

Информационное и программное обеспечение ИВС КТ

Нардюжев Виктор Иванович — доцент кафедры информационно-компьютерных дисциплин Обнинского института атомной энергетики, кандидат технических наук. E-mail: vin@kaluga.ru
Нардюжев Иван Викторович — доцент кафедры автоматизированных систем управления Обнинского института атомной энергетики, кандидат технических наук. E-mail: vin@kaluga.ru

Автоматизированная компоновка компьютерных тестов

Исходные материалы поступают от авторов в виде тестовых заданий, содержащих формулировки вопросов, варианты ответов и реквизиты: номер задания в тесте, номер правильного ответа, весовой коэффициент (например, определённый экспертом уровень трудности).

Ввод и оформление полученных материалов выполняются с помощью редактора Microsoft Word, который позволяет:

- использовать форматированный текст, списки и таблицы;
- создавать сложные математические выражения и формулы;
- вставлять схемы и рисунки, подготовленные в графических пакетах;
- применять шаблоны, импортировать и экспортировать текст и графику;
- программировать макрокоманды на языке WordBasic для выполнения сложных или часто повторяющихся операций;
- расставлять закладки для перестановки ответов при компоновке тестов;
- проверять орфографию, искать в тексте по ключевым словам;
- формировать контрольные распечатки банка.

С помощью редактора Microsoft Word каждое тестовое задание вводится в отдельный документ, имя которого формируется по шаблону НОМ-О-ВК.doc, где НОМ — номер задания в банке (от 001 до 999). О — номер правильного ответа (от 0 до 9 и от А до Z), ВК — весовой коэффициент (от 01 до 99). Документ сохраняется в папке на диске в зависимости от номера задания в тесте.

Размещение документов с тестовыми заданиями о дисциплине, имеющей код 03, будет следующим.

Число заданий в тесте определяет количество папок в каталоге 03. В папке 03\001 хранятся все кандидаты на роль первого задания в тесте, получившие порядковые номера в банке от 001 до 015.

Схема допускает два варианта хранения реквизитов тестовых заданий:

- объектно-ориентированный — тестовое задание лежит в отдельном файле, содержащем все необходимые реквизиты;
- реляционный — реквизиты находятся в специальной таблице и связаны с тестовыми заданиями по ключевому полю.

Объектно-ориентированный вариант является логически ясным, но очень медленным, так как предполагает перебор большого количества отдельных файлов. Реляционный вариант быстрее, но требует специальных механизмов для поддержания целостности связей. Оба подхода позволяют:

- обрабатывать исходный авторский материал;
- вносить необходимые изменения и дополнения;
- назначать и поддерживать реквизиты тестовых заданий;
- одновременно работать разным сотрудникам с одним банком;
- выполнять автоматизированную компоновку компьютерных тестов;
- проводить анализ результатов тестирования и быстро переходить от цифры в отчёте к стоящему за ней тестовому заданию;
- хранить, переносить в архив, восстанавливать и передавать по каналам связи весь банк тестовых заданий или отдельные его части.

Схема размещения документов с тестовыми заданиями, тесно связанная с процедурой автоматизированной компоновки компьютерных тестов, состоит в следующем:

- a) определяется дисциплина, по которой нужно компоновать тесты;
- b) делается рабочая копия соответствующей части банка тестовых заданий;
- c) из полученной копии удаляются все задания, не удовлетворяющие критерию (например, не прошедшие апробацию или слишком трудные);
- d) запускается программа Компоновщик, созданная авторами работы в среде визуального программирования Delphi 3.0;
- e) из списка выбирается нужная дисциплина, указываются необходимое количество тестов и номер первого из них;
- f) Программа Компоновщик анализирует содержимое соответствующей части банка и регистрирует реквизиты всех имеющихся документов с тестовыми заданиями. При обнаружении ошибки в реквизитах на экран выдаётся сообщение и компоновка прерывается;
- g) в результате регистрации получают $M > 0$ списков по одному на каждое задание с номером $m = 1 \dots M$, содержащие по $N(m) > 0$ кандидатов на роль m -го задания в тесте.
- h) на основе кода выбранной дисциплины и номера первого компонуемого теста вычисляется начальное значение для генератора псевдослучайных чисел, что позволяет в случае необходимости при неизменных стартовых условиях повторять процесс компоновки и получать идентичные тесты;
- i) начинается компоновка $T > 0$ тестов с относительными номерами $t = 1 \dots T$;

$$mesm(t) = \{M(m, n(m, t) = \left[\begin{array}{l} R \bmod N(m) + 1, \text{если } (t - 1) \bmod N(m) = 0 \\ n(m, t - 1) + 1, \text{если } (t - 1) \bmod N(m) \neq 0 \end{array} \right] \},$$

где R — очередное значение генератора псевдослучайных чисел.

Как видно из математической формулы, каждый тест является набором из M -заданий, причём на роль m -го задания претендуют только элементы m -го списка. Из числа $N(m)$ кандидатов выбирается задание, реквизиты которого хранятся под номером $n(m, t)$. Для первого теста значение $n(m, t)$ определяется с помощью генератора псевдослучайных чисел. Для следующих тестов значение $n(m, t)$ увеличивается на единицу по модулю $N(m)$. Это позволяет программе последовательно перебрать все имеющиеся тестовые задания. При компоновке теста с относительным номером $t = N(m) + 1$ происходит очередное обращение к генератору псевдослучайных чисел и начинает формироваться новая последовательность заданий.

Предложенный алгоритм даёт возможность компоновать любое количество тестов $T > 0$ с любым числом заданий $M > 0$, при этом равномерно использует все имеющиеся в банке тестовые задания и уменьшает вероятность параллельных повторов в тестах при увеличении числа кандидатов $N(m) > 0$. Алгоритм является детерминированным и при неизменных стартовых условиях (код дисциплины, содержимое банка, количество тестов и номер первого из них) компоует идентичные тесты, используя метод случайной перестановки типовых тестовых заданий. Содержательная структура близких по характеристикам тестов при этом не нарушается, так как любое задание сохраняет свою позицию в тестах, меняясь только с ему аналогичными.

Для повышения секретности тестов дополнительно применяется случайная перестановка $V > 0$ вариантов ответа на задание с помощью упорядоченного списка P из $V!$ элементов от $\{1\}\{2\} \dots \{V-1\}\{V\}$ до $\{V\}\{V-1\} \dots \{2\}\{1\}$. При каждом использовании тестового задания на единицу увеличивается связанный с ним счётчик u , определяющий в списке P индекс нужной перестановки $p(u)$:

$$p(u) = \left[\begin{array}{l} R \bmod (V!) + 1, \text{если } (u - 1) \bmod (V!) = 0 \\ p(u - 1) + 1, \text{если } (u - 1) \bmod (V!) \neq 0 \end{array} \right]$$

где R — очередное значение генератора псевдослучайных чисел.

В конце работы программы Компоновщик появляется таблица, в которой для каждого теста указаны использованные тестовые задания и применённые перестановки. Ниже показана такая таблица для случая компоновки по дисциплине с кодом 03 двух тестов с номерами

001 и 002 (содержимое файла 03001002.dsc, сокращённое и представленное в виде двух колонок одна под другой — прим. сост. эл. версии).

* 001

.\001\008-5-01.doc, 45321, 2
.\002\014-2-01.doc, 15234, 3
.\003\001-2-02.doc, 42315, 2
.\004\007-2-01.doc, 52431, 2
.\005\010-1-01.doc, 23451, 5

.....

.\031\003-2-01.doc, 13245, 3
.\032\004-1-01.doc, 23451, 5
.\033\007-4-02.doc, 21354, 5
.\034\007-2-02.doc, 54132, 5
.\035\015-2-02.doc, 54312, 5

#

* 002

.\001\009-2-01.doc, 14532, 5
.\002\015-1-01.doc, 34521, 5
.\003\002-2-01.doc, 13524, 4
.\004\008-1-01.doc, 14235, 1
.\005\011-3-01.doc, 54321, 3

.....

.\031\004-4-01.doc, 15324, 5
.\032\005-3-01.doc, 32514, 1
.\033\008-3-02.doc, 45312, 3
.\034\008-3-01.doc, 13542, 2
.\035\001-1-01.doc, 34521, 5

#

Приведённая выше часть процедуры автоматизированной компоновки тестов является общей для бланкового и компьютерного тестирования. Например, можно запустить макрос, написанный авторами работы на языке WordBasic, указать на входе созданную программой Компоновщик таблицу с описанием тестов (файл с расширением dsc) и получить на выходе для каждого теста отдельный документ в формате редактора Microsoft Word, содержащий заголовок с названием дисциплины и номером теста, все перечисленные в таблице тестовые задания, автоматически пронумерованные по порядку, и колонтитулы со служебной информацией и номерами страниц.

Сформированные макросом документы выполняют роль оригинал-макетов бумажных тестов, готовых к печати на принтере или к тиражированию в типографии. Для подготовки компьютерного варианта этих тестов необходимо воспользоваться созданной в среде визуального программирования Delphi 3.0 программой Камера, взаимодействующей по протоколу DDE с одноимённым макросом, написанным на языке WordBasic.

Макрос получает от пользователя имя файла, в котором хранится сформированная программой Компоновщик таблиц с описанием тестов, открывает указанный файл, считывает реквизиты первого задания первого теста, выводит на экран содержимое соответствующего документа и посылает программе Камера по каналу DDE извещение.

Программа принимает извещение, «схватывает» графический образ тестового задания, сжимает его и заносит в базу данных вместе с другими реквизитами. Закончив обработку, программа Камера отправляет макросу подтверждение, что позволяет ему продолжить свою работу и перейти к следующему заданию в таблице с описанием тестов.

Все реквизиты тестовых заданий сохраняются в базе данных, состоящей из четырёх связанных таблиц: Subject.db, Test.db, Task.db и Question.db, структуры которых представлены в таблицах 1, 2, 3 и 4 соответственно.

Структура базы данных программы Камера

Subject.db	Test.db	Task.db	Question.db
Subject	Subject	Subject	Subject

Comment	Test Comment	Test Task Question	Task Question Answer Type Image
---------	-----------------	--------------------------	---

Таблица 1. Структура таблицы Subject.db

№	Поле	Тип	Размер	Ключ	Назначение
1	Subject	S	2	*	Код предмета
2	Comment	A	50		Полное название предмета

Таблица 2. Структура таблицы Test.db

№	Поле	Тип	Размер	Ключ	Назначение
1	Subject	S	2	*	Код предмета
2	Test	S	2	*	Номер теста
3	Comment	A	50		Комментарий к тесту

Таблица 3. Структура таблицы Task.db

№	Поле	Тип	Размер	Ключ	Назначение
1	Subject	S	2	*	Код предмета
2	Test	S	2	*	Номер теста
3	Task	S	2	*	Номер задания в тесте
4	Question	S	2		Номер задания в банке

Таблица 4. Структура таблицы Question.db

№	Поле	Тип	Размер	Ключ	Назначение
1	Subject	S	2	*	Код предмета
2	Task	S	2	*	Номер задания в тесте
3	Question	S	2	*	Номер задания в банке
4	Answer	A	5		Номер правильного ответа
5	Type	S	2		Тип задания
6	Image	G	~2048		Графический образ задания

Помимо обработки извещений, посылаемых макросом, программа Камера предоставляет возможность просматривать и изменять (при необходимости) полное название предмета и комментарий к тесту, список заданий в тесте, номер правильного ответа и тип выбранного задания. Просмотр графических образов позволяет заранее увидеть, как тестовые задания будут выглядеть на экране во время проведения сеанса компьютерного тестирования.

При обнаружении ошибок в тестовых заданиях из таблицы Question. db удаляются соответствующие записи, редактируются исходные документы и повторно запускаются макрос и программа Камера, чтобы получить новые графические образы, не содержащие ошибки, для всех заданий, перечисленных в таблице с описанием тестов, но отсутствующих в таблице Question.db.

В любой момент базу данных программы Камера можно очищать и дополнять новыми предметами и тестами, формируя целевые пакеты. Например, для ознакомления участников с процедурой компьютерного тестирования в пакет целесообразно включать по одному тесту каждого предмета, а при проведении аттестации, наоборот, требуется пакет из десяти тестов одного предмета. Программы, ответственные за проведение сеансов компьютерного тестирования, имеют соответствующие функции для работы с полученными пакетами.

Оценить размеры базы данных программы Камера можно по цифрам из таблицы 5, содержащей данные репетиционного тестирования через Интернет, для которого использовался банк из 5 кандидатов на каждое задание теста и компоновались 10 тестов по каждому из 28 предметов.

Таблица 5. Характеристики базы данных, использованной в репетиционном тестировании через Интернет

Имя таблицы	Содержимое	Число записей	Размер, Кб
Subject.db	Список предметов	28	8
Test.db	Список тестов	280	22
Task.db	Список заданий	10640	90

Видно, что размер базы данных определяет таблица Question.db, в которой хранятся графический образ и реквизиты тестового задания, включённого через таблицу Task.db в разные тесты. Использование ссылок на единственный экземпляр тестового задания даёт возможность компоновать из небольшого банка заданий любое количество тестов, практически не увеличивая размер итоговой базы данных.

Представленное выше информационное и программное обеспечение подсистемы автоматизированной компоновки компьютерных тестов позволяет:

- использовать банк бланкового тестирования, насчитывающий сегодня 45 000 тестовых заданий с множественным выбором по 31 предмету, для компоновки компьютерных тестов;
- хранить тестовые задания в виде чёрно-белых графических образов, объём которых после сжатия составляет в среднем 2 Кбайта, что даёт возможность передавать их через Интернет по каналам с небольшой пропускной способностью;
- формировать целевые пакеты, содержащие различные наборы предметов и тестов в зависимости от целей тестирования.

Предложенный вариант автоматизированной компоновки компьютерных тестов не является единственным. Альтернативным может быть создание тестов в открытом формате HTML, который позволяет использовать форматированный текст, графику, анимацию, звук, видеофрагменты и активные элементы для оформления тестовых заданий любого типа (с множественным выбором, с множественным ответом, с дополнением, со свободным ответом, на установление соответствия и на выявление последовательности). Однако в этом случае не удастся без существенных переделок воспользоваться имеющимся банком тестовых заданий для бланкового тестирования.

Размер тестовых заданий при переходе на текстовый формат HTML, как правило, уменьшается. Но число файлов может увеличиться, так как каждая картинка (схема, рисунок или математическая формула) хранится отдельно.

В последнее время появилось множество инструментальных средств, таких, как Microsoft FrontPage, обеспечивающих при работе с документами в формате HTML проверку орфографии и поиск по ключевым словам. Постепенно унифицируются средства просмотра, но все они работают с открытым форматом, что не всегда приемлемо с точки зрения секретности тестов.

Проведение сеансов тестирования на компьютерах

При создании информационного и программного обеспечения подсистемы проведения сеансов тестирования на компьютерах учитывались результаты:

- анализа и классификации современных систем компьютерного тестирования, разработки требований к ИВС КГ и определения показателя её эффективности;
- создания концептуальной модели системы, разработки аналитической модели блока обработки заявок, поступающих на тест-сервер, и имитационной модели телекоммуникационной подсистемы, проведения на их основе исследований работоспособности различных вариантов реализации ИВС КГ;
- разработки и исследования центральных алгоритмов взаимодействия рабочих станций и тест-сервера в рамках ИВС КГ.

Рассмотрим программу Оператор, разработанную авторами этой статьи в среде визуального программирования Delphi 3.0. Она обеспечивает проведение тестирования за компьютером через Интернет непосредственно на сервере центра тестирования по тестам, скомпонованным из банка тестовых заданий централизованного тестирования на бланках.

Программа Оператор работает под управлением операционной системы Windows 95 (русская версия), поставляется в виде автономного исполняемого модуля, который копируется в отдельную папку и занимает на жёстком диске примерно 325 Кбайт. Других программных компонент не требуется, так как для доступа к тест-серверу по протоколу TCP

используется стандартная для Windows сетевая библиотека WinSock. При этом предполагается, что у тестируемого уже есть настроенный выход в Интернет через обычный модем или локальную сеть.

Программа Оператор реализует алгоритм функционирования рабочей станции в режиме оперативной обработки ответов тестируемых и даёт возможность последовательно:

- получить список предметов и выбрать учебную дисциплину;
- получить перечень доступных тестов и выбрать нужный номер;
- получить структуру теста и выбрать интересующий раздел;
- получить тестовое задание и приступить к его выполнению;
- закончив работу, проверить ответы и подсчитать набранные баллы.

Все эти операции предполагают обмен данными между тест-сервером и рабочей станцией через Интернет по протоколу TCP, при этом пользователю программы доступна кнопка, позволяющая прервать сеанс связи с сервером, не дожидаясь его окончания.

Основной информационный поток идёт от тест-сервера к рабочей станции, так как именно в этом направлении пересылаются сжатые графические образы тестовых заданий, количество которых в отдельных тестах достигает 60, а средний размер составляет 2 Кбайта.

В процессе работы с тестом программа создаёт в папке, где лежит исполняемый модуль, подкаталог, имя которого совпадает с индивидуальным номером сеанса, и использует его в качестве дискового буфера для хранения графических образов тестовых заданий и различных списков, загруженных с тест-сервера через Интернет. При завершении работы с тестом подкаталог очищается и удаляется, чтобы освободить место на диске.

В большинстве заданий требуется выбрать один правильный ответ из предложенного набора, в некоторых вопросах необходимо отобрать все удовлетворяющие условию варианты (например, в задании по русскому языку: «Укажите слова с окончанием Е»).

Перейти к другому заданию теста можно либо через структуру теста, либо используя полосу прокрутки, расположенную над кнопками для выбора правильного ответа. Над полосой прокрутки выводится порядковый номер задания в списке, под ней — количество заданий в тесте.

При этом в правом нижнем углу окна программы в строке состояния работает секундомер, показывая затраченное на выполнение всего теста время. В соседней ячейке указаны номер задания в тесте и результат его последней проверки. В строке состояния выводится код и название предмета, номер и официальная продолжительность теста.

После выполнения заданий теста и возврата к его структуре можно проверить ответы и подсчитать баллы. Программа сообщит количество правильно решённых заданий и число набранных тестовых баллов. Напротив каждого задания в столбце «Ответ» появится либо знак «+» (правильный ответ), либо знак «-» (неправильный ответ). При желании список заданий можно отсортировать по столбцу «Ответ», тогда неправильно выполненные задания сгруппируются и будут следовать одно за другим.

Программа Оператор позволяет в учебных целях после получения результатов вернуться к неправильно выполненным заданиям для выявления и исправления допущенных ошибок или приступить к выполнению пропущенных ранее заданий (при аттестации такая возможность блокируется).

Для минимизации сетевого трафика пользователь может работать только с интересующими тестовыми заданиями, отобранными через структуру теста, не загружая с сервера ничего лишнего. Увидев на рабочем экране текст нужных заданий (загрузив их тем самым в дисковый буфер), можно отключиться от сети (если оплачивается время подключения к Интернет) и спокойно выполнять отобранные задания столько времени, сколько это необходимо для подготовки. Следующее подключение к Интернет потребуется только для проверки ответов и подсчёта набранных баллов или для подгрузки заданий другого раздела теста.

Если необходимо выполнить тест целиком, то следует загрузить весь тест на локальный диск, воспользовавшись кнопкой с изображением дискеты. Программа «пробежит» по всем тестовым заданиям и подгрузит недостающие вопросы с сервера в дисковый буфер. Вся

дальнейшая работа с тестом будет идти в рамках компьютера клиента без обращений к серверу через Интернет (за исключением проверок ответов и подсчёта набранных баллов).

Работу с тестом можно приостановить (кнопка с изображением паузы — два вертикальных прямоугольника). Программа сообщит, что выполнение теста можно будет продолжить, используя индивидуальный номер сеанса. Специальных ограничений на продолжительность пауз не устанавливается. Приостановив выполнение одного теста, можно начать выполнение другого. При желании в дисковый буфер можно подгрузить несколько тестов и работать с ними по очереди. Сеансы тестирования независимы друг от друга и различаются по индивидуальным номерам, что позволяет организовать на одном компьютере или на сервере локальной сети групповое тестирование.

Однако для группового тестирования в локальной сети лучше использовать программу Тестер, разработанную авторами работы в среде визуального программирования Delphi 3.0 для работы под управлением операционной системы Windows 95 (русская версия). Помимо сетевой библиотеки WinSock программа использует Borland Database Engine — набор библиотек для доступа к базам данных, поэтому установленный комплект занимает на диске примерно 2,5 Мбайта, не считая самих тестов и информации о сеансах тестирования.

Возможны локальная и сетевая установки. При локальной установке в выбранную папку, которая может размещаться как на локальном, так и на сетевом диске, копируются все необходимые файлы (программные модули, документация, базы данных и т.д.), а в системном реестре регистрируются соответствующие ссылки. Этот вариант предназначен для установки программы на автономный компьютер и для первой установки в локальной сети на общедоступный диск.

После выполнения первой установки остальные компьютеры локальной сети могут воспользоваться её результатами. На них достаточно выполнить сетевую установку, выбрав папку на общедоступном диске с установленным комплектом. При этом выполняется только регистрация соответствующих ссылок в реестре очередного компьютера (рабочей станции). Сетевая установка позволяет совместно использовать единственный комплект программных модулей и баз данных с тестами, что экономит дисковое пространство и упрощает обновление версий, а также позволяет вести единую базу по тестируемым — это облегчает контроль со стороны инструктора и формирование протоколов тестирования для отправки на обработку.

При запуске программы Тестер на экране появляются списки предметов и тестов, загруженных в базу данных из пакетов, созданных программой Камера. После выбора предмета и теста появляется диалог, предлагающий участнику тестирования пройти регистрацию.

При самоконтроле регистрация, как правило, отсутствует. При аттестации в масштабах страны она превращается в полноценное анкетирование. Перечень регистрируемых данных определяется организаторами тестирования, которые хотят собрать как можно больше полезной информации, способной помочь им интерпретировать получаемые результаты.

Структура соответствующей таблицы User.db (формат Paradox), в которой программа Тестер хранила данные об участниках компьютерного тестирования, проведённого в компьютерных классах региональных представительств Центра тестирования выпускников общеобразовательных учреждений РФ в 1998 году, показана в таблице 6.

Таблица 6. Структура таблицы User.db

№	Поле	Тип	Размер	Ключ	Назначение
1	User	+	4	*	Индивидуальный номер участника
2	Predmet	S	2		Код предмета
3	Test	S	2		Номер теста
4	Region	S	2		Код региона
5	Raion	S	2		Код района
6	Punkt	S	2		Код населённого пункта
7	Shkola	S	2		Номер школы
8	Tip	S	2		Тип учебного заведения
9	Pol	S	2		Пол участника
10	Rezerv	S	2		Резервное поле
11	FIO	A	50		Ф.И.О. Участника
12	Seria	A	10		Серия документа

13	Pasport	I	4	Номер документа
14	Task	S	2	Номер текущего задания
15	Time	I	4	Время выполнения теста
16	Mode	S	2	Статус участника

После регистрации начинается выполнение тестовых заданий, графические образы которых, в отличие от программы Оператор, загружаются не с удалённого тест-сервера через Интернет, а с общего диска локальной сети.

Выполнение задания заключается в выборе правильного ответа. Для этого используются кнопки с цифрами, расположенные на экране программы справа. Ответы всех тестируемых хранятся вместе в отдельной таблице Answer.db, структура которой представлена ниже в таблице 7.

Таблица 7. Структура таблицы Answer.db

№	Поле	Тип	Размер	Ключ	Назначение
1	User	I	4	*	Индивидуальный номер участника
14	Task	S	2	*	Номер задания в тесте
15	Answer	A	5		Ответ тестируемого

Переход между заданиями теста осуществляется с помощью полосы прокрутки, над которой выводится номер текущего задания, а под ней — количество заданий в тесте. При этом в правом нижнем углу экрана работает секундомер, показывая время, затраченное на выполнение теста, номер и официальная продолжительность которого видны в строке состояния рядом с кодом и полным названием предмета.

Работу с тестом можно в любой момент приостановить или завершить по желанию тестируемого с разрешения инструктора. В первом случае выполнение теста можно будет продолжить, указав индивидуальный номер участника. Программа Тестер восстановит сеанс, используя запись в таблице User.db. Отсчёт времени будет идти с момента приостановки (при техническом сбое — с точностью до минуты), а на экране появится последнее тестовое задание, выполненное тестируемым.

Существенным отличием от программы Оператор, ориентированной на индивидуальное тестирование через Интернет непосредственно на удалённом тест-сервере, является наличие режима инструктора, в рамках которого можно: визуально наблюдать за работой тестируемых в локальной сети, загружать в базу данных на общем диске новые пакеты тестов, формировать протоколы и отправлять их на тест-сервер для отложенной обработки.

Доступ в режим инструктора защищается паролем, назначенным в момент установки программы Тестер. Успешный ввод пароля позволяет увидеть список тестируемых из таблицы User.db, в котором для каждого участника указаны индивидуальный номер, Ф.И.О., код предмета, номер теста, затраченное время и текущий статус, который может принимать следующие значения: «Выполняет тест», «Приостановил работу», «Завершил работу», «Готовится к отправке», «Отправлен на обработку».

Инструктор может удалить из списка любые записи, которые он считает лишними, при этом особая осторожность нужна по отношению к участникам, которые в данный момент работают с тестом и имеют статус «Выполняет тест». Чтобы увидеть последние изменения в списке при работе в локальной сети, инструктор может воспользоваться кнопкой с изображением бинокля. Если необходимо, то можно изменить статус любого участника (две кнопки со стрелками влево и вправо).

Программа Тестер позволяет динамически загружать новые пакеты тестов, сформированные программой Камера и полученные через WWW или FTP сервер, по электронной почте, на дискетах или компактных дисках. Каждый пакет является Zip-архивом, содержащим следующие таблицы: Subject.db, Test.db, Task.db и Question.db.

В процессе загрузки пакета программа распаковывает архив во временном каталоге и добавляет все записи из полученных таблиц в свою базу данных, расположенную, как правило, на общем диске в локальной сети. Сразу после успешной загрузки пакета новые тесты становятся доступными для выбора и выполнения участниками тестирования, а временный

каталог очищается, чтобы освободить место на диске.

Для реализации отложенной обработки ответов тестируемых в программе Тестер есть возможность формировать протоколы компьютерного тестирования и отправлять их тест-серверу. При этом используется информация из таблиц User.db и Answer.db, хранящих соответственно реквизиты и ответы участников тестирования. Отбираются все записи, имеющие статус «Завершил работу» или «Готовится к отправке». При успешном окончании операции они получают статус «Отправлен на обработку».

Протокол формируется по команде инструктора и является текстовым файлом с разделителями, который после упаковки передаётся через Интернет удалённому тест-серверу в соответствии с рассмотренным в предыдущей главе алгоритмом функционирования рабочей станции в режиме отложенной обработки ответов тестируемых.

В качестве ответа тест-сервера программа Тестер получает упакованный текстовый файл с разделителями, содержащий результаты проверки ответов и подсчёта набранных баллов каждого участника тестирования, попавшего в протокол. Этот файл готов к подгрузке в таблицу StatInfo.db для проведения статистического анализа результатов тестирования.

Примером реализации алгоритма функционирования тест-сервера, обслуживающего множество рабочих станций в рамках ИВС КТ, может быть программа ТСервер, созданная авторами этой статьи в среде визуального программирования Delphi 3.0 для работы под управлением операционной системы Windows 95 (русская версия). Помимо стандартной сетевой библиотеки WinSock программе необходим Borland Database Engine — набор библиотек для доступа к базам данных.

Согласно алгоритму при старте программа ТСервер открывает следующие таблицы:

- с описанием тестов и заданий (Subject.db, Test.db, Task.db, Question.db);
- с реквизитами и результатами участников тестирования (StatInfo.db).

Первые четыре таблицы, созданные программой Камера, используются для:

- формирования списков имеющихся в банке предметов и тестов;
- получения графического образа запрошенного тестового задания;
- сравнения ответов с ключами и подсчёта набранных баллов.

Таблица StatInfo.db содержит полученные от рабочих станций реквизиты тестируемых, собранные в ходе регистрации, их ответы, выбранные в процессе выполнения тестовых заданий, и результаты, рассчитанные на тест-сервере.

Структура таблицы StatInfo.db зависит от целей тестирования и типов тестовых заданий. В качестве примера в табл. 8 представлен вариант, использованный в 1999 году для централизованного тестирования выпускников общеобразовательных учреждений РФ по тестам, в которых присутствовали задания с множественным выбором, с множественным ответом, с дополнением.

Таблица 8. Структура таблицы StatInfo.db

№	Поле	Тип	Размер	Ключ	Назначение
1	N	+	4	*	Индивидуальный номер участника
2	Anketa	I	4		Номер анкеты
3	Strana	S	2		Код страны
4	Region	S	2		Код региона
5	Organ	S	2		Код организации,.
6	Punkt	S	2		Код пункта тестирования
7	Vid	S	2		Вид тестирования
8	Predmet	S	2		Код предмета
9	Test	S	2		Номер теста
10	Indeks	I	4		Почтовый индекс
11	Shkola	S	2		Номер школы
12	Tip	S	2		Тип учебного заведения
13	Sobstv	S	2		Вид собственности
14	Reserv_2	S	2		Резервное поле №2
15	Pol	S	2		Пол участника
16	Reserv_1	S	2		Резервное поле №1
17	Fio	A	50		Ф.И.О. участника
18	Seria	A	10		Серия документа

19	Nomer	A	10	Номер документа
20	Ball	S	2	Набранный балл
21	Procent	S	2	Набранный процент
22	Razdel_1	S	2	Балл по разделу №1
23	Razdel_2	S	2	Балл по разделу №2
24	Razdel_3	S	2	Балл по разделу №3
25	Otvet	A	250	Ответы тестируемого
26	Proverka	A	60	Результаты проверки
27	Status	S	2	Текущий статус участника
28	Client	A	4	Код представителя
29	Protocol	A	8	Номер протокола
30	Stamp	@	8	Дата и время обработки

Открыв таблицы и создав очередь заявок, программа ТСервер начинает прослушивать известный клиентам ТСР-порт, ожидая запросов на установление соединения. Получив соответствующий сигнал, программа открывает канал, читает запрос рабочей станции и регистрирует его в списке на экране и в журнале на диске, фиксируя параметры, IP-адрес клиента, дату и время поступления на тест-сервер.

При обслуживании запроса все операции, требующие много времени, выполняются в асинхронном режиме, позволяя тест-серверу организовать на единственном процессоре псевдопараллельную обработку заявок от разных клиентов. При работе с протоколом после обслуживания очередной записи с информацией о сеансе тестирования текущий запрос становится в конец очереди, из которой выбирается первая заявка. Это даёт возможность избежать значительного увеличения времени пребывания оперативных запросов в очереди при обработке больших протоколов.

Результат обслуживания запроса регистрируется в списке на экране и в журнале на диске. По счётчикам в строке состояния можно судить о количестве успехов, ошибок (как правило, это прерванные передачи по сети) и отказов (попыток запросить у тест-сервера несуществующую информацию). Используя сортировку списка по типу события, по IP-адресу инициатора и по моменту его свершения, можно уточнить суть ошибок, определить активность конкретного клиента или восстановить последовательность его действий. Закончив анализ, список можно очистить, чтобы в следующий раз рассматривать только новые события. При этом вся информация сохраняется на диске в журнале, который на каждые сутки выделяет отдельный файл. Данные журнала используются, например, для определения среднего времени обслуживания запроса.

Помимо счётчиков строка состояния показывает время работы тест-сервера с момента последнего перезапуска, а заголовок программы ТСервер в скобках выводит число одновременно обслуживаемых в данный момент запросов.

В качестве примера реализации алгоритмов компьютерного тестирования по тестам, созданным в формате HTML, можно назвать программу HTMLTest, разработанную авторами этой статьи в среде визуального программирования Delphi 3.0 для работы под управлением операционной системы Windows 95 (русская версия). Она предназначена для проведения сеансов тестирования на автономных компьютерах или в локальной сети без выхода в Интернет. В локальной сети исполняемый модуль (размером примерно 516 Кбайт), страницы в формате HTML, графические картинки и звуковые фрагменты можно разместить в единственном экземпляре на общем диске и сделать доступными всем рабочим станциям для чтения.

Все файлы теста хранятся упакованными. Это позволяет уменьшить их размер и ограничить доступ к их содержимому, созданному в открытых и широко известных форматах. Распаковка файлов осуществляется при знании пароля в оперативной памяти программы HTMLTest непосредственно в процессе тестирования.

Тестирование начинается с регистрации участника, в результате которой ему присваивается индивидуальный номер. Дальнейшие действия зависят от специфики теста. Например, при тестировании граждан зарубежных стран по русскому языку предполагается выбор одного из пяти субтестов.

Выполнение теста связано с просмотром страниц, содержащих текст, графику, анимацию и звуковые фрагменты, переходы между которыми осуществляются с помощью гипертекстовых ссылок. При этом на одной странице могут располагаться сразу несколько тестовых заданий. Для выбора правильных ответов используются стандартные для формата HTML кнопки, списки, текстовые поля, имеющие в пределах страницы уникальные имена. Выбор связывается с именем элемента и заносится в файл ответов, назначенный участнику тестирования при регистрации.

При завершении работы с тестом ответы тестируемого сравниваются с ключами, которые лежат в одном из упакованных файлов, и результаты по каждому из субтестов появляются в виде процента правильных ответов.

Статистический анализ результатов тестирования

Для оперативного статистического анализа результатов тестирования используется программа СтатИнфо, созданная авторами этой статьи в среде визуального программирования Delphi 3.0 для работы под управлением операционной системы Windows 95 (русская версия). Так как программе необходим Borland Database Engine— набор библиотек для доступа к базам данных, то установленный комплект занимает на диске 2,5 Мбайта. Для хранения результатов 1000 тестируемых дополнительно требуются 0,5 Мбайта. Допускается локальная и сетевая установки. В последнем случае единственный комплект программных модулей и баз данных совместно используется всеми рабочими станциями локальной сети.

Информационная основа программы СтатИнфо — таблица StatInfo.db, которая накапливается на центральном тест-сервере и в пункте тестирования. Перечень полей в таблице используется для формирования запросов.

При формировании запросов каждому полю таблицы можно назначить фильтр, указав диапазон допустимых значений. Например, установив для поля «Предмет» фильтр «-03 09-11 15 51-», из таблицы можно отобрать все записи, содержащие такой код предмета П, что $(П \leq 03)$ или $(09 \leq П \leq 11)$ или $(П = 15)$ или $(П \geq 51)$. Используя дополнительный фильтр «15-» для поля «Балл», можно отобрать всех участников тестирования по указанным предметам, получивших не менее 15 баллов.

В выпадающем списке «Ключ» выбирается поле таблицы, по значениям которого будут группироваться записи о тестируемых. Выпадающие списки «Значения» и «Предметы» задают режимы подведения итогов по ключевому полю и по предметам: «один итог» (показывать без распределения), «с итогом» (показывать как итог, так и распределение), «без итога» (показывать только распределение, без итога).

По кнопке «Принять» программа СтатИнфо принимает сформированный запрос и выдаёт на экран сводку с результатами.

Для сводки можно выбрать интересующую статистическую форму: количество тестируемых, поставленные оценки, полученные баллы или правильные ответы на задания теста. Любую форму можно сначала посмотреть на экране в цифрах или в графике, а потом напечатать на принтере.

При необходимости её можно сохранить в отдельном текстовом файле с разделителем (точка с запятой), который затем загрузить в виде стандартной электронной таблицы в Microsoft Excel и использовать весь богатый набор возможностей этого пакета для построения нужных диаграмм.

Программа СтатИнфо обеспечивает работу со списком всех тестируемых, отобранных в результате выполнения запроса, позволяя:

- выбрать необходимые для просмотра на экране и печати на принтере реквизиты и определить порядок их следования друг за другом;
 - отсортировать список тестируемых по любой совокупности реквизитов;
 - найти участника тестирования, используя сортировку и быстрый поиск по индексам.
- Перед печатью форм и списков на принтере можно указать комментарий к будущей

распечатке и получить представление о ней в окне предварительного просмотра. При этом с помощью кнопок панели инструментов можно:

- изменить ориентацию и поля страниц;
- установить наиболее подходящий масштаб;
- выбрать для просмотра любую страницу будущего отчёта;
- вывести на печать все страницы, только текущую или диапазон номеров. Вывод осуществляется на принтер, используемый системой Windows 95 по умолчанию. При необходимости содержимое окна предварительного просмотра можно преобразовать в графический файл и использовать для печати в других документах или, например, для публикации на Web-сервере в Интернете.

Программа СтатИнфо применяется в пунктах тестирования для загрузки в таблицу StatInfo.db результатов отложенной обработки ответов тестируемых на удалённом тест-сервере, на котором программа ТСервер ведёт центральную таблицу StatInfo.db по всем участникам компьютерного тестирования, позволяя администраторам проводить оперативный статистический анализ по мере поступления новых данных.

По итогам тестирования, как правило, формируется отчёт и проводится оценка качества тестов и тестовых заданий на основе классического и дистракторного анализа. При этом определяются минимальное, среднее, максимальное значение, стандартное отклонение и коэффициент вариации полученных тестовых баллов; номера тестов, в которых использовалось задание; количество тестируемых, выполнявших тест; статистические параметры задания — уровень трудности, точечно-бисериальная корреляция, дифференцирующая способность; работа дистракторов — частота выбора ответов всеми, слабейшими и сильнейшими, отклонения частот от среднего значения, наличие отклонений от нормы; независимость заданий в тесте.

Пример расчёта статистических параметров тестовых заданий

- 1 — Количество давших верный ответ
 2 — Уровень трудности
 3 — Точечно-бисериальная корреляция
 4 — Дифференцирующая способность

03 Физика	2086				№1				2026				№2				1988				№3			
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1	1726	82,7	39,8	37,2	1003	49,5	44,6	50,8	611	30,7	51,6	51,1												
2	1663	79,7	40,6	41,3	1257	62	47,2	54,3	1233	62	45,9	50,7												
3	1037	49,7	39,9	42,9	1826	90,1	31,6	22,3	1020	51,3	52,2	60,6												
4	549	26,3	47,3	40,1	542	26,8	8,6	2,4	722	36,3	57,2	60,2												
5	1114	53,4	48,9	56,2	1109	54,7	56,3	66,7	858	43,2	58,1	65,2												
6	665	31,9	40,6	39	766	37,8	48,7	55,4	945	47,5	38,6	45,6												
7	1377	66	44	48,3	1262	62,3	47,1	51,9	850	42,8	48,8	52,8												
8	1375	65,9	45,2	50,8	1243	61,4	44,6	51,4	1125	56,6	59,8	71,7												
9	757	36,3	31,5	29,9	766	37,8	55,6	60,8	546	27,5	53,7	50,6												
10	1435	68,8	38,7	42,9	1619	79,9	40,8	38,3	1344	67,6	48,7	56												
11	1565	75	37,7	38,6	1200	59,2	37,7	41,7	1545	77,7	38	37,4												
12	1507	72,2	40	42,3	796	39,3	51,1	55,8	1036	52,1	55,1	65,6												
13	1462	70,1	36	38,2	981	48,4	55,9	66,6	1383	69,6	32,1	35												
14	1177	56,4	52	61,2	1445	71,3	52,2	58,3	1153	58	47,5	55,9												
15	221	10,6	45,8	23,9	1231	60,8	33	34,5	615	30,9	13,2	6,2												
16	848	40,7	32,4	31,9	1649	81,4	42,5	39,9	924	46,5	52,6	59,7												
17	1213	58,1	48,5	55,2	986	48,7	37,2	39,2	1259	63,3	37,7	41,9												
18	739	35,4	41,5	39,5	794	39,2	46,8	49,4	742	37,3	34,7	34,5												
19	628	30,1	33	30,4	936	46,2	5,5	-1,9	326	16,4	15,8	6,7												
20	1047	50,2	41,7	45,3	1171	57,8	41,8	46,1	351	17,7	48,1	35,4												
21	2086	100	0	0	1650	81,4	40,1	36,6	941	47,3	48,1	56,1												
22	1283	61,5	55,5	65,8	1554	76,7	44,3	45,1	489	24,6	10,5	4,9												
23	1059	50,8	46	52,3	738	36,4	37,8	34,8	382	19,2	36,1	26,3												
24	678	32,5	38,8	38,3	1092	53,9	37,3	38,8	897	45,1	55,3	62,7												
25	953	45,7	48,1	53,2	548	27	9,4	9,1	448	22,5	29,4	25,1												

26	836	40,1	46,4	49,1	798	39,4	55,2	61,5	606	30,5	53,7	54,8
27	562	26,9	51,5	43,4	759	37,5	51,1	54,5	737	37,1	52,7	54,7
28	1221	58,5	48,3	55,4	1258	62,1	42,6	48,9	1649	82,9	34,1	30,3
29	1206	57,8	53,9	63,4	924	45,6	50,8	59	1236	62,2	51	59
30	929	44,5	40,4	44,8	716	35,3	44,2	45,3	933	46,9	34,5	37,2
31	998	47,8	56,3	64,8	1015	50,1	48,2	55,3	1225	61,6	44,8	50,9
32	625	30	50,5	47,5	1039	51,3	53,8	62,2	749	37,7	34,2	34,1
33	846	40,6	45,6	46,5	611	30,2	53,3	51,3	1183	59,5	39,5	43,2
34	1276	61,2	46,4	51,8	949	46,8	50	56,6	1293	65	43,7	49,9
35	1111	53,3	44,6	49,9	988	48,8	39,2	43,5	1245	62,6	33,6	35,9
36	826	39,6	53,1	58,6	937	46,2	55,6	63,9	1263	63,5	49,2	58,1
37	431	20,7	52,1	42,6	717	35,4	51,7	53,6	882	44,4	21,9	19
38	800	38,4	58,4	63,4	1119	55,2	55,8	63,9	916	46,1	50,8	57
39	864	41,4	63,4	71,7	883	43,6	50	55,4	867	43,6	43,7	48,1
40	806	38,6	65	73,9	405	20	57,5	46,5	654	32,9	57,7	59

Среднее значение

1037,5	49,7	44,7	46,8	1032	50,9	43,9	46,7	929,6	46,8	42,8	45,2
--------	------	------	------	------	------	------	------	-------	------	------	------

Средний балл теста

47,9				49,5				45,3			
------	--	--	--	------	--	--	--	------	--	--	--

Стандарт теста

21,1				20,8				20,2			
------	--	--	--	------	--	--	--	------	--	--	--

Коэф. вариации теста

0,4				0,4				0,4			
-----	--	--	--	-----	--	--	--	-----	--	--	--

Пример анализа работы дистракторов в заданиях по физике

1	2	3	4	5	6	7	8	9	10	11	12
001	001	009	3888	74,2	0	0,5	0,9	0,1	- 0,8		
		020			1	3,4	5,6	1,3	- 4,2	- 3,0	b
					2	12,1	26,7	2,6	- 24,2	+ 5,6	b
					3	2,6	4,3	1,2	- 3,1	- 3,9	b
					4	7,2	8,4	4,8	- 3,6	+ 0,8	
					5*	74,2	54,0	89,9	+ 35,9		ac
001	002	010	1852	71,7	0	0,5	1,1	0,0	- 1,1		
					1	5,8	11,9	1,3	- 10,6	- 1,3	
					2*	71,7	48,4	90,9	+ 42,6		ac
					3	7,1	9,9	5,2	- 4,8	- 0,0	
					4	5,2	11,9	0,6	- 11,2	- 1,9	
					5	9,7	16,8	1,9	- 14,9	+ 2,6	
001	003	011	1705	65,6	0	0,4	0,7	0,5	- 0,2		
					1*	65,6	39,9	88,1	+ 48,2		ac
					2	8,0	17,4	2,1	- 15,3	- 0,6	
					3	14,5	24,2	4,4	- 19,7	+ 5,9	b
					4	9,3	13,8	4,4	- 9,4	+ 0,7	
					5	2,2	4,0	0,5	- 3,5	- 6,4	b

1. Номер задания в тесте.
2. Номер задания в банке.
3. Номера тестов, в которых использовалось задание.
4. Количество тестируемых, решавших задание.
5. Частота выбора правильного ответа на задание (в процентах).
6. Номера ответов (0 — задание пропущено, * — правильный ответ).
7. Частота выбора ответа (в процентах).
8. Частота выбора ответа слабейшими (25% тестируемых по каждому из тестов столбца 3, получившие наименьшие тестовые баллы).
9. Частота выбора ответа сильнейшими (25% тестируемых по каждому из тестов столбца 3, получившие наибольшие тестовые баллы).

10. Дифференцирующая способность ответа (столбец 9 — столбец 8).
11. Отклонение частоты выбора неправильного ответа от среднего значения.
12. Сигналы о наличии отклонений от нормы:
 - а) частота выбора правильного ответа отличается от V больше, чем на $0,5 * V$ (где $V = 100/N$, aN — количество ответов на задание);
 - б) частота выбора неправильного ответа отличается от V , больше чем на $0,4 * V$ (где V — среднее значения частот выбора неправильных ответов);
 - с) процент правильных ответов (столбец 5) меньше 40 или больше 60.

Хранение всей необходимой информации в единой реляционной базе данных позволяет значительно ускорить процесс расчёта базовых величин и даёт возможность автору или тестологу быстро перейти от полученной в отчёте цифры к стоящему за ней тестовому заданию. Для определения комплексных характеристик, таких, как надёжность, валидность, эффективность тестов, как правило, используются специальные статистические пакеты.

Методика синтеза ИВС КТ

Эта методика определяет последовательность действий, выполнение которых позволяет синтезировать ИВС КТ, а именно:

1. Формулируются цели проведения компьютерного тестирования (например, мониторинг, аттестация, самоконтроль).
2. По материалам авторов создаются тестовые задания, определяется их тип и средний размер при передаче по каналам связи.
3. Выбираются вариант предъявления тестов (на автономном компьютере, в локальной сети или через Интернет) и схема обработки ответов тестируемых на тест-сервере (оперативная или отложенная).
4. Производится оценка возможного числа участников и рассчитывается интенсивность сеансов компьютерного тестирования.
5. На основе интенсивности сеансов тестирования и среднего размера тестовых заданий, исходя из ограничений на значение показателя эффективности, с помощью разработанной имитационной модели определяются требования к параметрам телекоммуникационной подсистемы.
6. С учётом интенсивности сеансов тестирования и соотношения между объёмами оперативной и отложенной обработки ответов тестируемых с помощью разработанной аналитической модели выбирается комплекс технических средств для тест-сервера таким образом, чтобы среднее время пребывания одной заявки в блоке обработки не превышало допустимого значения.
7. Оценивается стоимость технических средств и эффективность их эксплуатации (загрузка процессора, процент использования пропускной способности канала сервера и очереди маршрутизатора).
8. На тест-сервере, управляющих и рабочих станциях устанавливается необходимое программное обеспечение и проводятся комплексные испытания синтезированной ИВС КТ.
9. На тест-сервере запускается программа ТСервер. На управляющих станциях организуется банк тестовых заданий и с помощью программ Компоновщик и Камера формируются компьютерные тесты.
10. Для тестирования через Интернет в режиме оперативной обработки на рабочих станциях запускается программа Оператор и проводятся несколько сеансов, включая регистрацию и выбор теста, загрузку тестовых заданий и их выполнение, проверку ответов и подсчёт набранных баллов. При этом регистрируется среднее время отклика на запрос клиента со стороны ИВС КТ.
11. Для тестирования в локальной сети в режиме отложенной обработки на общий диск устанавливается программа Тестер, загружаются пакеты тестов и проводятся сеансы тестирования. В режиме инструктора осуществляется контроль за работой тестируемых и формируются протоколы, которые затем отправляются на удалённый тест-сервер. При этом реги-

стрируется среднее время получения результатов обработки в пересчёте на одну запись протокола. Полученные с тест-сервера данные загружаются в таблицу StatInfo.db и анализируются с помощью программы СтатИнфо.

12. На тест-сервере в программе ТСервер администратор отслеживает все события, регистрируя среднее время обработки одного запроса со стороны клиента. С помощью программы СтатИнфо, установленной на любой управляющей станции, анализируются результаты тестирования, накапливаемые в центральной таблице StatInfo.db.

13. По данным проведённых комплексных испытаний делается вывод о работоспособности синтезированной ИВС КТ.

Внедрение результатов исследования

Разработанные авторами этой статьи математические модели, центральные алгоритмы и программное обеспечение ИВС КТ используются в Центре тестирования выпускников общеобразовательных учреждений Российской Федерации и в Головном центре тестирования граждан зарубежных стран по русскому языку.

В Центре тестирования выпускников общеобразовательных учреждений Российской Федерации с 1998 г. применяется подсистема автоматизированной компоновки компьютерных тестов, с помощью которой из банка тестовых заданий централизованного тестирования на бланках формируются тесты для репетиционного тестирования на компьютерах.

Репетиционное тестирование для учащихся учебных заведений и будущих абитуриентов проводилось с 1 октября 1998 г. по 31 марта 1999 г. Использовались программы Компоновщик, Камера, Тестер, Оператор, ТСервер и СтатИнфо. Итоги этого тестирования таковы:

- по каждому из 28 предметов 9-го, 10-го и 11-го классов сформированы 10 компьютерных тестов из банка тестовых заданий образца 1998 года;
- в 44 регионах страны прошли 8164 сеанса бланкового и компьютерного тестирования в классах с последующей обработкой протоколов через Интернет и выдачей участникам тестирования справок с результатами;
- с Web-сервера Центра тестирования разным адресатам выдано 1184 копии программы Оператор, с помощью которой проведено 8255 сеансов индивидуального тестирования за компьютером через Интернет;
- программой ТСервер, запущенной на тест-сервере, обработано 245 373 заявки на получение тестового задания или на подсчёт набранных баллов, причём весь комплекс репетиционного тестирования через Интернет работал круглосуточно в автоматическом режиме.

С 1 апреля по 25 мая 1999г. с помощью программ ТСервер и СтатИнфо осуществлялась обработка через Интернет протоколов централизованного тестирования выпускников на бланках. 15 региональных представительств (см. табл. 9) успешно освоили предложенную технологию взаимодействия с Центром тестирования, предполагающую подготовку протоколов с исходными материалами тестирования, передачу их на обработку через Интернет, получение на тест-сервере результатов, загрузку их в региональную таблицу StatInfo.db, получение списков тестируемых и оперативной статистики в программе СтатИнфо. Программой ТСервер, запущенной на тест-сервере для круглосуточной работы в автоматическом режиме, по мере поступления данных обработаны и занесены в центральную таблицу StatInfo.db сведения о 95 743 сеансах бланкового тестирования (это около 25% общего объёма централизованного тестирования в 1999 году).

Таблица 9. Региональные представительства, работавшие в 1999 году через Интернет

Код	Название организации
0201	Уфимский Государственный Авиационный Технический Университет
0301	Восточно-Сибирский Государственный Технический Университет
0801	Калмыцкий Государственный Университет
1202	Марийский Государственный Университет
1501	ИПК РО Владикавказа
2204	Алтайский Государственный Университет

2401 Красноярский Государственный Технический Университет
2402 Красноярский Государственный Университет
5301 Новгородский Государственный Университет
5601 Оренбургский Государственный Университет
5602 ИПК РО Оренбурга
6102 Донской Государственный Технический Университет
6104 Донская Государственная Академия Сервиса
6602 Уральский Государственный Университет
8601 Комитет образования Нефтеюганска

Перед завершением централизованного тестирования выпускников общеобразовательных учреждений РФ вся информация, накопленная в ходе сканирования бланков ответов и подсчёта набранных баллов, поступает в центральную таблицу StatInfo.db, что позволяет анализировать результаты по России в целом и по каждому региону в отдельности, получая с помощью программы СтатИнфо нужные списки тестируемых и статистические формы: количество участников, поставленные оценки, полученные баллы, правильные ответы на задания теста. Сейчас программа СтатИнфо с данными 1999 года (таблица StatInfo.db содержит 380 тыс. записей) эксплуатируется в Центре тестирования и в 20 его региональных представительствах на базе ведущих вузов России.

Ежегодно по итогам централизованного тестирования выпускников общеобразовательных учреждений РФ формируется отчёт, основой которого являются такие статистические формы:

- с информацией об участниках тестирования на уровне страны, региона, представительства и учебного заведения (23 типа);
- с данными (результаты классического и дистракторного анализа тестов и отдельных заданий) о качестве тестовых материалов (5 типов).

Все расчёты выполняются с помощью информационного и программного обеспечения, разработанного в рамках данного исследования.

В Головном центре тестирования граждан зарубежных стран по русскому языку используется программа ТРТестер, которая позволяет на автономных или объединённых локальной сетью компьютерах проводить тестирование по русскому языку как иностранному. Применяются мультимедийные тесты в формате HTML, содержащие пять разделов: лексика и грамматика, чтение, письмо, аудирование, ведение беседы. Программа прошла апробацию на филологическом факультете МГУ.

Для учёта граждан зарубежных стран, прошедших тестирование по русскому языку, в Головном центре и его представительствах используется разработанная в рамках данного исследования программа ТРПротокол, которая позволяет регистрировать реквизиты участников тестирования, их результаты и выданные им документы.

Результаты проведённого исследования способствуют повышению теоретической, методической и программной оснащённости выполняемых работ.