Секция «Моделирование и защита данных в информационных системах»

Е.В. Воробьёва
Научный руководитель: канд. техн. наук, доцент А.А. Захаров
Муромский институт (филиал) Владимирского государственного университета
Владимирская обл., г. Муром, ул. Орловская, д.23
E-mail: evgenya2411@mail.ru

Исследование и реализация алгоритмов электронно-цифровой подписи

В современном мире люди все чаще и чаще сталкиваются с проблемой передачи данных по незащищенным каналам. В последние годы эта проблема удостоилась пристального внимания, так как появилось большое число хакеров, которые понимают самые глубины работы компьютерных систем и намерено изменяют данные для собственной выгоды. Исходя из этого, можно сделать вывод, что данные нуждаются в безопасности, а так же конфиденциальности. Необходимый уровень безопасности информации обеспечивается за счет шифрования пересылаемой информации и применения механизма электронно-цифровой подписи.

Электронная цифровая подпись защищает электронный документ от искажений и является его реквизитом. Этот реквизит получают в результате криптографического преобразования информации. При этом используется закрытый ключ. Тем самым устанавливается защита информации в любых электронных документах.

Технология электронно-цифровой подписи позволяет реализовать защиту от таких действий злоумышленника, как контроль целостности передаваемого документа, защита от подделки документа, невозможность отказа от авторства и подтверждение авторства документа.

Совместно с электронно-цифровой подписью применяют алгориты вычисления хэш-функций. Использование хэш-функций гарантирует, что сообщение не имеет искажений, и получатель получил именно то сообщение.

Существует несколько схем построения цифровой подписи:

- на основе алгоритмов симметричного шифрования. Данная схема предусматривает наличие в системе третьего лица, пользующегося доверием обеих сторон. Авторизацией документа является процесс его шифрования секретным ключом и передача зашифрованного сообщения третьей стороне.
- на основе алгоритмов асимметричного шифрования. На данный момент такие схемы электронно-цифровые подписи наиболее распространены и находят широкое применение.
- разновидности цифровых подписей (групповая подпись, неоспоримая подпись, доверенная подпись), которые являются модификациями описанных выше схем. Их появление обусловлено разнообразием задач, решаемых с помощью электронно-цифровой подписи.

В ходе работы был разработан программный продукт, являющийся наглядным примером генерации и проверки цифровой подписи, где были рассмотрены наиболее удобные средства защиты электронных документов от искажений, при помощи алгоритма RSA.

Н.С. Гоголева
Научный руководитель: канд. техн. наук, доцент С.В.Еремеев
Муромский институт (филиал) Владимирского государственного университета
Владимирская обл., г. Муром, ул. Орловская, д.23
E-mail: natallixxx@mail.ru

Моделирование линейных и полигональных объектов и определение их топологических признаков

С современном мире всё более актуальными становятся геоинформационные системы. Основное применение линейных и полигональных объектов связано с землеустройством, градостроительством и учётом природных ресурсов на различных территориях. Для взаимодействия линейных и полигональных объектов необходимо определить их топологических признаки.

Поскольку требовалось моделировать два типа объектов: линейные и полигональные, то разработка происходила в два этапа. Вначале моделируются линейные объекты, которые состоят из n количества отрезков, расположенных в двумерном пространстве.

Затем происходит моделирование полигональных объектов, поверхность которых строится из набора полигонов.

Моделирование полигональных объектов отличается от моделирования линейных тем, что начальная и конечная вершины должны совпадать. Также минимальное число отрезков должно быть равно 3.

Определение топологических признаков заключается в том, пересекаются ли линейные и полигональные объекты. При пересечении объектов элементы одной фигуры не должны пересекаться между собой.

В результате разработана программа в среде Microsoft Visual Studio 2010, язык C#. Программа моделирует линейные и полигональные объекты и определяет их топологические признаки.

А.Г. Капков
Научный руководитель: канд. техн. наук, доцент С.В. Еремеев
Муромский институт (филиал) Владимирского государственного университета
Владимирская обл., г. Муром, ул. Орловская, д.23
E-mail: afftarkapkov@mail.ru

Моделирование трехмерных фрактальных деревьев

С течением времени трехмерное моделирование занимает все более значимое место в геоинформационных системах. При проектировании трехмерных моделей местности построение объектов живой природы вызывает значительные трудности реализации, так как они имеют сложную, несимметричную структуру, которую трудно повторить.

Построение деревьев осуществлено в среде 3D Studio Max. Для отображения моделей в приложении использована одна из самых популярных технологий DirectX10. Для экспорта моделей из 3D Studio Max в формат, понятный для DirectX использован плагин PandaDirectXMaxExporter.

Основное приложение реализовано в среде программирования Microsoft Visual Studio 2010 на языке C# с использованием технологии Windows Forms. Для взаимодействия среды разработки с трехмерными моделями использован драйвер Microsoft DirectX SDK и его библиотеки. Для полноценной работы приложения на компьютере должен быть установлен пакет средств DirectX, а также платформа .NET Framework 4.0.

В результате реализовано приложение, отображающее несколько видов фрактальных деревьев, реализованных в полнофункциональной программной системе для создания и редактирования трёхмерной графики. Пример отображения построенного дерева представлен на рис. 1.

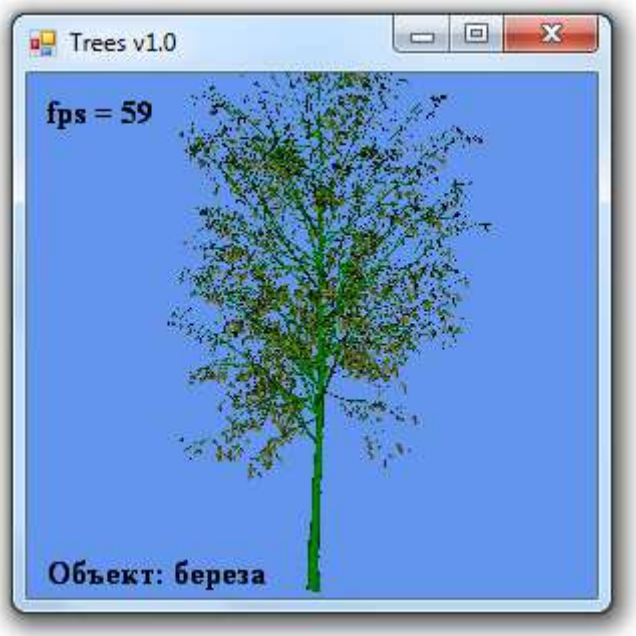

Рис. 1. Форма программы, отображающая выбранное дерево с фоном

Данное приложение ориентировано на просмотр, редактирование, сохранение различных сцен с деревьями живой природы.

Е.А. Капкова
Научный руководитель: канд. техн. наук, доцент С.В. Еремеев
Муромский институт (филиал) Владимирского государственного университета
Владимирская обл., г. Муром, ул. Орловская, д.23
E-mail: katerinaxxllxxll@mail.ru

Имитационное моделирование линейных объектов и определение их топологических признаков

В настоящее время в геоинформационных системах для взаимодействия объектов наиболее актуальными являются топологические отношения. В работе предлагается проанализировать, разработать и исследовать топологические отношения между линейными объектами, к которым можно отнести такие пространственные объекты как: реки, дороги, инженерные коммуникации. Проанализированы следующие виды отношений между линейными объектами: перпендикулярность, пересечение, параллельность и схожесть.

Пример отношения прямых представлена на рис.1.

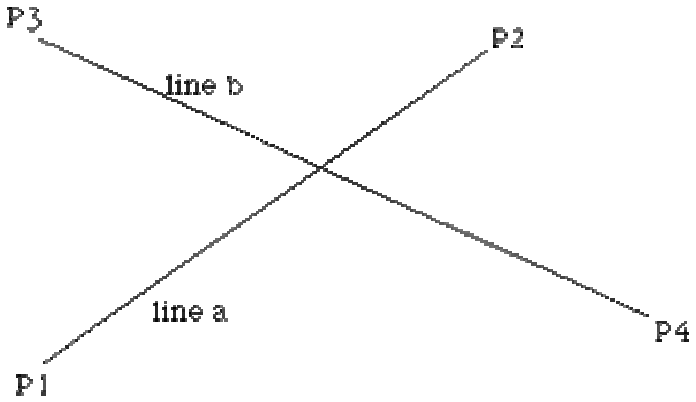

Рис. 1. Отношение прямых

Для определения пересечения прямых необходимо воспользоваться формулой:

$$u_a = \frac{(x4 - x3)(y1 - y3) - (y4 - y3)(x1 - x3)}{(y4 - y3)(x2 - x1) - (x4 - x3)(y2 - y1)}$$

$$u_b = \frac{(x2 - x1)(y1 - y3) - (y2 - y1)(x1 - x3)}{(y4 - y3)(x2 - x1) - (x4 - x3)(y2 - y1)}$$

Если значения u_a и u_b находятся в интервале от 0 до 1, то данные прямые пересекаются.

Формула для определения перпендикулярности прямых:

$$(x2 - x1)(x4 - x3) + (y2 - y1)(y4 - y3) = 0$$

Формула для определения параллельности прямых:

$$(x2 - x1)(y4 - y3) - (y2 - y1)(x4 - x3) = 0$$

Программа разработана в среде программирования Visual Studio 2010 на языке C# с использованием технологий Windows Forms. Основная форма приложения показана на рис.2.

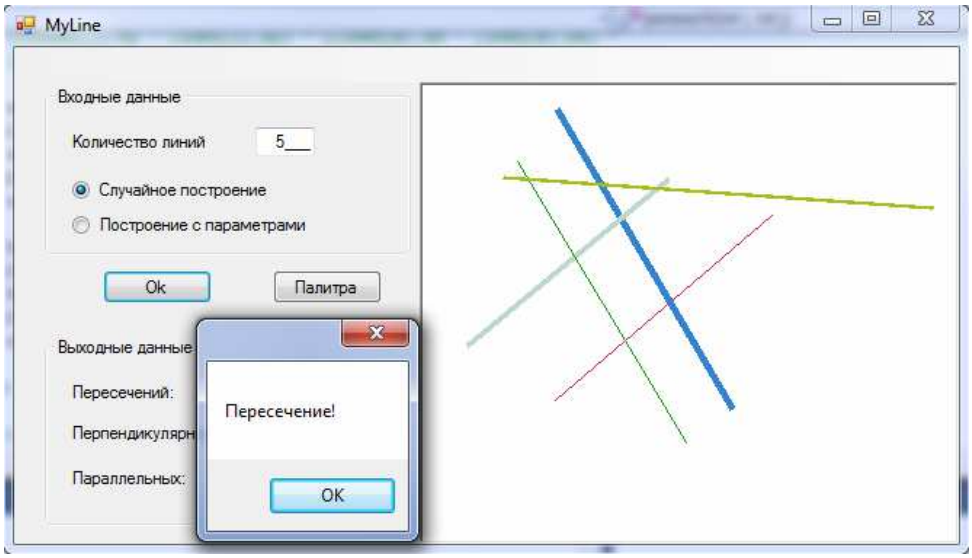

Рис. 2. Обнаружение пересечений

В результате разработано приложение, моделирующее построение линейных объектов и анализирующее их топологические признаки.

К.Д. Кокурин
Научный руководитель: канд. техн. наук, доцент С.В. Еремеев
Муромский институт (филиал) Владимирского государственного университета
Владимирская обл., г. Муром, ул. Орловская, д.23
E-mail: KokurinWork@yandex.ru

Разработка программы пространственно-временного хранения данных

В настоящее время геоинформационные системы (ГИС) дают информацию о карте на текущий момент времени. При обновлении карты предыдущая информация не сохраняется. В рамках научной работы разработана программа, которая позволяет хранить карты в единой системе координат разных времен. Практическое применение разработки – это туризм и историческое картографическое наследие.

Для хранения изменений используются пространственные регистры в виде журнала изменений. Информация о пространственных данных объекта записывается в пространственный регистр, который имеет вид (рис. 1):

Дата	Геометрия объекта
01.10.1970	Геометрия 1
10.02.1998	Геометрия 2
15.02.2013	Геометрия 3

Рис. 1. Структура пространственного регистра

Необходимо для каждого пространственного объекта хранить дату его создания, а также даты всех изменений, которые происходили с этим объектом, и наименование изменения.

Каждая запись в базе пространственно-временных данных привязана к конкретному объекту на электронной карте. База пространственно-временных данных хранит следующую информацию по каждому объекту:

-дату постройки объекта;
-дату демонтажа (ликвидации) объекта;
-даты изменений объекта;
-виды изменений объекта (перестроение, изменение и т.д.).

Система обеспечивает построение и выполнение пространственно-временных запросов следующих типов:

-выборка объектов, входящих в выбранные слои;
-выборка объектов, построенных в указанном диапазоне дат;
-выборка объектов, ликвидированных в указанном диапазоне дат;
-выборка списка объектов с событиями, происходившими в указанном диапазоне дат.

Данный подход позволяет работать с векторными пространственно-временными данными. Карта содержит в себе всю необходимую информацию в разные моменты времени.

Е.А Колесникова
Научный руководитель: канд. техн. наук, доцент А.А. Захаров
Муромский институт (филиал) Владимирского государственного университета
Владимирская обл., г. Муром, ул. Орловская, д.23

Разработка системы стеганографии на основе изображений

Проблема надежной защиты информации от несанкционированного доступа во все времена являлась актуальной. Существует немало способов решения этой проблемы. Одним из них является применение методов стеганографии (с греческого steganos (секрет, тайна) и graphy (запись)). Стеганография играет важную роль в обеспечении безопасной передачи информации. Сокрытие сообщения с помощью методов стеганографии существенно снижает вероятность обнаружения самого факта передачи сокрытого сообщения. А если это сообщение еще и зашифровано, то оно имеет еще один дополнительный уровень защиты.

Компьютерная стеганография подразумевает сокрытие сообщения или файла в другом сообщении или файле. В качестве контейнеров для сокрытия и передачи сообщений могут использоваться различные компьютерные форматы файлов: изображение, текст, звук, видео. Сейчас уже существует довольно большое количество стеганографических программ, использующих компьютерные изображения в качестве контейнеров. Примерами таких программ являются Steganography Tools, Secure Engine, Steganos Security Suite.

В настоящее время наиболее популярными методами стеганографии на основе изображения являются: метод последнего бита, метод дискретно-косинусного преобразования, метод Langelaar. Для реализации был выбран метод последнего бита.

Целью исследований является разработка системы стеганографии на основе изображения. К задачам работы относятся: проведение системного анализа предметной области и существующих программ-аналогов, проведение анализа методов стеганографии и выбор оптимального метода, программная реализация метода последнего бита. В качестве графического контейнера выбран контейнер bmp-формата. Необходимым условием реализации алгоритма является то, что сообщение не должно превосходить 1/8 размера bmp-файла.

В качестве визуальной среды программирования используется Borland C++ 6.0. В программе задействованы следующие функции: функция преобразования двоичного кода в байт, функция преобразования байта в двоичный код, функция преобразования текста в двоичный код, функция получения байта и перевода в двоичный код, подмена последнего бита, функция чтения и проверки ключа, функция вскрытия сообщения из картинки.

Е.А. Кондратьева
Научный руководитель: канд. техн. наук, доцент Е.Е. Канунова
Муромский институт (филиал) Владимирского государственного университета
Владимирская обл., г. Муром, ул. Орловская, д.23
E-mail: kondrateva_kate1994@mail.ru

Программа шифрования и дешифрования текстовой информации.

В рамках научной работы рассмотрены алгоритмы шифрования и дешифрования текстовой информации, в частности алгоритм перестановки и алгоритм замены. Решаемые с их помощью задачи могут быть использованы для обеспечения безопасной передачи текстовой информации в любых сферах деятельности.

Актуальность работы определяется необходимостью защищать информацию от несанкционированного постороннего доступа, а также сохранение конфиденциальности информации.

Потребность в использовании шифров возникла задолго до появления компьютеров. Работа по шифрованию и дешифрованию текстовой информации была трудоёмким, отнимающим время и требующим внимания процессом. С компьютеризацией общества объёмы передаваемой информации увеличились, в связи с этим повысились требования к качеству шифров, возникла необходимость в автоматизации данного процесса.

Целью работы является создание программы помогающей автоматизировать данный процесс, сравнение алгоритмов перестановки и замены при помощи среды программирования Microsoft Visual Studio и языка C++.

Программа обрабатывает текст, введенный с клавиатуры или содержащийся в уже имеющемся файле. Для этого были реализованы задачи по созданию программ кодирования и декодирования текстовой информации, как для алгоритма замены, так и для алгоритма перестановки. Результаты кодирования сохраняются в файле.

К.В. Купцов
Научный руководитель: канд. техн. наук, доцент С.В. Еремеев
Муромский институт (филиал) Владимирского государственного университета
Владимирская обл., г. Муром, ул. Орловская, д.23
E-mail: kirill-kuptsov@rambler.ru

Проверка топологической корректности расположения трехмерных объектов

Довольно часто от сотрудников строительных организаций, архитекторов требуется проверить расположение объектов относительно друг друга. Можно ли изменить их взаимное расположение? Существует ли возможность повернуть данный объект, не изменяя нахождение близлежащего объекта? Насколько корректно текущее расположение?

Для этих целей предназначена программа проверки топологической корректности расположения трехмерных объектов. Данное приложение позволяет создавать трехмерные объекты, после чего возможно произвести проверку на их взаимное расположение. Если такое размещение возможно, то программа выдаст соответствующее уведомление. Если же данное размещение объектов невозможно, программа также сообщит об этом.

Для построения объектов используются технологии WPF. Далее рассматривается построение трехмерного объекта на примере куба.

Первая задача в построении куба: определить способ его разбиения на треугольники. Каждый треугольник подобен простой двухмерной фигуре.

Куб состоит из шести квадратных сторон. Для отображения каждого квадрата необходимо два треугольника.

Для сокращения накладных расходов и повышения производительности в приложении, которое формирует трехмерные объекты, принято избегать визуализации тех объектов, которые невидимы. Например, если известно, что задняя грань куба, показанного на рисунке, никогда не будет видна, то нет необходимости определять треугольники для этой стороны.

Следующим шагом является определение углов куба. Определение начинается с четырех точек задней стороны, затем добавляются четыре точки лицевой стороны и эти точки отображаются на треугольники.

При определении треугольники должны находиться в направлении против часовой стрелки, чтобы их лицевая сторона смотрела вперед. Но в кубе нарушено это правило. Квадраты лицевой стороны определяются в порядке против часовой стрелки, но поверхность задней стороны описана по часовой стрелке. Это объясняется тем, что обратная сторона куба должна обращать свою лицевую сторону назад. Для того чтобы лучше представить это, допустим, что куб будет вращаться вокруг оси Y, так что обратная сторона переместится вперед. Теперь те треугольники, которые смотрели назад, будут повернуты вперед, что сделает их полностью видимыми, и получается именно то поведение, которое нужно.

Подводя итог работы, перечисляются основные возможности программы, проверяющей топологическую корректность расположения трехмерных объектов.

Приложение позволяет:

- создавать трехмерные объекты;
- изменять взаимное расположение объектов.

А также проверять топологическую корректность расположения трехмерных объектов, в результате чего, пользователю выдается уведомление: корректно текущее расположение или же нет.

В.Д. Малыгина
Научный руководитель: старший преподаватель М.И. Ткачук
Муромский институт (филиал) Владимирского государственного университета
Владимирская обл., г. Муром, ул. Орловская, д.23
E-mail: Lazygirl93@rambler.ru

Разработка ИС безопасной передачи файлов по сети

Введение

Передаваемые данные по сети всегда представляют цель атаки злоумышленника (особенно если данные представляют ценность).

Отсюда возникает вопрос их защиты. Для этого используется шифрование. Так как различных требований к защите данных могут быть разные, поэтому необходимо иметь возможности шифрования данных разными способами без переписывания программы. Этот вопрос решается разработкой плагинов.

1. Симметричное шифрование

Симметричное шифрование – способ шифрования, в котором для шифрования и расшифровывания применяется один и тот же криптографический ключ. До изобретения схемы асимметричного шифрования единственным существовавшим способом являлось симметричное шифрование. Ключ алгоритма должен сохраняться в секрете обеими сторонами. Алгоритм шифрования выбирается сторонами до начала обмена сообщениями. [1]

Алгоритмы шифрования и дешифрования данных широко применяются в компьютерной технике в системах сокрытия конфиденциальной и коммерческой информации от злонамеренного использования сторонними лицами. Главным принципом в них является условие, что передатчик и приемник заранее знают алгоритм шифрования, а также ключ к сообщению, без которых информация представляет собой всего лишь набор символов, не имеющих смысла. [2]

Основные алгоритмы шифрования

– DES (*Data Encryption Standard*) – симметричный алгоритм шифрования, разработанный фирмой IBM и утвержденный правительством США в 1977 году как официальный стандарт (FIPS 46-3). DES имеет блоки по 64 бита и 16 цикловую структуру сети Фейстеля, для шифрования использует ключ с длиной 56 бит. Алгоритм использует комбинацию нелинейных (S-блоки) и линейных (перестановки E, IP, IP-1) преобразований.

– Rijndael – симметричный алгоритм блочного шифрования (размер блока 128 бит, ключ 128/192/256 бит), принятый в качестве стандарта шифрования правительством США по результатам конкурса AES. Этот алгоритм хорошо проанализирован и сейчас широко используется. [6]

2. Архитектура системы

Система состоит из 3 подсистем (клиентская часть, серверная часть, плагины шифрования). Клиентская часть состоит из 6 модулей, серверная – 5 модулей. Между модулями и подсистемами осуществляется взаимодействие через информационные потоки. На рис. 1 представлена структурная схема системы и взаимодействия между ее модулями.

3. Механизм работы системы

Система предназначена для безопасной передачи файлов по сети. Приложение состоит из 3 исполняемых модулей sockets_ sharp _client.exe, sockets_sharp_server.exe и плагины (независимо компилируемый программный модуль, динамически подключаемый к основной программе и предназначенный для расширения и/или использования её возможностей), содержащие методы для шифрования файлов.

Алгоритм передачи файлов с шифрованием выглядит следующим образом:

- пользователь и сервер загружают ключ шифрования;
- выбирается файл для передачи;
- файл шифруется и передается по сети;
- принимающая сторона расшифровывает и сохраняет исходный файл.

Алгоритм загрузки плагинов:

- читается опция из файла конфигурации с каталогом плагинов;
- плагины загружаются в цикле и вызывается метод GetMethod;
- название метода сравнивается с требуемым;
- если требуемый метод найден, то остальные плагины не загружаются.

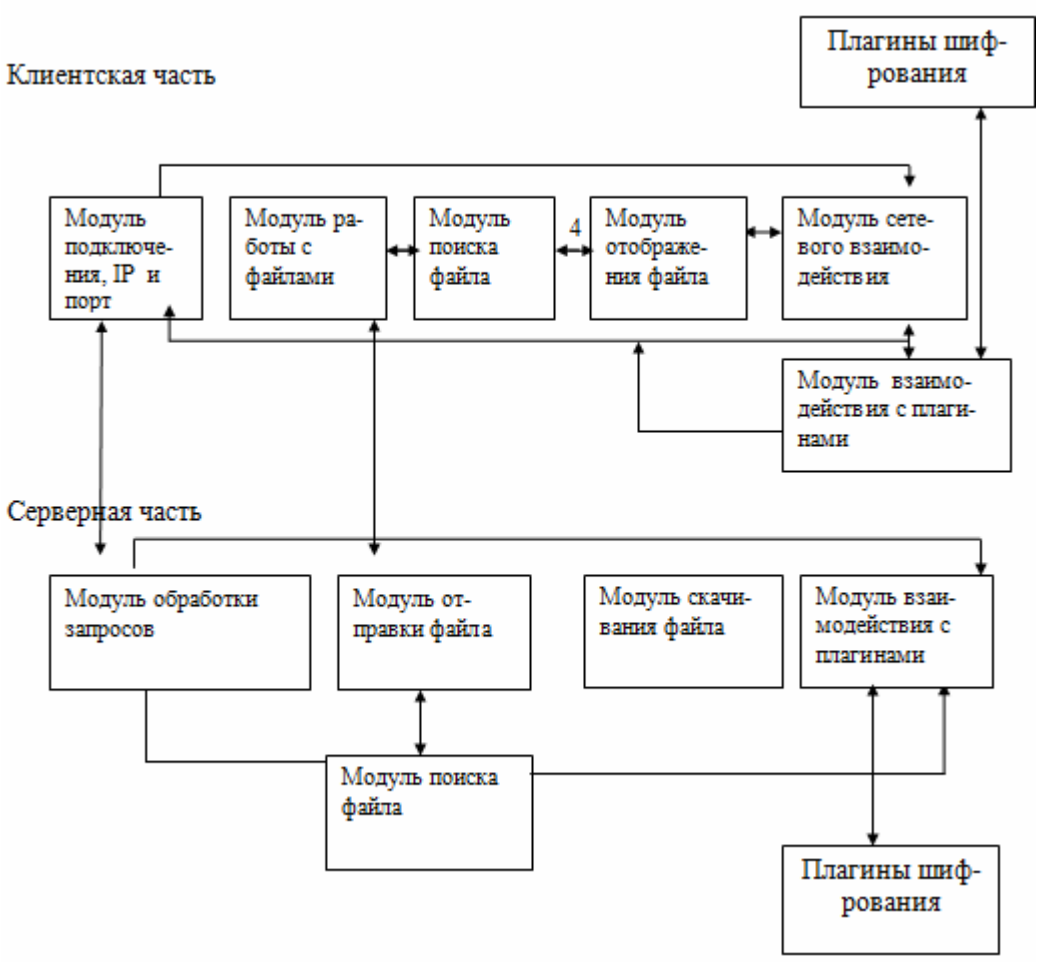

Рис. 1. Архитектура системы

Вывод

В итоге было разработано приложение, которое отвечает всем требованиям, предъявленным в ходе разработки. В итоге определены следующие функции ИС: загрузка файлов с сервера и на сервер, отправление файлов, обновление списка файлов, шифрование и дешифрование файлов. Основное приложение независимо оперирует плагинами шифрования, предоставляя пользователям возможность динамически добавлять и обновлять плагины без необходимости внесения изменений в основное приложение.

Литература

1. Уотсон, Карл и др. Visual C# 2008: базовый курс.: Пер. с англ.- М.: ООО «И. Д. Вильямс», 2009.- 1216 с.
2. Ватсон. C# на примерах.- СПб.: БХВ-Петербург, 2011. - 608 с.
3. Петцольд. Програмирование с использованием Microsoft Windows Forms. Пер. с англ.-М.: Русская Редакция; СПб.: Питер, 2006. - 432с.
4. http://www.citforum.ru
5. http://www.sql.ru
6. msdn.microsoft.com

Т.А. Москвина
Научный руководитель: старший преподаватель М.И. Ткачук
Муромский институт (филиал) Владимирского государственного университета
Владимирская обл., г. Муром, ул. Орловская, д.23

Разработка подсистемы управления модулями шифрования данных

Введение

Шифрование – это способ сокрытия исходного смысла сообщения или другого документа, обеспечивающий искажение его первоначального содержимого. Шифрование применяется для хранения важной информации в ненадёжных источниках и передачи её по незащищенным каналам связи. Такая передача данных представляет из себя два взаимно обратных процесса: Перед отправлением данных по линии связи или перед помещением на хранение они подвергаются шифрованию. Для восстановления исходных данных из зашифрованных применяется процедура дешифрования.

1. Виды шифрования

Существует множество алгоритмов шифрования данных и их стоит применять в зависимости от государственных стандартов или ~~каких-либо~~ и определенных требований безопасности. Наиболее распространены следующие методы:

Data Encryption Standard (DES), предназначенный для использования в государственных и правительственных учреждениях США для защиты от несанкционированного доступа важной, но несекретной информации. Основные достоинства алгоритма DES: относительная простота алгоритма обеспечивает высокую скорость обработки информации; достаточно высокая стойкость алгоритма.

Advanced Encryption Standard (AES), также известный как Rijndael — симметричный алгоритм блочного шифрования, принятый в качестве стандарта шифрования правительством США по результатам конкурса AES. Этот алгоритм хорошо проанализирован и сейчас широко используется, как это было с его предшественником DES. Преимуществами данного алгоритма являются: не подвержен многим видам криптоаналитических атак; высокое быстродействие Rijndael на различных платформах.

ГОСТ 28147-89 – отечественный стандарт шифрования. Достоинства ГОСТ 28147-89: бесперспективность силовой атаки (XSL-атаки в учёт не берутся, так как их эффективность на данный момент полностью не доказана); эффективность реализации и соответственно высокое быстродействие на современных компьютерах; наличие защиты от навязывания ложных данных (выработка имитовставки) и одинаковый цикл шифрования во всех четырех алгоритмах ГОСТа.

2. Разработка интерфейса модуля шифрования

Так как существует множество алгоритмов симметричного шифрования, то часто требуется использовать разные алгоритмы для решения различных задач, руководствуясь государственными стандартами и прочими факторами. Поэтому каждый алгоритм шифрования должен быть реализован в виде отдельного модуля расширения, что позволит изменять способ шифрования, не изменяя основную программу.

Определим базовый класс BaseEncrypt – общий интерфейс для всех алгоритмов шифрования, который для каждого алгоритма должен быть реализован. В нем прописаны следующие методы:

```
string GetMethod();
void SetKey(string key);
byte[] Encrypt(byte[] data)
byte[] Decrypt(byte[] cryptData)
```

Первый метод возвращает название алгоритма шифрования. Метод SetKey задает ключ шифрования. Encrypt нужен для шифрования информации. Decrypt необходим для расшифрования зашифрованного сообщения. Все методы реализуются в методах класса - наследника.

3. Реализация модулей шифрования в MS .Net Framework

Рассмотрим реализацию модуля шифрования – DES. Мы создаем класс DESEncrypt, наследуемый от BaseEncrypt. Далее создаем объект шифрования DES.Create. В данном случае DES – класс, методами которого реализуется алгоритм DES в .NET Framework. Для него создаются все реализации System.Security.Cryptography.DES. В методе SetKey формируем ключ из пароля с помощью расширения алгоритма PBKDF1. Далее задаем имя хэш-алгоритма для данной операции (будем использовать SHA512). Затем получаем ключ из значения хеш функции и задаем режим блочного шифра для дальнейшего использования, например режим CBC. Задаем вектор инициализации для алгоритма симметричного шифрования. В методе Encrypt создаем поток в памяти, содержащего зашифрованные данные. С помощью CryptoStream шифруем информацию, т.е. сначала записываем данные в поток для шифрования, далее шифруем последний блок и в итоге сохраняем зашифрованные данные в массив байт. В методе Decrypt реализована расшифровка. По аналогии с шифрованием создаем поток с зашифрованными данными, далее создаем поток дешифрации, связанный с outstream-объектом класса CryptoStream. Далее расшифрованные данные заносим в массив байт и возвращаем это значение.

4. Механизм взаимодействия с модулями шифрования

Механизм взаимодействия с модулями шифрования приведен на следующем рисунке:

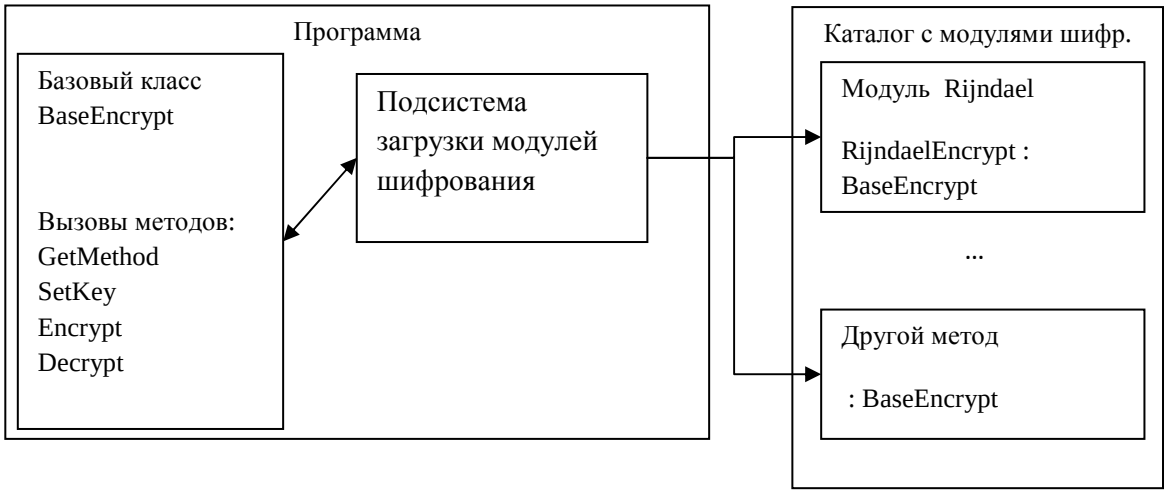

Рис. 1. Механизм взаимодействия с модулями

Заключение

В статье рассмотрены основные методы шифрования, все они отличаются и могут быть использовании в соответствии с различными государственными стандартами. Для того, чтобы была возможность использовать различные методы без переписывания системы, был разработан и реализован механизм плагинов: создали базовый класс, модули которого его реализуют. Один модуль – один алгоритм шифрования, содержащий класс, унаследованный от базового.

Н.С. Мохов
Научный руководитель: канд. техн. наук, доцент С.В. Еремеев
Муромский институт (филиал) Владимирского государственного университета
Владимирская обл., г. Муром, ул. Орловская, д.23
E-mail: moxov@outlook.com

Моделирование загруженности автомобильных дорог на основе предыдущих наблюдений

Моделирование различных процессов, связанных с жизнедеятельностью человека, помогает разрабатывать геоинформационные модели, которые служат для сбора, хранения, анализа и представления в графическом виде данных.

В современном мире человеку часто приходиться передвигаться из одного места в другое, например, из дома до работы и обратно, доехать до родственников и т.д. В современных реалиях большой загруженности дорог в крупных городах возникает необходимость предсказывать появление пробок и заторов на дорогах для планирования своей деятельности.

Моделирование загруженности дорог рассматривалось на основе полученных данных за определенный промежуток времени и прогнозировании загруженности на последующий период. При моделировании использовался дорожный граф. Он представляет из себя одну из основных частей геоинформационного приложения. Он располагает внутри себя информацию о дорогах (перекрестки, геометрию дорог, длину и т.д.). Граф дорог различается с «естественным» графом тем, что в нем существуют продублированные вершины и дуги. Это необходимо для того чтобы учитывать правила дорожного движения (проезды, запрещенные повороты).

Каждая дуга (улица) имеет характеристику (длина отрезка улицы, скорость движения транспортного потока без загруженности дороги). Данные о пробках берутся за n дней учётного периода, в каждом дне имеется градация по часам (имеются данные о загруженности за каждый час).

В итоге разработана программа для моделирования загруженности автомобильных дорог на основе предыдущих наблюдений, позволяющая просматривать граф дорог и получать прогноз о загруженности дорог. Программа написана на языке C# в визуальной среде разработки Visual Studio 2012 с использованием технологии Windows Forms. Программа протестирована и готова к использованию.

О.Ю. Палутина
Научный руководитель: канд. техн. наук, доцент С.В. Еремеев
Муромский институт (филиал) Владимирского государственного университета
Владимирская обл., г. Муром, ул. Орловская, д.23
E-mail: helga0989@yandex.ru

Разработка программы поиска схожих городских территорий на основе изоморфности графов

В настоящее время рост урбанизации стремительно увеличивается. Зачастую работникам в среде проектирования приходится изо дня в день решать задачи, связанные с планированием построек домов, дорог, мостов и т.д. В связи с этим было принято решение о разработке программы поиска схожих городских территорий на основе изоморфности графов.

Программа сравнивает ранее сформированные графы города. Признаком схожести графов является изоморфизм графов.

В работе использован алгоритм определения изоморфности двух графов A и B на основе их матрицы смежности. Для этого проверяется ряд возможных условий, включая проверку количества вершин, количества ребер, вычисление степеней вершин, суммарное кратчайшее расстояние между вершинами с использованием индекса Винера.

Для определения неизморфности графов используется оценка степени различия, которая рассчитывается на основе сравнения числа вершин с различными степенями, индекса Винера и индекса Рандича.

Для решения поставленной задачи по определению схожести графов городских территорий разработана программа, включающая два модуля:

- Модуль-редактор графов: необходим для создания графов городских территорий и сохранения их в файлы для последующего анализа.
- Модуль для анализа изоморфности графов. Выполняет открытие двух ранее сформированных графов из файлов и их сравнение.

Д.С. Потапов
Научный руководитель: канд. техн. наук, доцент С.В. Еремеев
Муромский институт (филиал) Владимирского государственного университета
Владимирская обл., г. Муром, ул. Орловская, д.23
E-mail: karrsarr@mail.ru

Имитационное моделирование движения автобусов для составления оптимального расписания

В современном обществе управление автобусным парком становится наиболее сложным, в связи с развитием городов, строительством новых дорог и увеличением автотранспорта. Для повышения эффективности управления транспортных систем необходимо автоматизировать и оптимизировать процессы управления, прибегая к помощи имитационного моделирования и вычислительной техники.

Одним из основных факторов составления оптимального расписания движения автобуса является объем пассажиропотока. На основе данных, полученных экспериментальным путем, производится анализ количества прибывших на остановку пассажиров, а также время их прибытия, в результате чего составляется оптимальное расписание движения автобуса по маршруту. Также в программе учитываются дополнительные данные:

1. количество мест в автобусе;
2. время движения автобуса по маршруту;
3. количество автобусов на маршруте и выделение дополнительных.

Приложение разработано с помощью среды Microsoft Visual Studio 2010 с использованием технологии Windows Forms. В приложении графически реализовано движение автобуса по маршруту, появление пассажиров на остановках, а также вывод оптимального расписания для движения автобусов и их количества.

А.А. Трифонов
Научный руководитель: канд. техн. наук, доцент С.В. Еремеев
Муромский институт (филиал) Владимирского государственного университета
Владимирская обл., г. Муром, ул. Орловская, д.23
E-mail: antoon1993@yandex.ru

Имитационное моделирование реагирования служб на чрезвычайные ситуации

Современная жизнь общества характеризуется постоянными угрозами и происшествиями. С каждым годом увеличивается попадание людей в различного рода чрезвычайные ситуации. В настоящее время существует множество служб, которые помогают обществу справиться с этими ситуациями. Успех помощи пострадавшим зависит в большинстве случаев от скорости приезда службы, поэтому им необходимо знать информацию об оптимальных путях к любому месту происшествия или аварии.

Разработана программа, которая рассчитывает оптимальный маршрут до места происшествия и по этому маршруту имитирует движение службы. Оптимальный маршрут рассчитывается с помощью алгоритма Дейкстры. Входными параметрами программы является место нахождения службы, место происшествия, а также матрицы расстояний и загруженности дорог, по которым рассчитывается оптимальный путь.

Работа программы состоит из следующих этапов:

1. пользователем выбирается место чрезвычайной ситуации;
2. вычисление оптимального маршрута;
3. имитируется движение службы до чрезвычайной ситуации.

Приложение реализовано в среде Microsoft Visual Studio 2010 с использованием технологии Windows Forms. Приложение имеет простой пользовательский интерфейс и позволяет графически отслеживать движение служб.

Хорева Д.А.
Научный руководитель: канд. техн. наук, доцент А.А. Захаров
Муромский институт (филиал) Владимирского государственного университета
Владимирская обл., г. Муром, ул. Орловская, д.23

Разработка системы аутентификации на основе паролей

Аутентификация – это процедура проверки подлинности. Проверка подлинности может быть односторонней или взаимной. Она проводится с помощью криптографических методов.

Существуют следующие виды аутентификации:

- аутентификация по многоразовым паролям;
- аутентификация по одноразовым паролям;
- многофакторная аутентификация.

Аутентификация по многоразовым паролям состоит в использовании пользовательского идентификатора, называемого «логином» – некой конфиденциальной информации. Достоверная пара логин-пароль хранится в специальной базе данных.

В работе была создана система аутентификации на основе многоразовых паролей.

Т.к. хранить пароль в явном виде небезопасно, то используется процедура хеширование паролей. Хеширование – преобразование по заданному алгоритму входных данных произвольной длины в выходную битовую строку фиксированной длины. Один из самых распространенных хеш-алгоритмов – это алгоритм MD5. Этот алгоритм был разработан в 1991 году Рональдом Л. Ривестом. Формирование хеша происходит в несколько этапов: выравнивание потока, добавление длинны сообщения, инициализация буфера, вычисления значений в цикле.

В рамках исследования был разработан модуль, наглядно демонстрирующий защиту от несанкционированного доступа. Система включает в себя приложение и базу данных. В базе данных содержится вся нужная информация о пользователях, а также пароли для входа в систему каждого сотрудника. Пароли хранятся в хешированном виде. Хеширование реализовано с помощью алгоритм MD5.

Начало работы с приложением начинается с регистрации пользователя. Имеются следующие необходимые для заполнения поля: фамилия, имя, логин и пароль. Пароль должен состоять из определенного числа букв и цифр. Если данные условия не будут соблюдены, то система выдаст ошибку и регистрация будет прервана.

Если все поля были успешно заполнены, то пользователь сможет войти в систему, введя ранее заполненные данные (логин и пароль). В системе введено ограничение на количество попыток ввода неправильного пароля. Это было введено в качестве защиты от подбора пароля.

С.В. Шатков
Научный руководитель: канд. техн. наук, доцент С.В. Еремеев
Муромский институт (филиал) Владимирского государственного университета
Владимирская обл., г. Муром, ул. Орловская, д.23
E-mail: shatkov.s@yandex.ru

Моделирование полигональных объектов и определение топологических признаков

В настоящее время большую роль в геоинформационных системах занимает определение размещения фигур на плоскости в зависимости друг от друга. Это обусловлено распространением GPS-навигации для определения местоположения. Для того чтобы спроектировать некоторый план местности требуется проанализировать и исследовать топологические отношения между геометрическими фигурами, к которым можно отнести такие пространственные объекты как здания, спортивные объекты и с/х поля. При анализе рассмотрены следующие признаки:

1. Пересечение нескольких геометрических фигур;
2. Положение фигур относительно друг друга;
3. Вхождение геометрического тела в другое геометрическое тело.

В ходе проектирования использованы следующие математические формулы:

1. Пересечения прямых;
2. Вхождение точки в фигуру;
3. Положение точки (прямой) относительно прямой.

В итоге разработана программа, моделирующая полигональные объекты на плоскости с возможностью изменения положения отдельного геометрического тела и количества углов для многоугольных фигур. Основным направлением программы является определение топологических зависимостей объектов относительно друг друга, а именно положение и пересечение геометрических фигур. Приложение написано в среде программирования Visual Studio с использованием технологии Windows Forms.