УДК 004.42

Крыжановская Ю.А., Кашко В.В.

Воронежский государственный университет, г. Воронеж, Россия

РАЗРАБОТКА КОМПЛЕКСА ОБУЧАЮЩИХ ПРОГРАММ ДЛЯ КУРСА «МАТЕМАТИЧЕСКИЕ ОСНОВЫ ЗАЩИТЫ ИНФОРМАЦИИ И ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ»*

АННОТАЦИЯ

В статье обсуждается разработанный комплекс обучающих программ для курса «Математические основы защиты информации и информационной безопасности», изучаемого обучающимися в магистратуре по направлению «Фундаментальная информатика и информационные технологии». Рассматривается структура комплекса программ и его использование в различных режимах работы.

КЛЮЧЕВЫЕ СЛОВА

Комплекс программ; программирование; функциональный состав комплекса программ; режим работы; формирование тестов; просмотр теоретических данных; тестирование; результаты тестирования.

Yuliana Kryzhanovskaya, Vasily Kashko

Voronezh State University, Voronezh, Russia

COURSEWARE ATHORING FOR COURSE "DATA PROTECTION AND INFORMATION SECURITY MATHEMATICAL FOUNDATIONS"

ABSTRACT

The article describes courseware authoring for course “Data protection and information security mathematical foundation” which is studied by students of "Fundamental computer science and information technology” Master's Degree programme. The program complex structure and its application in different operation modes are considered.

KEYWORDS

Program complex; programming; program complex functional structure; operation mode; test generation; theoretical data viewing; testing; test results.

Введение

Дисциплина «Математические основы защиты информации и информационной безопасности» входит в вариативную часть общенаучного цикла учебного плана магистерской программы по направлению подготовки 02.04.02 (010300) Фундаментальная информатика и информационные технологии и является дисциплиной по выбору в 3 семестре. Информация о дисциплине приводится в основной образовательной программе высшего образования [1]. Цель изучения данной дисциплины – формирование у студентов знаний по математическим основам обеспечения информационной безопасности информационно-управляющих и информационно-логистических систем. Задача дисциплины: дать студентам необходимые знания, умения и навыки, в том числе: о математических основах построения криптографических алгоритмов и систем защиты, теоретические и практические знания в области проблем обеспечения информационной безопасности информационно-управляющих и информационно-логистических систем навыки самостоятельного, творческого использования теоретических знаний для предотвращения незаконного использования информации в практической деятельности.

В рамках дисциплины «Математические основы защиты информации и информационной безопасности» предусмотрены занятия лекционного и лабораторного типов. Для повышения качества усвоения излагаемого материала и обеспечения дополнительной возможности по

* **Труды XI Международной научно-практической конференции «Современные информационные технологии и ИТ-образование» (SITITO’2016), Москва, Россия, 25-26 ноября, 2016**

самостоятельной работе студентов разработан комплекс обучающих программ по ряду тем, входящих в программу курса, в частности, таких, как:

- элементы теории чисел и модулярная арифметика;
- трудные проблемы теории чисел, используемые в криптографии;
- тесты разложимости и тесты простоты, факторизация чисел;
- генераторы псевдослучайных последовательностей;
- криптография: основные определения, цели, задачи, типы криптосистем, атаки на криптосистемы;
- алгоритмы сжатия;
- классификация шифров;
- криптостойкость, имитостойкость;
- идентификация и аутентификация;
- цифровая подпись;
- управление криптографическими ключами.

Состав комплекса программ и режимы работы

- Комплекс программ состоит из отдельно разработанных частей, обеспечивающих:
- работу в режиме разделения доступа, реализацию пользовательского интерфейса;
- возможность ознакомления с информацией по выбранной теме;
- проведение тестирования;
- отображение результатов.

Также включен ряд демонстрационных программ, реализующих некоторые алгоритмы, представленные в теоретических материалах. Представленный комплекс обучающих программ разработан среде visual studio 2015 [2].

Данные о пользователях хранятся в базе данных, реализованной с использованием SQLite [3]. Теоретические материалы, тестовые задания и отображаемые варианты ответов на вопросы хранятся в виде отдельных файлов и предоставляются пользователям по мере надобности.

Для разделения функциональных возможностей различных категорий пользователей комплекс программ предусматривает два режима, ориентированных на преподавателя и студента. На настоящий момент функции администратора доступны в профиле преподавателя. Ведется работа по выделению части полномочий, связанных с обработкой учетных записей, в отдельную категорию Администратор.

Режим, допустимый для конкретного пользователя, определяется после прохождения процедур идентификации и аутентификации. В рассматриваемом программном комплексе реализована парольная аутентификация. Информация об учетных записях хранится в закрытом виде в базе данных.

Режим работы преподавателя, как упоминалось, предполагает возможность работы с учетными записями студентов, т.е. регистрации новых студентов, редактирование записей, удаления учетных записей (пример окна регистрации приведен на Рис. 1).

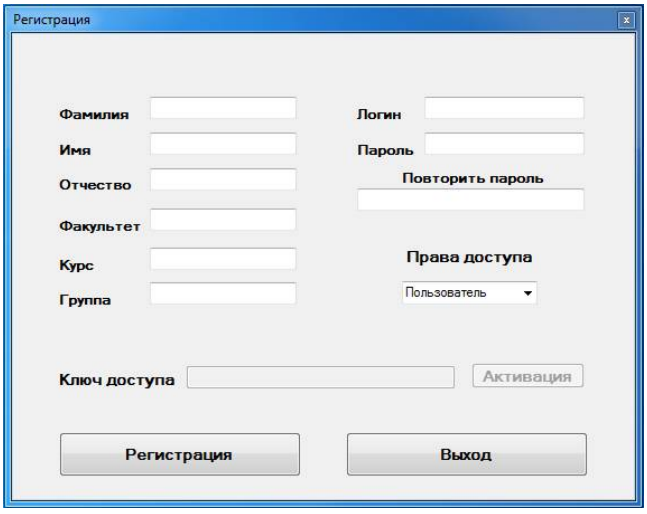

Рис.1. Форма регистрации

Ключ доступа задается в том случае, если проходящим регистрацию пользователь будет преподаватель (задается полем права доступа).

Присутствует также возможность просмотра результатов прохождения тестов по указанной теме зарегистрированными студентами. С целью обеспечения большей гибкости системы предусматривается модификация содержания теоретического материала путем загрузки нового файла. Кроме того, реализовано редактирование составляющих тесты вопросов и соответствующих им вариантов ответов.

Возможности студента по работе с комплексом программ представлены возможностями по ознакомлению с информацией по выбранной теме, прохождением тестирования (с фиксацией результатов в базе данных) и просмотре полученных результатов с указанием оценки за конкретный тест.

Оформление теоретических сведений и подготовка тестирования

Теоретический материал для данного комплекса программ оформляется в виде pdf-файла. Для его редактирования/формирования можно воспользоваться любым текстовым редактором, допускающим экспорт в формат .pdf. Затем полученный файл располагается по следующему пути: <Буква диска>:\Visual Studio\<CourseWare>\<Название тематической программы>\bin\Debug. Файл должен иметь имя Test, в противном случае теория будет недоступна из приложения.

Тестовый материал состоит из трёх основных частей:

1. Файл, содержащий вопросы.
2. Файл, содержащий номера правильных ответов.
3. Папка, в которой хранятся изображения с вариантами ответов.

Для изменения или формирования тестовой информации нужно выполнить следующие действия:

1. Редактирование файла вопросов и ответов осуществляется при выборе в меню программы пункта Администрирование\Редактировать тесты\Редактор вопросов (в случае редактирования файла вопросов).
2. Администрирование\Редактировать тесты\Редактор ответов (в случае редактирования файла ответов).

После чего на экране появится содержимое соответствующего файла. Редактирование необходимо выполнять непосредственно в текстовом поле. При этом каждый вопрос и каждый ответ записывается в отдельной строке и должен заканчиваться специальным знаком ($), который является разделителем. В файле вопросов приводятся все вопросы теста, а в файле ответов прописываются номера правильных ответов, которые таким же образом разделяются разделителем $. При этом если у вопроса несколько правильных ответов, то их номера прописываются через запятую.

3. Редактирование изображений с вариантами ответов.

Варианты ответов приводятся в формате изображений, которые должны размещаться по заданному адресу.

Для создания новых изображений можно воспользоваться текстовым редактором, а затем – графическим редактором (например, paint). Процесс редактирования изображений ответов можно разделить на несколько шагов:

Шаг 1: В текстовом редакторе набрать пронумерованные варианты ответов на конкретный вопрос;

Шаг 2: Далее применить Print Screen с набранным текстом;

Шаг 3: Открыть Paint (или другой графический редактор) и сделать заготовку размером 579*160 пикселей, либо воспользоваться уже готовым паттерном, расположенным по следующему пути:

<Буква диска>:\Visual Studio\<CourseWare>\<Название тематической программы>\packages\Code\Шаблон рисунка.bmp.

Шаг 4: Поместить изображение в паттерн и сохранить в папке по следующему пути: <Буква диска>:\Visual Studio\<CourseWare>\<Название тематической программы>\bin\Debug\Img

Имена файлов изображений ответов задаются целочисленными значениями (номерами вопросов, которым соответствуют варианты на изображении). Нумерация начинается с 0.

Прохождение теста и просмотр результатов

Для старта тестирования студент должен ввести свои логин и пароль. В случае первого использования программы необходимо пройти регистрацию, которая осуществляется преподавателем. Для этого в режиме работы преподавателя необходимо заполнить регистрационную форму.

Если студент ранее был зарегистрирован, то функционал преподавателя/администратора

станет недоступен, а будет предложено выбрать раздел, изучение которого планируется, а затем – одно из действий: просмотреть теоретический материал, пройти тестирование или просмотреть результаты. При выборе просмотра теории будет предъявлена информация по заданной теме. В случае, если выбрано прохождение тестирования, будут последовательно предъявляться вопросы теста (пример вопроса по теме Модулярная арифметика с вариантами ответа приведен на Рис. 2).

Правильный (по мнению обучающегося) ответ засчитывается при нажатии кнопки Принять ответ. После этого становится активной копка Следующий вопрос.

После ответа на предложенные вопросы теста обучающемуся сообщается результат прохождения теста с указанием процента правильных ответов и полученной оценки.

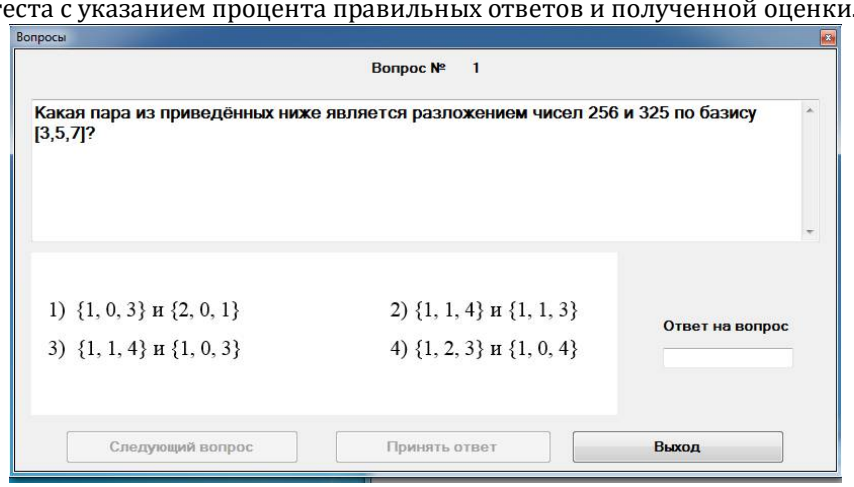

Рис. 2. Пример вопроса теста по теме Модулярная арифметика с вариантами ответа

Для просмотра результатов тестирования используются пункты меню Файл -> Результаты тестирования. Следует отметить, что студенту доступны для просмотра только свои собственные результаты.

Выход из программы возможен с использованием кнопки "Выход" или пунктов меню Файл -> Выход. Для того, чтобы свернуть окно в трей, можно использовать пункт меню Файл -> Свернуть.

Примеры демонстрационных программ

Демонстрационные программы для каждой из рассматриваемых тем включены в разработанный комплекс программ. Рассмотрим некоторые из них.

1. Генерация «большого» нечетного числа и проверка его на простоту по методу Миллера -Рабина. Данная программа состоит из двух основных модулей: класс BigNumber, класс Program (в котором производится инициализация объекта класса BigNumber).

Работа класса BigNumber основана на использовании типа BigInteger который включён в пространство имён usyng System.Numerics. Он включает в себя следующие методы:

- public BigInteger GetNumber(int count) - генератор большого числа заданного размера. Параметр count предназначен для указания количества блоков чисел (один блок состоит из 10 символов). Генерация числа производится следующим образом: Есть целочисленная переменная, в которую в цикле (количество итераций равно count) генерируется случайное число в диапазоне [2000000000, 2147483647], и есть строка, в которую поочерёдно, путём конкатенации, добавляются сгенерированные ранее, но преобразованные в строку числа. Таким образом, в конце выполнения count итераций в строке будет записано число, размером count*10 символов. Далее создаётся переменная типа BigInteger, в которую путём применения встроенной функции для данного типа BigInteger.Parse(<string>) записывается строка и преобразовывается в тип BigInteger. В конце работы метод возвращает полученное число.
- public bool MillerRabin(BigInteger number) - функция проверки на простоту по методу Миллера-Рабина. На вход данного метода подаётся проверяемое число. Далее данное число представляется в виде n-1=(2^k)*m, откуда находятся значения k и m. Далее генерируется число a из диапазона [2,n-2] (так называемый свидетель). На следующем этапе выполняем операцию T = (a^m) mod n. Если в результате T == 1 или T == -1, то число с вероятностью 25% является простым, иначе в цикле по переменной k выполняется операция T = (T^2) mod n и выполняется проверка значения T. Если T = 1, то число не является простым; если T = -1, то число с вероятностью 25% простое. Результатом работы данного метода является булевская переменная, которая имеет значение true - в случае, когда проверяемое число простое и false - когда не является простым.

В классе Program создаётся объект класса BigNumber, далее в цикле вызывается метод

генерации числа и его проверка на простоту. Цикл остановится лишь в том случае, когда метод проверки вернёт значение true, что означает генерацию простого числа.

2. Демонстрация работы алгоритма инкрементного кодирования (фронтальное сжатие) [4]. Программа определяет общие префиксы или суффиксы и их длины записываются таким образом, чтобы избегать дублирования данных. Этот алгоритм хорошо подходит для сжатия отсортированных данных, например, списка слов в словаре.

Пример работы демонстрационной программы в режиме сжатия приведен на Рис.3.

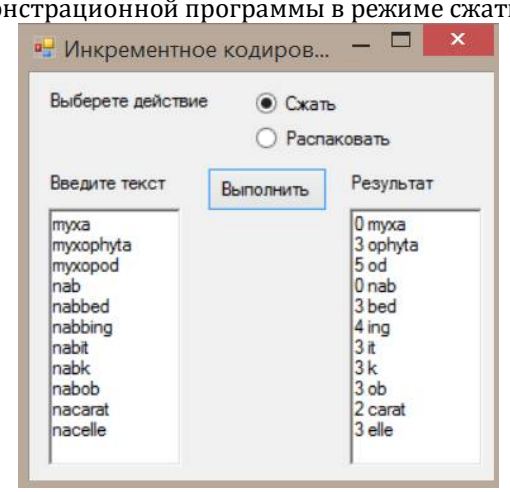

Рис. 3. Пример работы программы

При сжатии считываются слова в массив размерности, определяемой количеством введенных слов (n), после этого, начиная с 0 и до n-1 слова производится посимвольное сравнение слова: 0 и 1 слово, 1 и 2, и так далее до n-3 и n-2 слов. В итоге выводится количество символов которые совпали, а потом оставшиеся символы. У 0 слово индекс 0, так как нам не с чем его сравнивать.

При распаковке сжатый код считывается в массив, где у цифр, которые означают количество данных, четные номера, а у не совпавших букв – нечетные. После этого в переменную запоминается очередное слово, начиная с нулевого. При выводе следующего сначала выводятся совпавшие буквы, а затем дописываются оставшиеся буквы. Данная операция повторяется до n-1 слова (так как в массиве индексация начинается с 0).

3. Сжатие без потерь в соответствии с алгоритмом Лемпеля-Зива-Велча [5]. Интерфейс демонстрационной программы показан на Рис.4. В данном случае приводится пример реализации алгоритма для текстовой информации, при использовании следует учесть, что для большей наглядности следует задавать текст с многократным повтором символов или последовательностей символов в тексте. В противном случае явление сжатия информации может быть незначительным или вообще отсутствовать.

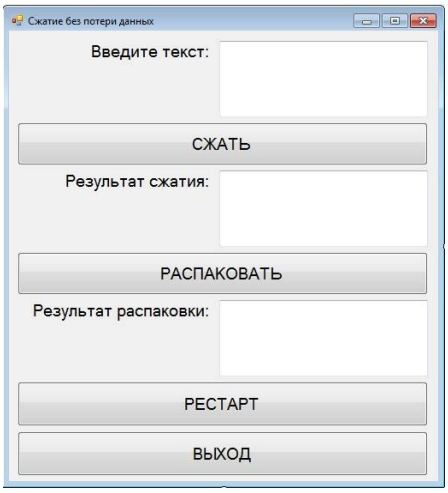

Рис. 4. Интерфейс демонстрационной программы

Вывод

По итогам выполнения поставленной задачи был разработан комплекс обучающих программ по дисциплине «Математические основы защиты информации и информационной

безопасности» для обучающихся в магистратуре по направлению подготовки 02.04.02 (010300) Фундаментальная информатика и информационные технологии. Данный программный продукт предоставляет возможности по организации изучения ряда тем, рассматриваемых в рамках курса. Он может использоваться студентами в качестве сопровождающего обучение средства, допускающее как самостоятельное изучение ряда тем дисциплины, так и проведение самоконтроля. Преподаватель имеет возможность контролировать освоение обучающимися материала тем путем просмотра продемонстрированных результатов тестирования. Также предусматривается обновление или замена теоретического и тестового материала. Список демонстрационных программ допускает расширение. Кроме того, реализована работа комплекса программ в режимах, предназначенных для преподавателя и для обучающегося.

Планируется развитие представленного комплекса обучающих программ, в том числе, в направлении реализации отдельного функционала администратора.

Литература

1. Основные образовательные программы ВГУ. URL ttps://moodle.vsu.ru (дата обращения 20.10.2016).
2. Microsoft Visual Studio [Электронный ресурс]. URL https://msdn.microsoft.com/library/52f3sw5c(v=vs.100).aspx (дата обращения 20.10.2016).
3. SQLite Homepage [Электронный ресурс]. URL http://www.sqlite.org (дата обращения 20.10.2016).
4. Ватолин Д. Методы сжатия данных : Устройство архиваторов, сжатие изображений и видео / Д.Ватолин [и др.] .— М. : Диалог-МИФИ, 2003 .— 381 с.

References

1. Osnovnye obrazovatel'nye programmy VGU. URL https://moodle.vsu.ru (data obrashcheniya 20.10.2016).
2. Microsoft Visual Studio [Elektronnyy resurs] URL https://msdn.microsoft.com/library/52f3sw5c(v=vs.100).aspx.
3. SQLite Homepage [Elektronnyy resurs]. URL http://www.sqlite.org (data obrashcheniya 20.10.2016) .
4. Vatolin D. Metody szhatiya dannykh : Ustroystvo arkhivatorov, szhatie izobrazheniy i video / D.Vatolin [i dr.] .— M. : Dialog-MIFI, 2003 .— 381 s.

Поступила: 2.10.2016

Об авторах:

Крыжановская Юлиана Александровна, старший преподаватель кафедры ERP-систем и бизнес-процессов Воронежского государственного университета, kryzhanovskaya_ya@amm.vsu.ru;

Кашко Василий Васильевич, студент 2 курса магистратуры факультета прикладной математики, информатики и механики Воронежского государственного университета, vasya.kashko@mail.ru.