически приемлемого решения.

Для реализации некоторого количества опытов нам понадобится использовать для упрощения кода такое **понятие как процедура**.

Долгое время процедуры и функции играли не только функциональную, но и архитектурную роль.

Весьма популярным при построении программных систем был метод функциональной декомпозиции "сверху вниз", и сегодня еще играющий важную роль.

Но с появлением ООП архитектурная роль функциональных модулей отошла на второй план. Для ОО-языков, к которым относится и язык C#, в роли архитектурного модуля выступает класс.

Программная система строится из модулей, роль которых играют классы, но каждый из этих модулей имеет содержательную начинку, задавая некоторую абстракцию данных.

Процедуры и функции связываются теперь с классом, они обеспечивают функциональность данных класса и называются методами класса. Главную роль в программной системе играют данные, а функции лишь служат данным. Напомню здесь, что в C# процедуры и функции существуют только как методы некоторого класса, они не существуют вне класса.

В данном контексте понятие класс распространяется и на все его частные случаи - структуры, интерфейсы, делегаты.

В языке C# нет специальных ключевых слов – **procedure** и **function**, но присутствуют сами эти понятия. Синтаксис объявления метода позволяет однозначно определить, чем является метод -процедурой или функцией.

Прежнюю роль библиотек процедур и функций теперь играют библиотеки классов. Библиотека классов FCL (статическая компонента Framework .Net.), доступная в языке C#, существенно расширяет возможности языка. Знание классов этой библиотеки и методов этих классов совершенно необходимо для практического программирования на C# с использованием всей его мощи.

Процедуры и функции. Отличия

Функция отличается от процедуры двумя особенностями:

- всегда вычисляет некоторое значение, возвращаемое в качестве результата функции;
- вызывается в выражениях.

Процедура C# имеет свои особенности:

- возвращает формальный результат `void`, указывающий на отсутствие результата;
- вызов процедуры является оператором языка;
- имеет входные и выходные аргументы, причем выходных аргументов- ее результатов - может быть достаточно много.

Хорошо известно, что одновременное существование в языке процедур и функций в каком-то смысле избыточно. Добавив еще один выходной аргумент, любую функцию можно записать в виде процедуры.

Мы пока не будем вдаваться подробности описания классов и методов в полной мере, но попробуем создать метод (процедуру) для класса `Form`, который будем использовать как блок команд при повторяющихся входящих данных для решения одной и той же математической задачи.

Ход работы:

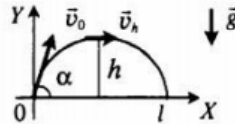
1. Описать физическое явление (процесс, задачу) с помощью определений и формул.

Что такое движение тела брошенного под углом к горизонту?

Определение

Движением тела под углом к горизонту в физике называют – _____

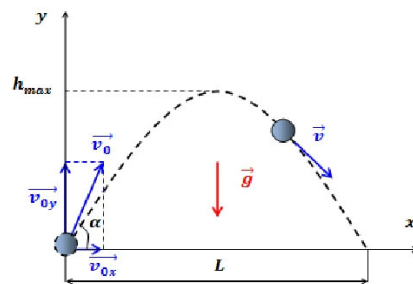
В процессе подбрасывания объекта вверх под углом к горизонту наблюдают _____



На графике изображено схематичное движение тела, которое подбросили под углом к горизонту.

В этом случае α является углом, под которым объект начал свое перемещение со скоростью v_0

2. Решить физическую задачу, так, чтобы можно было явно выделить типы данных, описать переменные для ввода-вывода данных. Описать алгоритм решения задачи для дальнейшего программирования.

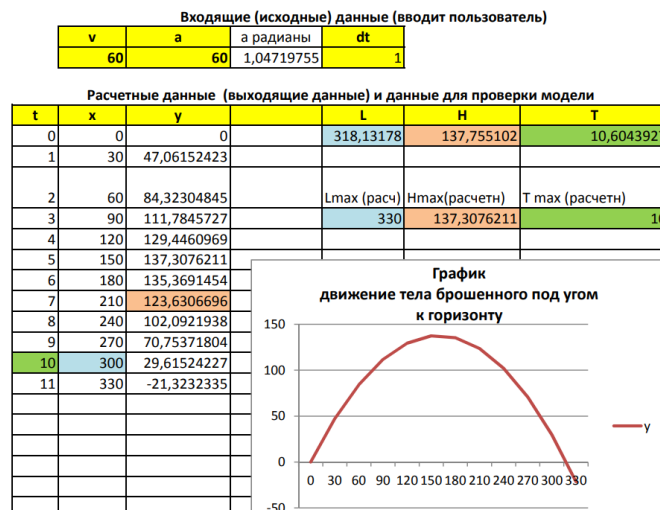


Формулы:

1. $t_{\text{пол}} = \frac{2V_0^2 \times \sin \alpha}{g}$
2. $t = \frac{V_0 \times \sin \alpha}{g}$
3. $L = \frac{V_0^2 \times \sin 2\alpha}{g}$
4. $H_{\text{max}} = \frac{V_0^2 \times \sin^2 \alpha}{2g}$
5. $v = \sqrt{V_0^2 \cos^2 \alpha + (V_0 \sin \alpha - gt)^2}$
6. $tg \varphi = \frac{V_y}{V_x}$

• Это движение в плоскости, поэтому для описания движения необходимо 2 координаты.

3. Проверить адекватность модели с помощью решения задачи с разными вариантами данных (провести несколько опытов). Если необходимо, то использовать дополнительные средства (например Excel)



4. Реализовать алгоритм задачи в среде программирования с использованием пользовательской формы. Можно использовать уже построенную форму предыдущей лабораторной работы, как шаблон приложения изменив, модуль ввода-вывод данных и функцию вычисления.

```
double g = 9.8;
double pi = 3.1415;
double v = Convert.ToInt32(textBox1.Text);
double dt = Convert.ToInt32(textBox3.Text);

double alpha = Convert.ToInt32(textBox2.Text) * pi / 180;
double T = Math.Round(2 * v * Math.Sin(alpha) / g, 2);
double S = Math.Round(v * v * Math.Sin(2 * alpha) / g, 2);
double H = Math.Round(v * v * Math.Sin(alpha) * Math.Sin(alpha) / (2 * g), 2);
```

5. Реализовать визуализацию данных средствами среды программирования. При правильном использовании шаблона предыдущей работы, для новых данных визуализация должна построиться автоматически.

6.

Form1

Движение тела, брошенного под углом к горизонту

tabPage3 tabPage1 tabPage2 tabPage4

Ввод данных:

Введите скорость $v =$ 60

Введите угол $\alpha =$ 60

Введите точность $\epsilon =$ 1

Посчитать

Уравнение координаты записывают в следующем виде:

$$x = v_0 \cos \alpha \times t$$

$$y = v_0 \sin \alpha \times t - \frac{gt^2}{2}$$

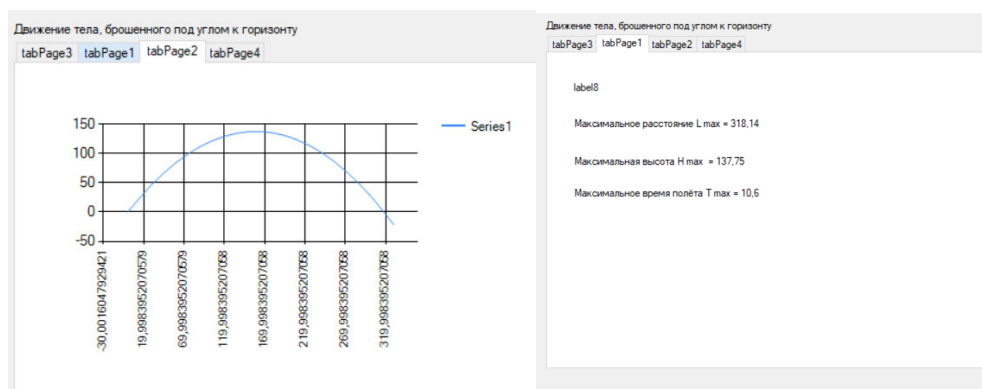
Полет тела будет длиться определенное время,

$$t = \frac{2v_0 \sin \alpha}{g}$$

Максимальная высота:

$$h_{max} = \frac{v_0^2 \sin^2 \alpha}{2g}$$

Максимальная дальность полета составит:



7. * Модернизировать задачу для проведения нескольких опытов с помощью использования процедуры. *(повышенной сложности)

Задание:

Создайте математическую модель физического процесса и визуализируйте опыт с помощью графика функции и таблицы значений функции.

1. Вычислить время полета тела, брошенного вверх под углом к горизонту в зависимости от его начальной скорости и угла бросания.
2. Вычислить длину полета тела, брошенного вверх под углом к горизонту в зависимости от его начальной скорости и угла бросания.
3. Вычислить высоту полета тела, брошенного вверх под углом к горизонту в зависимости от его начальной скорости и угла бросания.

Заключение.

- Переименуйте форму по названию проекта.
- Сделайте дизайн формы удобный для пользователя.
- Сформируйте отчет о создании приложения «Визуализация физического явления» по образцу. (Приложение 1 – Образец отчета лабораторной работы)
- Выложите отчет в систему «Элинос».

