

УДК 004.4

О РАЗРАБОТКЕ СЕТЕВОЙ АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ ДЛЯ КОНТРОЛЯ ЗНАНИЙ ПО ПРОГРАММИРОВАНИЮ

Антипин А.Ф.

*Стерлитамакский филиал Башкирского государственного университета,
Стерлитамак, e-mail: andrejantipin@ya.ru*

Научно-технический прогресс, произошедший за последние десятилетия, позволил снизить долю ручного труда преподавателей в общем объеме их учебно-методической работы за счет введения в рабочий процесс различных прикладных программ и программных комплексов, активно использующих современные информационные технологии обучения. В статье рассматриваются особенности разработанной автором автоматизированной системы, предназначенной для текущего и итогового контроля знаний, умений и навыков обучающихся в процессе обучения их языкам программирования. Описывается схема и принцип работы системы, структура и содержание таблиц базы данных, а также возможности специального программного обеспечения, устанавливаемого на автоматизированных рабочих местах пользователей. Приводится полный алгоритм тестирования программ, служащий для оценки правильности их написания, начиная от генерации тестовых файлов и заканчивая оценкой результатов тестирования.

Ключевые слова: автоматизированная система, образование, контроль знаний, программирование, тестирование

ABOUT THE DEVELOPMENT OF THE NETWORK AUTOMATED SYSTEM FOR CONTROL OF KNOWLEDGE IN PROGRAMMING

Antipin A.F.

Sterlitamak branch of Bashkir State University, Sterlitamak, e-mail: andrejantipin@ya.ru

Scientific and technical progress which has occurred in recent decades, has allowed to reduce the share of manual labor of teachers in the total amount of their teaching work by introducing into the workflow of various applications and software packages, actively using modern information technologies of training. The article discusses the features of the automated system, developed by the author, for current and final control of knowledge, abilities and skills of students in the learning process of programming languages. Describes the scheme and operation of the system, the structure and content of database tables, as well as the possibilities of special software installed on the workstations of users. Provides the complete algorithm of testing of programs that is used to evaluate the correctness of their writing, ranging from the generation of test files to the evaluation of test results.

Keywords: automated system, education, control of knowledge, programming, testing

В настоящее время, в условиях автоматизации образования, одним из элементов которого является контроль знаний обучающихся, все более актуальной становится разработка соответствующих специальных систем и программных комплексов [1].

Наиболее эффективной формой контроля знаний является теоретическое тестирование, поскольку оно достаточно легко поддается автоматизации. Существует большое разнообразие программ, как платных, так и бесплатных, позволяющих организовать процесс тестирования быстро и с наименьшими затратами.

В то же время для оценки практических навыков обучающихся невозможно ограничиться только этой формой контроля. Так, при обучении программированию и смежным с ним дисциплинам необходимо оценивать не только теоретические знания синтаксиса и семантики изучаемых языков программирования, но и умение обучающихся писать и тестировать программы.

Как правило, такая оценка осуществляется преподавателями аудиторно, что

связано в первую очередь с наличием лицензионного программного обеспечения, установленного на рабочих местах обучающихся, при проведении практических занятий с использованием тестовых данных и может проводиться неоднократно в случае несоответствия получаемых результатов. Все это может существенно замедлять процесс контроля для группы обучающихся. Не говоря уже о дистанционном обучении, где получить консультацию преподавателя в разы сложнее.

Внедрение специальных автоматизированных систем в этот процесс позволит сделать его более эффективным и не зависящим от конкретного преподавателя, что, помимо прочего, исключает предвзятость при проверке заданий, обеспечивает сбор статистики, а также формирование необходимой отчетности в автоматическом режиме [2].

Рассмотрим реализацию подобной автоматизированной системы для контроля знаний по программированию, разработанную автором.

Описание схемы автоматизированной системы

При разработке системы учитывались прежде всего ее простота и скорость развертывания в организациях разного уровня с использованием современных информационно-коммуникационных технологий и вне зависимости от используемого аппаратного обеспечения.

На рис. 1 приведена краткая схема автоматизированной системы для контроля знаний по программированию, предлагаемая автором.

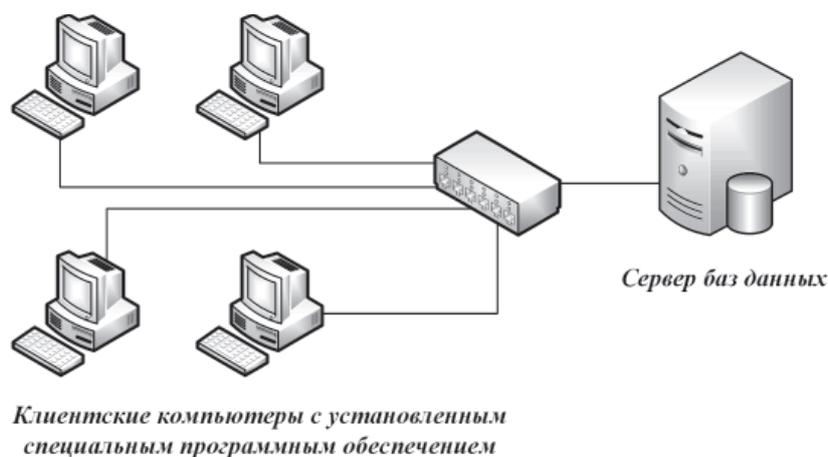


Рис. 1. Краткая схема автоматизированной системы для контроля знаний

Как видно из рис. 1, данная схема является достаточно простой [1] и может использоваться в подавляющем большинстве организаций и учебных учреждений. На сервере баз данных, который может выполнять одновременно и другие функции (например, функции файл-сервера), хранится база данных автоматизированной системы, доступ к которой осуществляется с клиентских компьютеров, подключенных к серверу по локальной сети, при помощи специально разработанного программного обеспечения.

База данных используется, прежде всего, для хранения информации об обучающихся и изучаемых ими учебных дисциплинах, а также тестовых заданий и статистики их прохождений и может быть разработана практически в любой современной системе управления базами данных. В настоящее время наиболее популярной системой для работы с базами данных является Microsoft Access, входящая в пакет Microsoft Office.

Специальное программное обеспечение для работы с этой базой данных, устанавливаемое на клиентских компьютерах, также

может быть разработано с помощью современных средств разработки, предоставляющих возможности для взаимодействия с таблицами баз данных. В настоящее время одной из популярных сред разработки многофункциональных графических приложений и программ под популярные операционные системы, такие как Microsoft Windows, Google Android и пр., является Embarcadero RAD Studio Delphi, которая содержит в своем составе компоненты для работы с базами данных, созданными при помощи разных технологий.

Контроль знаний в данной системе реализуется следующим образом: обучающиеся, находясь в компьютерном классе, подключенном к компьютерной сети учебного учреждения, запускают на своих рабочих местах (клиентских компьютерах) установленное на них специальное программное обеспечение, или приложение, в котором они могут выбрать одно из следующих действий: пройти тест на проверку теоретических знаний и провести проверку написанной ими на одном из изучаемых языков программирования программы, которая служит результатом выполнения той или иной работы.

После прохождения тестов, как теоретических знаний, так и работы разработанных программ, обучающиеся получают баллы, которые в дальнейшем учитываются преподавателями при выставлении итоговой оценки. Заработанные обучающимися баллы хранятся в базе данных и могут быть использованы в целях статистики.

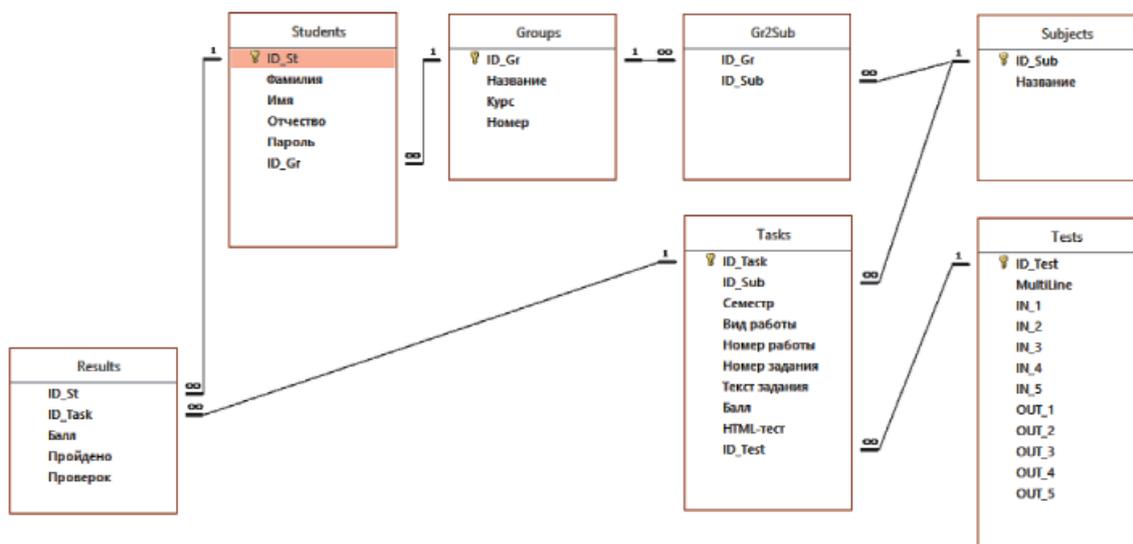


Рис. 2. Схема данных

Описание структуры базы данных

На рис. 2 представлена схема данных разработанной автором базы данных автоматизированной системы контроля знаний.

Как видно из схемы данных на рис. 2, база данных включает в себя минимально необходимый набор таблиц, таких как:

1. Students – содержит основную информацию об обучающихся, такую как фамилия, имя и отчество (при необходимости), а также пароль для доступа к базе данных.

2. Groups – предназначена для хранения основной информации о группах обучающихся, такой как название, курс и номер группы (при обучении двух и более групп на одном курсе).

3. Subjects – используется для хранения списка учебных дисциплин, в рамках которых необходимо осуществлять контроль знаний.

4. Tasks – предназначена для хранения информации о заданиях, предлагаемых к выполнению в зависимости от семестра, в котором изучается та или иная учебная дисциплина, а именно: вид работы (например, контрольная работа, лабораторная работа и т.п.), номер работы и номер задания в рамках данной работы, текст задания (при необходимости), максимальный балл за выполнение задания, а также путь к HTML-файлу с тестом для проверки теоретических знаний по теме работы.

5. Tests – содержит наборы тестовых примеров для проверки правильности разработанных программ.

6. Results – используется для хранения информации (статистики) о результатах деятельности обучающихся, такой как полученный балл за выполнение задания, количество выполненных проверок и количество пройденных тестов при последней проверке.

7. Gr2Sub – предназначена для установки соответствия между группами обучающихся и изучаемых ими учебных дисциплин.

Для установки связей между таблицами базы данных выбран тип отношения *один-ко-многим* с использованием ключевых полей, хранящих уникальные идентификационные номера обучающихся, групп обучающихся, учебных дисциплин и пр. с обеспечением целостности данных при помощи функций автоматического каскадного обновления связанных полей и каскадного удаления связанных записей.

Заполнение таблиц базы данных может осуществляться преподавателями непосредственно через интерфейс Microsoft Access.

Описание работы клиентского приложения

Клиентское приложение автоматизированной системы для контроля знаний по программированию имеет вид, представленный на рис. 3.

В начале работы с приложением пользователь выбирает группу, к которой он относится, и необходимую учебную дисциплину, затем он находит свои данные в выпадающем списке обучающихся в выбранной группе и указывает пароль для доступа к базе данных (при необходимости).

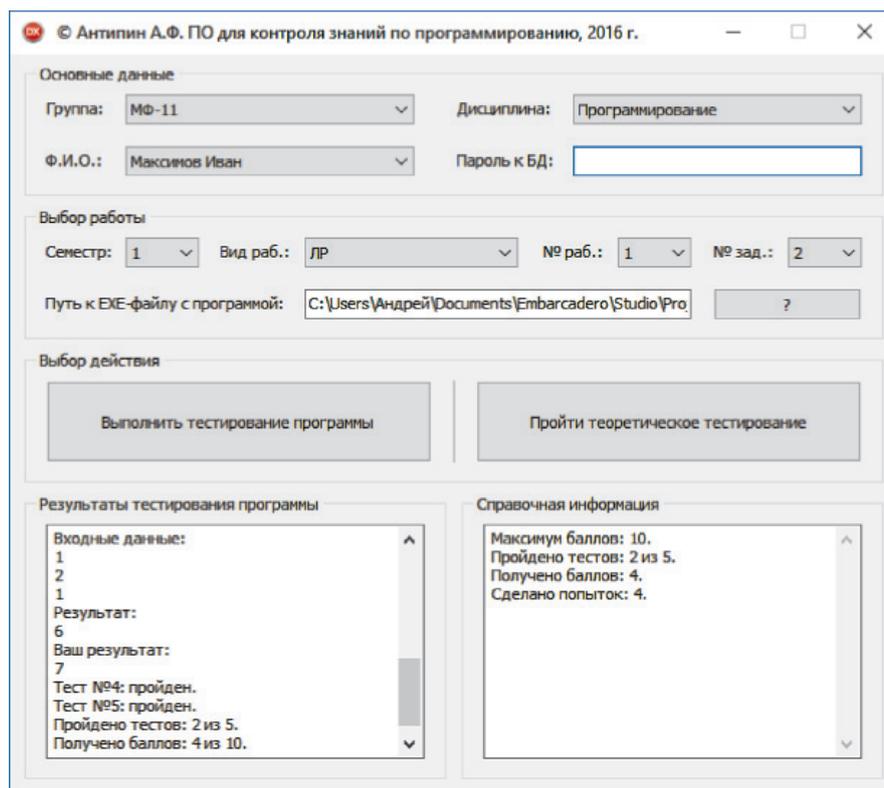


Рис. 3. Главное окно клиентского приложения

Далее ему необходимо выбрать семестр (при изучении дисциплины на двух и более семестрах), вид и номер работы, а также номер задания. После чего ему становятся доступны возможности проверки разработанной им программы, являющейся результатом выполнения задания и прохождения теоретического теста.

Для проверки правильности программы пользователю следует указать путь к ней, после чего запустить процесс тестирования, щелкнув по соответствующей кнопке.

Алгоритм тестирования пользовательской программы включает в себя следующую последовательность этапов:

Этап 1. Генерация тестовых файлов.

На данном этапе производится формирование SQL-запросов к таблице Tests базы данных на выборку тестовых данных. Результат запроса сохраняется в текстовом файле, который служит входным файлом с данными для проверки пользовательской программы. Файл помещается в директорию, которую пользователь указал в качестве места расположения своей программы.

Этап 2. Запуск пользовательской программы.

На данном этапе осуществляется запуск программы пользователя, которая в каче-

стве входных данных использует информацию, записанную в созданном на предыдущем этапе текстовом файле. После чего приложение ожидает создания выходного файла, содержащего результат работы пользовательской программы.

Этап 3. Сравнение полученных результатов.

На данном этапе производится сравнение полученных результатов с информацией, содержащейся в таблице Tests базы данных. Если они идентичны, то пользователю засчитывается успешное прохождение теста, иначе в соответствующее поле приложения выводится сообщение о том, что текущий тест не пройден, а также приводятся сведения об использованной тестовой выборке и требуемом результате для того, чтобы пользователь мог найти ошибки в программе [3, 4].

По окончании данного этапа приложение осуществляет проверку того, все ли тесты были использованы. При наличии непройденных тестов осуществляется переход к этапу 1, иначе переход к этапу 4.

Этап 4. Оценка результатов тестирования.

На данном этапе осуществляется окончательная оценка результатов проверки пользовательской программы с подсчетом итогового числа баллов [5] и выдачей информации о количестве успешно пройденных тестов.

Результаты последнего тестирования сохраняются в базе данных в таблице Results.

На рис. 4 представлена краткая блок-схема алгоритма работы подпрограммы тестирования, реализованного в приложении.

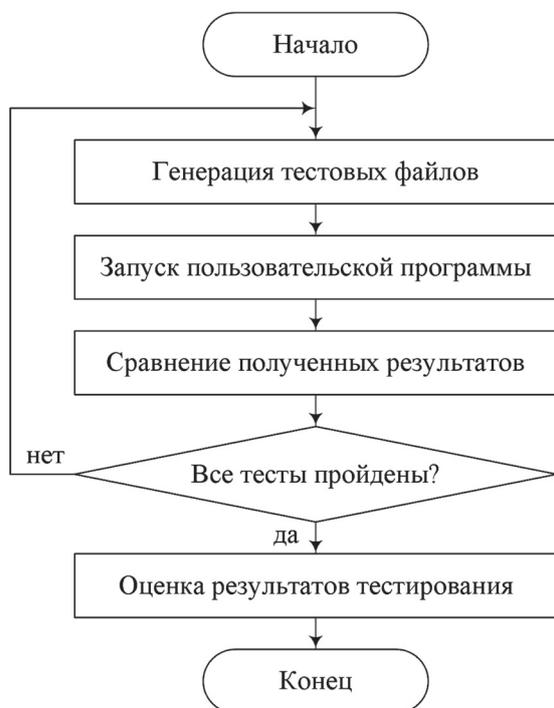


Рис. 4. Блок-схема алгоритма работы подпрограммы тестирования

Также в соответствующем поле приложения отображается справочная информация о результате предыдущей проверки программы, максимальном балле за задание, а также сведения об общем количестве проведенных проверок.

При необходимости пользователь также может пройти тест для проверки теоретических знаний по теме работы. При нажатии на соответствующую кнопку приложение открывает связанный с заданием HTML-файл. HTML-файлы с тестами могут быть разработаны в любом современном HTML-редакторе (при наличии соответствующих знаний и навыков), а также при помощи специального программного обеспечения и веб-ресурсов для разработки тестов в формате языка гипертекстовой разметки HTML.

Заключение

В заключение необходимо отметить, что при достаточно большом разнообразии систем и программных комплексов, позволяющих осуществлять контроль знаний, предлагаемая автором автоматизированная система содержит весь необходимый набор функций для тестирования разработанных программ и оценки теоретических знаний в рамках обучения программированию и может быть развернута практически в любой организации и учебном учреждении, где имеется компьютерный класс, подключенный к локальной сети.

При отсутствии последних, а также при необходимости передачи клиентского приложения обучающимся для осуществления проверки разработанных программ вне аудитории, имеется возможность локального доступа к базе данных. Для этого достаточно поместить ее в ту же директорию, что и исполнительный файл.

Дальнейшее развитие клиентского приложения предполагает его интеграцию с разработанным автором клиент-серверным программным комплексом [2] для автоматизации учебной и учебно-методической работы преподавателей, существенно расширяющем возможности ведения групп обучаемых, разработки электронных учебных пособий и организации тестирования, а также выдачи различных форм отчетности при необходимости реализации рейтинговой системы обучения.

Список литературы

1. Антипин А.Ф. Организация эффективной работы преподавателей в условиях рейтинговой системы обучения // Прикладная информатика. – 2014. – № 3. – С. 48–59.
2. Антипин А.Ф. Система автоматизированной разработки учебно-методических комплексов на основе многомерных логических регуляторов // Программные продукты и системы. – 2011. – № 2. – С. 119–122.
3. Антипин А.Ф. О проверке программ автоматизированных систем управления // Современная техника и технологии. – 2015. – № 1. – С. 62–66.
4. Антипин А.Ф. К вопросу о семантическом анализе программ автоматизированных систем управления // Информационные системы и технологии. – 2015. – № 5. – С. 45–52.
5. Антипина Е.В., Антипин А.Ф. Применение интеллектуальных технологий для анализа многомерных данных // Молодой ученый. – 2014. – № 19. – С. 172–175.