

**ГОСУДАРСТВЕННЫЙ КОМИТЕТ СВЯЗИ, ИНФОРМАТИЗАЦИИ И  
ТЕЛЕКОММУНИКАЦИОННЫХ ТЕХНОЛОГИЙ РЕСПУБЛИКИ  
УЗБЕКИСТАНА  
ТАШКЕНТСКИЙ УНИВЕРСИТЕТ ИНФОРМАЦИОННЫХ  
ТЕХНОЛОГИИ**

К защите  
Зав. Кафедрой «ПОИТ»  
Проф. Нишонов А.Х.

«\_\_» \_\_\_\_\_ 2013 г.

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА**

На тему: Специальные обучающие курсы по повышению компьютерной грамотности

Выпускник \_\_\_\_\_ Юнусов Ф. Ф.

Руководитель \_\_\_\_\_ Нишанов А.Х.

Рецензент \_\_\_\_\_ Примова Х.

Консультант по БЖД \_\_\_\_\_ \_\_\_\_\_

Ташкент 2013

ГОСУДАРСТВЕННЫЙ КОМИТЕТ СВЯЗИ, ИНФОРМАТИЗАЦИИ И  
ТЕЛЕКОММУНИКАЦИОННЫХ ТЕХНОЛОГИЙ РЕСПУБЛИКИ  
УЗБЕКИСТАНА  
ТАШКЕНТСКИЙ УНИВЕРСИТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

Факультет ИТ кафедра ПОИТ  
Направление (специальность) 5521900 – Информатика и информационные технологии

**УТВЕРЖДАЮ**

Зав кафедрой \_\_\_\_\_  
« \_\_\_\_\_ » \_\_\_\_\_ 2013 г.

**ЗАДАНИЕ**

на выпускную квалификационную работу  
Юнусов Фаррух Фарходович

1. Тема работы: Специальные обучающие курсы по повышению компьютерной грамотности
2. Утверждена приказом по университету от №145-07 11.02.2013
3. Срок сдачи законченной работы 1.06.2013
4. Исходные данные к работе: автоматизированные обучающие системы, компьютерные учебные программы и С# учебный курс
5. Содержание расчётно – пояснительной записи (перечень подлежащих разработке вопросов): Обзор обучающих программ в компьютерных технологиях, Разработка специальные обучающие курсы по повышению компьютерной грамотности, Эксплуатационная и экономическая часть, Безопасность жизнедеятельности
6. Перечень графического материала Презентация
7. Дата выдачи задания 1.03.2013

Руководитель \_\_\_\_\_  
(подпись)

Задание принял \_\_\_\_\_  
(подпись)

## 8. Консультанты по отдельным разделам выпускной работы

| Раздел         | Ф.И.О руководителя | Подпись дата  |                 |
|----------------|--------------------|---------------|-----------------|
|                |                    | Задание выдал | Задание получил |
| Введение       | Нишанов А.Х.       | 8.04.13       |                 |
| Основная часть | Нишанов А.Х.       | 17.04.13      |                 |
| БЖД            |                    | 1.05.13       |                 |
| Заключение     | Нишанов А.Х.       | 7.05.13       |                 |

## 9. График выполнения работы

| №  | Наименование раздела работы  | Срок выполнения | Отметка руководителя о выполнении |
|----|--|-----------------|-----------------------------------|
| 1. | Введение   | 8.04.13         |                                   |
| 2. | Обзор обучающих программ в компьютерных технологиях                          | 17.04.13        |                                   |
| 3. | Разработка специальные обучающие курсы по повышению компьютерной грамотности | 23.04.13        |                                   |
| 4. | Эксплуатационная и экономическая часть                                       | 27.04.13        |                                   |
| 5. | Безопасность жизнедеятельности   | 1.05.13         |                                   |
| 6. | Заключение   | 7.05.13         |                                   |

Выпускник \_\_\_\_\_  
(подпись)

«1» июн 2013 г.

Руководитель \_\_\_\_\_  
(подпись)

«1» июн 2013 г.

## **MAZMUNNOMA**

Битирув малакавий ишининг мақсади “Компьютер саводхонлиги” фани бўйича автоматлаштирилган таълим берувчи тизим яратишдан иборат. Ишда автоматлаштирилган таълим берувчи тизим ишлаб чиқилган, ҳамда факулуддан ташқари вазиятлар учун экспериментлар тестдан ўтказилган. Дастурнинг асосий кодлари C# дастурлаш тилида ишлаб чиқилган.

## **АННОТАЦИЯ**

Целью выпускной квалификационной работы является создание автоматизированную обучающую систему по дисциплине «Компьютерной грамотности» для студентов ВУЗов. В работе разработано автоматизированная обучающая система, а также протестировано, чтобы провести эксперименты исключительных ситуаций. Основные коды программы разработано на языке программирование C#.

## **SUMMARY**

## СОДЕРЖАНИЕ

|   |    |
|---|----|
| ВВЕДЕНИЕ .....  | 6  |
| 1. ОБЗОР ОБУЧАЮЩИХ ПРОГРАММ В КОМПЬЮТЕРНЫХ<br>ТЕХНОЛОГИЯХ .....                         | 9  |
| 1.1. Анализ литературных источников и сайтов Интернета.....                             | 9  |
| 1.2. Синтез результатов .....   | 19 |
| 1.3. Перспективы возможного продолжения работы .....                                    | 22 |
| 2. РАЗРАБОТКА СПЕЦИАЛЬНЫЕ ОБУЧАЮЩИЕ КУРСЫ ПО<br>ПОВЫШЕНИЮ КОМПЬЮТЕРНОЙ ГРАМОТНОСТИ..... | 23 |
| 2.1. Техническое задание.....   | 23 |
| 2.2. Конструкторская часть .....  | 25 |
| 2.3. Проектирование системы.....  | 29 |
| 3. ЭКСПЛУАТАЦИОННАЯ И ЭКОНОМИЧЕСКАЯ ЧАСТЬ.....  | 41 |
| 3.1. Эксплуатационная часть .....   | 41 |
| 3.2. Экспериментальная часть .....  | 45 |
| 3.3. Экономическая часть .....  | 47 |
| 4. БЕЗОПАСНОСТЬ ЖИЗНЕДЕЯТЕЛЬНОСТИ .....   | 52 |
| 4.1. Эргономика .....   | 52 |
| 4.2. Обеспечение пожарной безопасности .....  | 63 |
| ЗАКЛЮЧЕНИЕ .....  | 65 |
| Список литературы .....   | 66 |
| Приложение .....  | 68 |

## ВВЕДЕНИЕ

Постоянное увеличение объема информации и ограниченность учебного времени обуславливают необходимость интенсификации обучения, разработки и внедрения нетрадиционных технологий, базирующихся на использовании вычислительной техники с применением активных методов обучения во всем их разнообразии и комплексности. Реализация активных методов обучения – одна из основных задач дидактики, которая предполагает активизацию всего процесса, выявление системы, способов, приемов, способствующих повышению активности обучаемых через формирование положительной мотивационной структуры учебно-познавательной деятельности [12].

Лекционно-семинарная форма обучения давно потеряла свою эффективность - практика доказала, что почти 50% учебного времени тратится впустую. Изучая зарубежный опыт, можно выделить следующий важный аспект: преподаватель выступает не в роли распространителя информации (как это традиционно принято), а в роли консультанта, советчика, иногда даже коллеги обучаемого [9].

Это дает некоторые положительные моменты: студенты активно участвуют в процессе обучения, приучаются мыслить самостоятельно, выдвигать свои точки зрения, моделировать реальные ситуации.

Развитие информационных технологий предоставило новую, уникальную возможность проведения занятий с внедрением автоматизированных обучающих систем по дисциплинам в вузах. Она, во-первых, позволяет самому обучаемому выбрать и время и место для обучения, во-вторых, дает возможность использовать в обучении новые информационные технологии, в-четвертых, в определенной степени сокращает расходы на обучение. С другой стороны, внедрение в образование новых автоматизированных обучающих систем усиливает возможности индивидуализации обучения [11].

Достоинствами автоматизированных обучающих систем (АОС), являются: во-первых, их мобильность, во-вторых, доступность связи с развитием компьютерных сетей, в-третьих, адекватность уровню развития современных научных знаний. С другой стороны, создание АОС способствует также решению и такой проблемы, как постоянное обновление информационного материала. В них также может содержаться большое количество упражнений и примеров, подробно иллюстрироваться в динамике различные виды информации. Кроме того, при помощи АОС осуществляется контроль знаний - компьютерное тестирование [15].

В настоящее время традиционные подходы в области преподавания информатики и программирования в вузе не способны отследить быстроменяющуюся действительность в области информационных технологий, связанную с бурным развитием вычислительной техники, операционных систем, парадигм программирования, организацией, анализом, представлением информации и обеспечением доступа к ней, в том числе и в сетях.

Выход из создавшегося положения видится в несколько иной расстановке акцентов, как на принципы обучения, так и на сам процесс и условия обучения, позволяющие не только и не столько учить в прямом смысле этого слова, сколько помогать учиться, организовать процесс обучения так, чтобы развивались не только практические навыки в области информатики и программирования, но и соответствующее мировоззрение и творческий потенциал, позволяющие будущему специалисту с минимальными затратами осуществлять доступ к требуемым информационным ресурсам (в том числе и мировым), самостоятельно адаптироваться к действительности, определяемой появлением новых парадигм, сред и инструментальных средств [13].

Цель: создать автоматизированную обучающую систему по дисциплине «Компьютерной грамотности» для студентов ВУЗов.

Задачи:

1. Изучить и проанализировать предметную область.
  2. Изучить и сравнить программы-аналоги.
  3. Обобщить и систематизировать материал для автоматизированной обучающей системы.
  4. Написать техническое задание к проекту.
  5. Разработать автоматизированную обучающую систему, протестировать, провести эксперименты исключительных ситуаций.
  6. Провести экономический анализ эффективности проектного решения.
- Источниками информации при написании выпускной квалификационной работы служили учебные пособия для вузов, нормативные документы, электронные ресурсы сети Интернет.

Методы исследования:

1. Метод математического моделирования.
2. Монографический метод.
3. Метод экономического анализа.
4. Экспериментальный метод.
5. Аналитический метод.
6. Синтетический метод.

Выпускная квалификационная работа насчитывает 46 страниц основного текста, 30 рисунков, -4 таблицы и 7 формул.

# **1. ОБЗОР ОБУЧАЮЩИХ ПРОГРАММ В КОМПЬЮТЕРНЫХ ТЕХНОЛОГИЯХ**

## **1.1. Анализ литературных источников и сайтов Интернета**

### **Классификация обучающих программ**

Существует множество различных подходов к классификации обучающих компьютерных программ, но единого мнения и соответственно общей классификации нет. В [1] предлагается **2 классификации:**

1. основывается на целях и задачах обучающих программ или режимах использования автоматизированных обучающих систем:

- а) иллюстрирующие;
- б) консультирующие;
- в) операционная среда;
- г) тренажеры;
- д) обучающий контроль.

2. основывается на анализе и обобщении различных классификаций:

- а) тренировочные;
- б) наставнические;
- в) проблемного обучения;
- г) имитационные и моделирующие;
- д) игровые.

**Формами обучающих программ являются:**

- Электронный учебник (ЭУ)
- Автоматизированная обучающая система (АОС)
- Тестирующая программа

### **Электронный учебник**

*Электронный учебник (ЭУ) - компьютерное педагогическое программное средство, предназначенное для предъявления новой информации, дополняющей печатные издания, служащее для*

*индивидуального и индивидуализированного обучения и позволяющее в ограниченной мере тестировать полученные знания и умения обучаемого.*

ЭУ может быть открытой или частично открытой системой, т.е. позволяет внести изменения в содержание и структуру, что требуется для адаптации ЭУ к конкретному ВУЗу. Модификация ЭУ должна быть разрешена только опытному преподавателю, чтобы не нарушалась общая структура и содержание. Для ограничения от несанкционированного изменения должен применяться пароль.

### **Некоторые элементы ЭУ:**

1. Минимум текстовой информации. Существенное значение имеет размер, начертание шрифта, выделение слов или фраз цветом.
2. Большое количество иллюстративного материала.
3. Видеофрагменты, позволяющие передать в динамике процессы и явления.
4. Аудиофрагменты.
5. Гиперссылки по элементам ЭУ, ссылки на другие ЭУ и справочники, желательно содержание с быстрым переходом на нужную страницу.
6. Запуск других компьютерных программ для показа примеров, тестирования и др.
7. Возможность делать закладки в любом месте, отображать список закладок, отсортировав их в любом порядке.
8. Список рекомендованной литературы, включающий имеющиеся в библиотеке издания, ссылки на статьи в журналах, сборники научных конференций, электронные публикации, размещенные на серверах ВУЗа или в Internet.

ЭУ, обладающий мультимедийными и интерактивными возможностями, помогает усвоить большой по объёму и сложный материал, предъявляет необходимую информацию по запросу обучаемого, что приближает к обучению под руководством преподавателя. Постепенно

расширяется круг вопросов, которые ставит ЭУ перед обучающимся, и которые можно задать программе.

При разработке ЭУ, также как и других обучающих программ, необходимо соблюдать психологические принципы взаимодействия человек-компьютер. Нарушение может увеличить время на обучение, снизить мотивацию к учению и др. Применение ЭУ целесообразно только в комплексе с другими обучающими системами, дополняя печатные издания.

ЭУ часто ставится в один ряд с АОС, но их нельзя полностью отождествлять.

### **Диалог компьютера и студента. автоматизированная обучающая программа**

*Автоматизированная обучающая система (АОС) - компьютерное педагогическое программное средство, предназначенное для предъявления новой информации, усвоения навыков и умений, промежуточного и итогового тестирования (экзаменования), обладающее развитой системой помощи, как по самой обучающей программе, так и по изучаемому предмету, обладающее возможностью поднастройки к обучаемому (его уровню знаний, скорости и пути продвижения по изучаемому материалу и т.д.), обладающее развитой системой сбора и обработки статистической информации о каждом отдельном обучаемом, группе и потоке обучаемых, накапливающее информацию о часто встречающихся ошибках при работе с обучающей системой и ошибках по изучаемой теме или дисциплине.*

При разработке АОС необходимо решать **комплекс проблем**, включающих учебно-методические, психологические, организационные, технические, экономические аспекты, тесно связанные между собой. Подробно рассмотрены в [\[2\]](#).

#### **АОС может представлять собой:**

- одну управляющую программу, в которой реализованы все возможности АОС;

- комплекс взаимосвязанных программ-модулей, каждую из которых допускается использовать отдельно или запускать из единой интегрирующей оболочки.

#### **АОС должна включать:**

- информационные файлы, содержащие текстовую, графическую и др. информацию, используемую для предъявления в качестве новых порций знаний, файлы примеров, демонстраций, тестовых вопросов и заданий, ответов и т.д.
- сценарий учебного процесса, базу данных студентов с сохранением пройденных тем, результатов тестирования и экзаменования.
- базу данных нестандартных ответов и решений студентов, сохранение информации о сбоях системы и неадекватных ответах АОС на вопросы, анализ которых поможет совершенствовать АОС.

#### **Отличия АОС от других видов компьютерных обучающих программ:**

- развитый, интерактивный диалог между компьютерной системой и обучаемым;
- дидактические и психологические аспекты взаимодействия "человек - компьютер";
- органичное сочетание технических, учебно-методических, программных и организационных обеспечений на базе компьютерных технологий и средств, предназначенных для индивидуализации обучения.

#### **Этапы процесса передачи информации обучаемому:**

1. привлечь внимание обучаемого к передаче новой порции информации.
2. заинтересовать обучаемого в необходимости воспринимать ее.
3. поддерживать заинтересованность, чтобы у обучаемого появилось устойчивое желание получать новые порции знания по предлагаемому и смежным вопросам и дисциплинам, сформировать у

обучаемого внутреннюю потребность в совершенствовании своих навыков и умений.

Существующие обучающие программы, выпускаемые многими известными компьютерными фирмами, являются ЭУ. Их нельзя использовать в реальном учебном процессе, но для самостоятельного обучения в комплексе с традиционными методиками, их применять можно и нужно.

### **Стратегия информатизации образования в государстве**

По [3] низкие темпы информатизации образования в республике обусловлены, в первую очередь, отсутствием мультимедийных учебников, соответствующих принятым образовательным стандартам, вписывающихся в учебную программу и пригодных для использования в образовательных учреждениях.

Анализ объемов продаж показывает, что обычные семьи готовы приобретать качественные мультимедийные обучающие программы.

Отсутствие мультимедийных учебников есть следствие отсутствия программно-технологических комплексов для массового создания компьютерных обучающих программ. Многие программы создаются гипертекстовыми HTML системами и, в итоге, дискредитируют идею повышения эффективности обучения с помощью ПК, т.к. в них вся основная теоретическая часть изложена в текстовом виде, только частично использованы мультимедийные возможности.

Необходимо:

- классифицировать ЭУ по системным характеристикам;
- разработать технические условия, обеспечивающие высокий уровень обучающих программ и их функциональных возможностей;
- разработать систему гибких стандартов по интерфейсу, по протоколам обмена данными, чтобы в любом образовательном учреждении могли быть использованы ЭУ, созданные в различных авторских коллективах.

Концепция создания ЭУ на основе универсальных пакетов типа MacromediaDirector или HyperMethod является тупиковой, несмотря на возможность создания преподавателем собственного ЭУ. Возникает большое количество сложных технологических проблем. Обеспечить качество и технологичность создания ЭУ в состоянии только коллектив профессионалов.

В перспективе, через Internet-канал необходимо организовывать управление учебным процессом - текущий контроль, консультирование, тестирование и т.д.

### **Обучающие программы**

#### **Издательство "ММТ и ДО". Электронные учебники серии teachpro**

ООО "Мультимедиа Технологии и Дистанционное Обучение" (ММТ и ДО) специализируется в создании мультимедийных обучающих программ. Создано более 80 наименований CD ROM-учебников.

Одно из наиболее перспективных направлений для "ММТ и ДО" - создание ЭУ серии TeachPro. **TeachPro** - универсальный программно-технологический комплекс, оболочка и набор инструментальных, программных средств, обеспечивающих возможности создания в кратчайшие сроки мультимедийных ЭУ на CD-ROM практически по любым дисциплинам, по школьной и вузовской программе, пригодных для занятий в классах, самоподготовки, повышения квалификации. Это учебники и "решебники" по иностранным языкам, компьютерной грамотности, математике, физике, химии и др. Подробно в [3].

В курсах TeachPro реализован наиболее эффективный способ преподавания - наглядная демонстрация и одновременное объяснение. Обучаемый видит на экране фильм с высококачественным изображением, сопровождаемый пояснениями преподавателя. Для демонстрации используются серия статических слайдов, анимационные фильмы, имитация доски с цветными фломастерами, на которой преподаватель рисует схемы, формулы, тексты и т.п.

Другое направление деятельности "ММТ и ДО" - создание сетевых мультимедийных обучающих программ TeachPro, предназначенных для использования в локальных сетях при групповом компьютерном обучении по принципу "клиент-сервер". Сетевые обучающие программы снабжаются системой контроля и координации процесса обучения, которая позволяет оперативно отслеживать успеваемость каждого студента, устанавливать индивидуальные уровни сложности, тестировать, принимать зачеты и экзамены.

Достоинства учебников "ММТ и ДО":

- большой объем озвученного учебного материала на одном CD-ROM - 40-60 часов лекций, что сопоставимо с книгой более 1000 страниц.
- соответствие современным требованиям к обучающим программам - интерактивность, контроль пройденного материала, тестирование и оценка знаний;
- возможность использования курсов, как в учебных заведениях, так и для самостоятельного обучения, а также в системах дистанционного обучения;
- развернутая тестовая система по 4-5 тысяч вопросов для каждого;
- возможность издавать или распечатывать твердые копии учебников.

ЭУ создаются коллективом специалистов по предметной области и коллективом программистов, методистов и технологов "ММТ и ДО". Участники проекта: руководитель проекта, автор-предметник, его ассистент, 2-3 оператора по обработке данных (дизайн, анимация, обработка и архивация данных). Качество ЭУ многократно возрастает за счёт выполнения всех технологических процедур на профессиональном уровне. В зависимости от сложности и насыщенности материала разрабатываемого курса требуется примерно 3-4 месяца.

### **Обучающие программы на рынке CD**

Статья [4] посвящена обзору обучающих компакт-дисков. В ней рассмотрены:

1. Практический курс Internet Explorer 5.0. Издатель "Кирилл и Мефодий".
2. 1С:Репетитор. Русский язык. Издатель 1С.
3. Решебник по математике. Издатель Руссобит-М.
4. Химия общая и неорганическая. Издатель Республиканский мультимедиа центр.
5. Общая биология. Издатель Росучприбор.
6. TeachPro Win&Office 2000. Издатель "ММТ и ДО"
7. Практический курс Outlook Express. Издатель "Кирилл и Мефодий".
8. Adobe Photoshop: Шаг за шагом. Издатель Руссобит-М.
9. Диктант. Репетитор русского языка. Издатель Равновесие Медиа.

К моему сожалению и авторов [4], нового способа обучения или запоминания целого учебника, к примеру, за 10 минут, так и не придумали. Обучающие компакт-диски могут порадовать современного ученика и подучить, в том числе азам компьютерной грамотности, но в основном бесполезны.

### **Обучающие программы по математической логике**

#### **Электронный урок "представление логических функций"**

Электронный урок [5] позволяет изучить одно из базовых понятий алгебры логики - формы представления логических функций.

Основными способами представления функций алгебры логики являются таблица истинности, совершенные дизъюнктивная и конъюнктивная нормальные формы, которые могут быть представлены полной и сокращенной записями. Предполагается выполнение 5 заданий, каждое из которых требует по одной заданной форме логической функции составить 4 остальные.

Главное меню урока содержит: закладки "Теория", "Задание 1", ..., "Задание 5"; кнопку "Старт" для запуска урока; флажок "С оценкой/без оценки" для переключения режимов работы.

Урок может использоваться в двух режимах:

**1. без оценки.** Предназначен для самостоятельной тренировки. Пользователь может выбирать задания в любой последовательности, в любой момент ознакомиться с теоретическим материалом. Выполнение каждого задания требует ввода четырех ответов и подтверждается нажатием соответствующей кнопки. При этом проверяется введенный ответ по каждой введенной записи функции. При неправильном ответе пользователь может выбрать, желает ли он попробовать самостоятельно найти и исправить ошибку либо же хочет посмотреть правильный результат. После этого можно получить новый тест на данное задание или перейти к выполнению тестирования по любому другому заданию.

**2. с оценкой** Предназначен для контрольного тестирования. Теория становится недоступной, задания запускаются по очереди вне зависимости от правильности их выполнения (закладки "Задание 1", ..., "Задание 5" становятся пассивными). Ответ вводится пользователем по общей кнопке на все задание в целом. Ответ проверяется, выводятся правильные результаты и начисляются баллы, если ответы верны. После этого пользователю выдается следующее из 5 заданий. Таким образом, в этом режиме оценивается правильность 20 выполненных пунктов заданий. В конце работы на основании количества набранных баллов определяется оценка за весь урок.

Существенное достоинство урока, облегчающее жизнь обучаемого, - кнопки, с помощью которых вводятся переменные, скобки и знаки логических операций.

Урок реализован в среде программирования Delphi. Особенность его программного обеспечения - отсутствие базы данных заданий и ответов. При каждом запуске урока генерируется новое задание и вычисляются правильные ответы. Такой подход, хотя и значительно усложнил программное обеспечение урока, но сделал бессмысленной попытку взлома базы данных урока с целью получения правильных ответов.

**Комплекс программ по математической логике**

Комплекс [6] состоит из нескольких модулей, освещающих одну из тем курса математической логики. Бесперспективно изучать математику, читая текст с экрана, поэтому авторами были выбраны отдельные темы, наиболее подходящих для привлечения компьютера. Основные критерии:

1. **темы**, при изложении которых преподавателю затруднительно объяснять только с помощью рисования мелом на доске.

2. **темы**, где требуется приобретение некоторых навыков алгоритмического характера и где именно компьютер может осуществить безошибочную проверку, демонстрацию, генерирование большого числа примеров и задач, рутинную проверку результатов.

Хотя во всех программах имеются текстовые вставки с объяснением теоретического материала, они предполагают предварительное знание теории. Основная цель программ - показать в движении изучаемые объекты, дать возможность обучаемому самому управлять этим движением и, в итоге, решив серию контрольных задач, закрепить знание на уровне наглядных представлений и конкретных навыков.

Несмотря на то, что сценарии почти всех программ имеют сходную общую структуру (демонстрационная, тренировочная, контролирующая части), конкретные детали довольно сильно между собой разнятся. Реализованные программы могут использоваться независимо в процессе обучения:

1. Ашаев И.В., Беляев В.В., Беляев В.Я., Мясников А.Г., Печкин Д.А. "Текстовые логические задачи".
2. Падерин Е.В., Лютиков С.А. "Формулы исчисления высказываний".
3. Падерин Е.В. "Формулы логики предикатов".
4. Ашаев И.В., Ющенко А.Ю., Печкин Д.А. "Эквивалентные преобразования". Приведение к нормальным формам с помощью логических тождеств.
5. Бормотов Д.Ю., Чулковский А.Н. "Бескобочная запись формул". В игровой форме демонстрирует процесс преобразования обычной формулы логики высказываний в обратную запись.

6. Бормотов Д.Ю. "Полиномы Жегалкина". Тренажер для выписывания полинома Жегалкина по заданной таблице истинности функции алгебры логики.
7. Чулковский А.Н. "Классы Поста".
8. Оффенбах И.В. "Замкнутые классы булевых функций".
9. Кузнецов А.В., Хрущев Н.С. "Машина Тьюринга". Интерпретатор машины Тьюринга, выполненный в виде интегрированной оболочки для создания и исполнения программ.

Программы реализованы в среде программирования Pascal, C++, Turbo Assembler.

## 1.2. Синтез результатов

### Сопоставление точек зрения авторов и собственное мнение

Изучив несколько статей по теме работы и **сопоставив точки зрения**, я пришла к выводу, что исследуя разные проблемы обучающего процесса, авторы приходят к единому мнению. В настоящее время многие предлагаемые для продажи обучающие программы являются мультимедийными ЭУ. Необходимы современные программы, повышающие эффективность обучения. В ВУЗах разрабатываются и используются собственные компьютерные обучающие программы. Существующие программы можно и нужно использовать в комплексе с другими средствами, не отрицая печатные издания и диалог между учителем и обучающимся, при самостоятельном освоении материала, тестировании студентов.

**Я считаю**, что будущее за автоматизированными обучающими системами. Электронные учебники и тестирующие программы представляются мне как "ограниченные АОС", выполняющие "узкую" функцию (ЭУ - в основном представлять информацию и частично проверять полученные знания, а тестирующие - проверять знания и выдавать справочную информацию). Речь не о достоинствах и недостатках, а о разных выполняемых функциях. АОС включает в себя и тестирующую программу, и электронный учебник - "два в одном".

## **2.2 азначение разрабатываемой обучающей программы**

1. Возможность использования для проведения лекционных занятий.
2. Возможность использования для проведения практических занятий (с повторением теоретической части изучаемого раздела).
3. Подготовка к контрольной работе, промежуточному экзамену, итоговому экзамену, контролю остаточных знаний как по теоретической части курса, так и по решению практических задач. Возможна подготовка как по отдельной теме, так и по всему курсу.
4. Проведение контрольной работы, промежуточного экзамена, итогового экзамена, контроля остаточных знаний как по теоретической части курса, так и по решению практических задач. Возможен контроль, т.е. тестирование, как по отдельной теме, так и по всему курсу.

### **Требования, предъявляемые к разрабатываемой обучающей программе**

1. Обучающая программа должна содержать информацию, изложенную на доступном языке, аналогии, яркие примеры, шуточные замечания.
2. Наличие текстовой, графической информации, озвучивания, гипертекста (возможны анимация, видеофрагменты).
3. Наличие файлов примеров, тестовых вопросов и заданий, ответов.
4. Удобная навигация по теоретической части курса, наличие содержания.
5. Обучаемый шаг за шагом совместно с обучающей программой проходит весь путь решения предложенной ему задачи.
6. При неправильных действиях обучаемого обучающая программа выдаёт подсказки, рекомендации, необходимые в данный момент фрагменты теоретического материала, иллюстрирует ситуацию "что будет, если...".
7. Обучаемый может запросить и получить помощь по изучаемому курсу, как у обучающей программы, так и у преподавателя.
8. Сохранение задаваемых вопросов и ответов на них.
9. Обучаемый может запросить и получить помощь по обучающей программе.

10. Обучающая программа должна накапливать информацию об ошибках при работе с обучающей программой, т.е. о её недостатках.
11. Обучаемый должен выполнить задания для самоконтроля и контрольные тесты.
12. При самостоятельной тренировке (без оценки, обучение) обучаемый может в любой момент ознакомиться с теорией, выполнять задания в любой последовательности. Ввод ответа подтверждается нажатием кнопки, при этом проверяется ответ. При неправильном ответе пользователь может выбрать: а)попробовать самостоятельно найти и исправить ошибку; б)посмотреть правильный результат или ход решения.
13. При контрольном тестировании теория недоступна, задания запускаются по очереди вне зависимости от правильности их выполнения. Ответ вводится нажатием кнопки. Проверяются ответы, выводятся правильные результаты, начисляются баллы, определяется оценка. Тестирование проходит определённое время. Для экономии времени можно пропускать задания, которые не удаётся решить сразу. Если остаётся время, к ним можно вернуться.
14. Наличие системы сбора и обработки статистической информации о каждом студенте, группе. (База данных студентов с сохранением пройденных тем, результатов тестирования)
15. Наличие системы сбора информации о часто встречающихся ошибках по изучаемой теме, всему курсу.
16. Анализ результатов теста преподавателем, обучающей программой.
17. Обучающая программа оценивает знания, даёт советы каждому обучающему персонально: над чем стоит поработать, какие определения, свойство, правило, теорему, аксиому и т.п. изучить, почему могли возникнуть ошибки (незнания или простой невнимательности).
18. Возможность распечатывать курс лекций, практические задачи.

### **1.3. Перспективы возможного продолжения работы**

Данная реферативная работа выполнялась для ознакомления с обучающими программами в компьютерных технологиях. **В дальнейшем планируется:**

1. Разработка обучающей программы по курсу математическая логика.
2. Разработка обучающей программы по курсу дискретная математика.
3. Применение программы в ГИП в курсе математическая логика.
4. Тестирование обучающей программы, устранение ошибок, описание опыта использования.
5. Участие в телеконференции в Интернет по тематике работы.
6. Переписка в Интернет по тематике работы. Следовательно, возможно продолжение работы, а также перерастание в исследовательскую.

**С учётом изученных материалов возможны следующие темы для (скорее всего) реферативных работ:**

1. Системы дистанционного обучения.
2. Повышение эффективности обучения с помощью компьютера.
3. Повышение мотивации к учению с помощью образовательных программ.
4. Программные среды, используемые для создания обучающих программ.
5. Роль компьютерных обучающих программ.
6. Прогноз национального рынка обучающих программ.
7. Обзор тестирующих программ.

## **2. РАЗРАБОТКА СПЕЦИАЛЬНЫЕ ОБУЧАЮЩИЕ КУРСЫ ПО ПОВЫШЕНИЮ КОМПЬЮТЕРНОЙ ГРАМОТНОСТИ**

### **2.1. Техническое задание**

#### **Введение**

Настоящее техническое задание распространяется на разработку электронного учебника по дисциплине «Компьютерной грамотности» для использования студентов специальности «Прикладная информатика» при изучении курса программирование.

#### **Назначение разработки**

Программа предназначена для использования студентами при изучении курсов " Компьютерной грамотности ", «Высокоуровневые методы информатики и программирования».

#### **Технические требования**

Требования к функциональным характеристикам.

Программа должна обеспечивать возможность выполнения следующих функций:

- содержать теоретический материал тем по дисциплине «Компьютерной грамотности»;
- содержать систему тестирования;
- содержать систему оценивания знаний по результатам тестирования;
- хранение результатов тестирования в памяти;
- содержать задания для практических работ.

Исходные данные:

Материал по дисциплине «Компьютерной грамотности».

Организация входных и выходных данных.

Входные данные поступают с клавиатуры.

Выходные данные отображаются на экране и при необходимости выводятся на печать.

Требования к надежности.

Предусмотреть контроль вводимой информации.

Предусмотреть блокировку некорректных действий пользователя при работе с системой.

Требования к составу и параметрам технических средств.

Система должна работать на IBM-совместимых персональных компьютерах.

Минимальная конфигурация:

- тип процессора – Pentium III и выше;
- объем оперативного запоминающего устройства – 256 Мб и более;
- объем свободного места на жестком диске – 40 Мб.

Рекомендуемая конфигурация:

- тип процессора – Pentium Celeron 1,6 ГГц;
- объем оперативного запоминающего устройства – 512 Мб;
- объем свободного места на жестком диске – 60 Мб.

Требования к программной совместимости.

Программа должна работать под управлением семейства операционных систем Windows (Windows XP / Vista / 7 и т.п.).

### **Требования к программной документации**

Разрабатываемые программные модули должны быть самодокументированы, т.е. тексты программ должны содержать все необходимые комментарии.

Разрабатываемая программа должна включать справочную информацию о работе программы, описания методов сортировки и подсказки учащимся.

В состав сопровождающей документации должны входить:

- пояснительная записка на пяти листах, содержащая описание разработки;
- руководство пользователя.

## Календарный план работ

Таблица 2.1 Календарный план работ

| № этап а | Название этапа  | Сроки этапа                | Чем заканчивается этап  |
|----------|---|----------------------------|---|
| 1        | Изучение предметной области.<br>Проектирование системы.<br>Разработка предложений по реализации системы                                   | 05.09.2010 –<br>15.10.2010 | Предложения по работе системы   |
| 2        | Разработка программного модуля по сбору и анализу информации со счетчиков и устройств управления.<br>Внедрение системы в процесс обучения | 16.10.2010 –<br>20.10.2010 | Программный комплекс, решающий поставленные задачи для автоматизации учебного процесса. |
| 3        | Тестирование и отладка модуля. Пробное внедрение системы в процесс обучения   | 21.10.2010 –<br>5.11.2010  | Готовая автоматизированная система по дисциплине «Программирование»                     |

### 2.2. Конструкторская часть

#### Постановка задачи

На основе анализа, проведенного в выпускной квалификационной работе, нами установлено, что учебный процесс в его классической его форме в связи с развитием новых информационных технологий устарел. В результате внедрения предлагаемого программного продукта существенно изменяются подходы к организации учебного процесса. Произошедшие изменения наглядно иллюстрируют диаграммы IDEF0 А, Б, В на рисунках 2.1, 2.2, 2.3.





C# (Си-шарп) — объектно-ориентированный язык программирования для платформы .NET. Разработан в 2000 году Андерсом Хейлсбергом, Скоттом Вилтамутом и Питером Гольде под эгидой Microsoft Research. Основным постулатом C# является высказывание: "всякая сущность есть объект". Язык основан на строгой компонентной архитектуре и реализует передовые механизмы обеспечения безопасности кода.

C# был создан специально для технологии ASP.NET. В то же время на C# полностью написана и сама ASP.NET.

C# — это полнофункциональный объектно-ориентированный язык, который поддерживает все три «столпа» объектно-ориентированного программирования: инкапсуляцию, наследование и полиморфизм. Он имеет прекрасную поддержку компонентов, надежен и устойчив благодаря использованию «сборки мусора», обработки исключений, безопасности типов.

Язык C# разрабатывался "с нуля" и вобрал в себя много полезных свойств таких языков, как C++, Java, Visual Basic, а также Pascal, Delphi и др. При этом необходимость обратной совместимости с предыдущими версиями отсутствовала, что позволило языку C# избежать многих отрицательных сторон своих предшественников.

Как и Java, C# разрабатывался для Интернет и примерно 75% его синтаксических возможностей аналогичны языку программирования Java, его также называют «очищенной версией Java. 10% подобны языку программирования C++, а 5% – заимствованы из языка программирования Visual Basic. Объем новых концептуальных идей в языке C# около 10%.

Выделение и объединение лучших идей современных языков программирования делает язык C# не просто суммой их достоинств, а языком программирования нового поколения [16].

#### **Функциональная схема**

На рисунке 2.4. изображена функциональная схема автоматизированной обучающей системы по дисциплине

«Программирование». Она показывает взаимодействия компонентов программного обеспечения с описанием информационных потоков, состава данных в потоках и указанием используемых файлов и устройств.

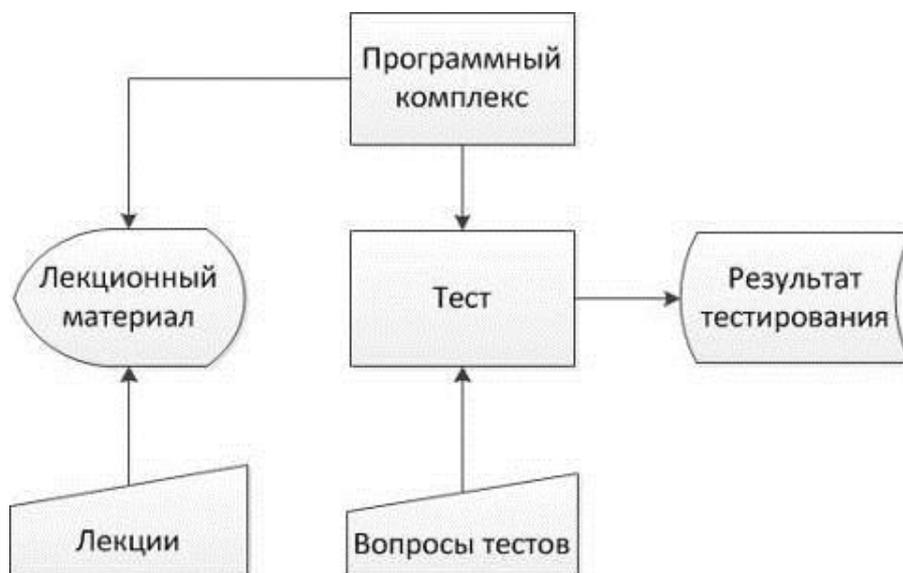


Рисунок 2.4. Функциональная схема программы

### 2.3. Проектирование системы

#### Модель вариантов использования

Диаграммы вариантов использования описывают функциональное назначение системы или то, что она должна делать.

Модель вариантов использования представляет функциональность системы или другого классификатора так, как она выглядит с точки зрения внешнего пользователя системы. Изображается такая модель в виде диаграмм использования.

Диаграмма использования представляет собой граф актёров, множество вариантов использования, заключенное в границы системы, ассоциации между актёрами и вариантами использования, отношения между вариантами использования и обобщения между актёрами. На основе выделенных требований была составлена диаграмма вариантов использования:

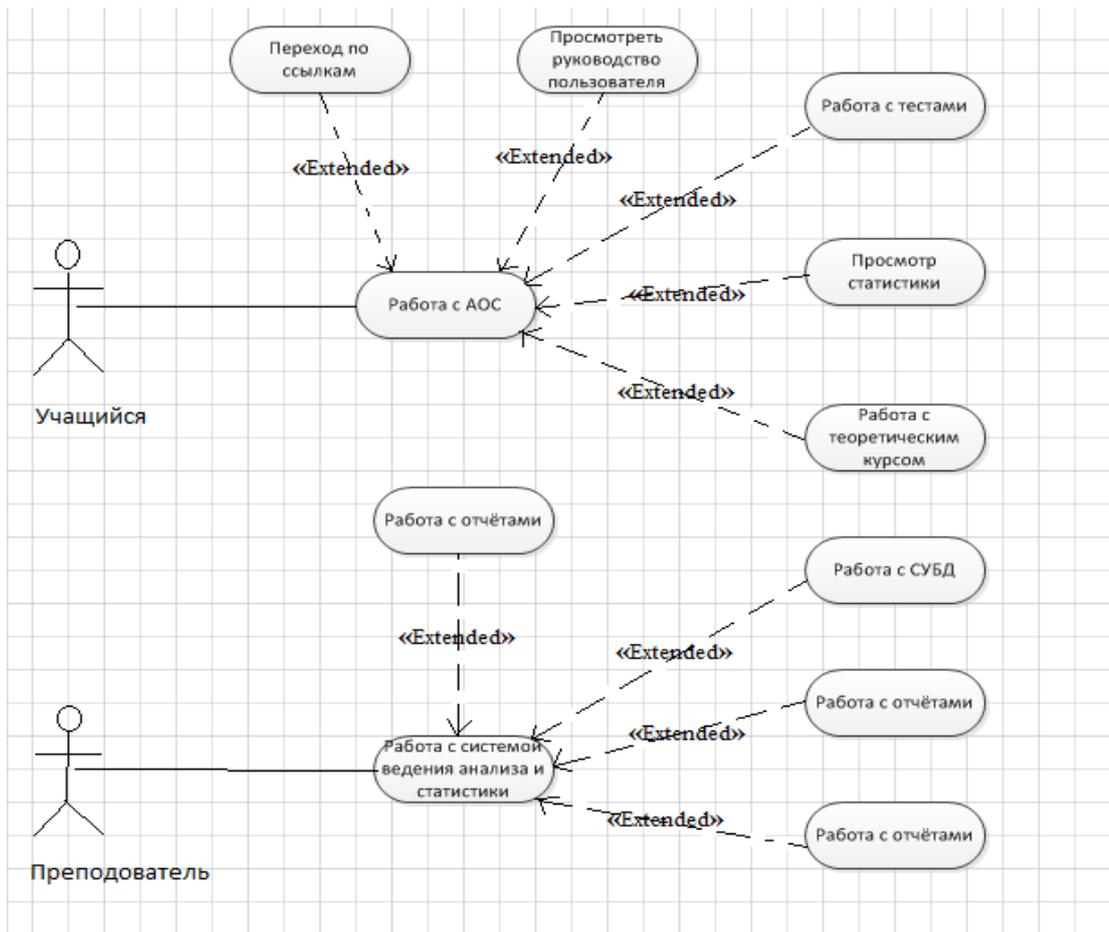


Рис. 3.1 Диаграмма вариантов использования

## **Модель предметной области**

Модель сущность-связь – модель данных, позволяющая описать концептуальные схемы предметной области. Для визуализации данной модели используются диаграммы сущность-связь (ER-диаграммы).

ER-диаграммы используются для разработки данных и представляют собой стандартный способ определения данных и отношений между ними. Таким образом, осуществляется детализация хранилищ данных. ER-диаграмма содержит информацию о сущностях системы и способах их взаимодействия, включает идентификацию объектов, важных для предметной области, свойств этих объектов и их отношений с другими объектами. Во многих случаях информационная модель очень сложна и содержит множество объектов.

Были выделены следующие ключевые сущности базы данных:

1. Ученик – клиент центра по обучению населения компьютерной грамотности.
2. Курс – номер группы проходящей обучение.
3. Тест – контрольные задания для проверки усвоения материала учащимся.
4. Автоматизированная обучающая система (АОС) – система, спомощью которой ведётся обучение.
5. Учебно-методический комплекс (УМК) – комплекс теоретической информации по информатике.

Между основными сущностями были выявлены следующие отношения:

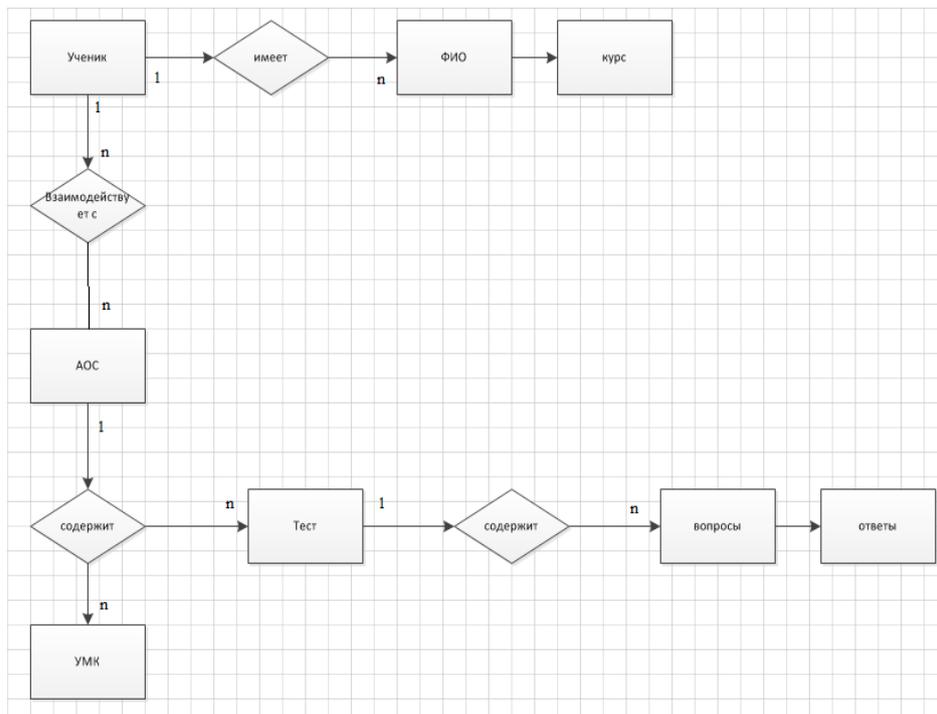


Рис. 3.2 ER-диаграмма

На основании ER-диаграммы была построена схема базы данных:

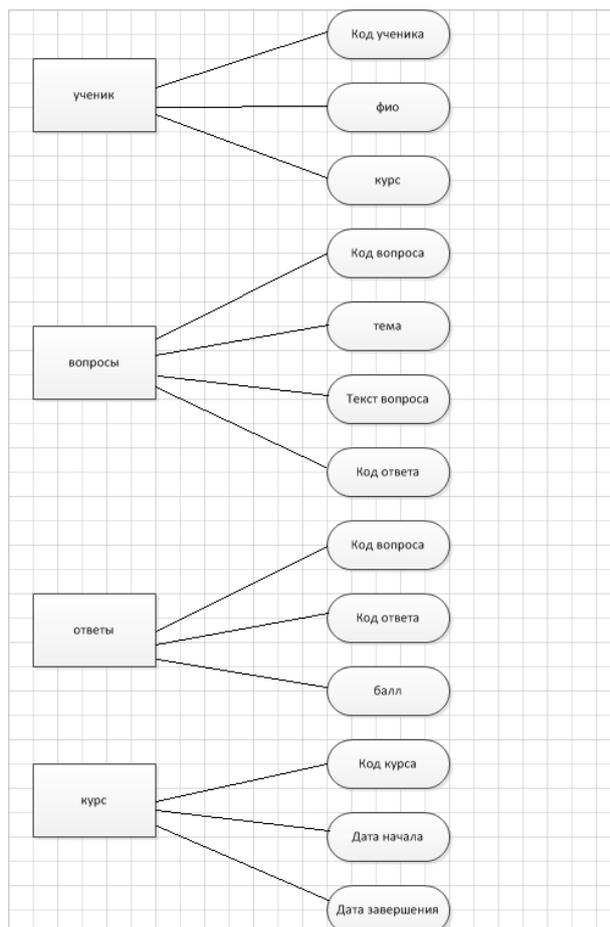


Рис. 3.3 Схема базы данных

## Топология сети

Термин «топология», или «топология сети», характеризует физическое расположение компьютеров, кабелей и других компонентов сети. Топология — это стандартный термин, который используется профессионалами при описании основной компоновки сети.

Выделяют 3 базовых топологии:

- 1) Шина
- 2) Кольцо
- 3) Звезда

Звезда — одна из наиболее распространённых топологий, поскольку проста в обслуживании. В основном используется в сетях, где носителем выступает кабель [витая пара](#). Чаще всего в компьютерных классах применяют именно эту топологию.

Звезда — базовая топология компьютерной сети, в которой все компьютеры сети присоединены к центральному узлу, образуя физический сегмент сети. Подобный сегмент сети может функционировать как отдельно, так и в составе сложной сетевой топологии. Весь обмен информацией идет исключительно через центральный компьютер или сетевой коммутатор, на который таким способом возлагается очень большая нагрузка, поэтому ничем другим, кроме сети, он заниматься не может. Как правило, именно центральный компьютер является самым мощным, и именно на него возлагаются все функции по управлению обменом. Никакие конфликты в сети с топологией звезда в принципе невозможны, потому что управление полностью централизовано.

Существует два варианта топологии звезда:

- 1) Активная звезда (В центре сети содержится компьютер, который выступает в роли сервера);
- 2) Пассивная звезда (В центре сети с данной топологией содержится сетевой концентратор)

В центре повышение компьютерной грамотности на базе центральной библиотеки компьютерная сеть построена по топологии Пассивная звезда.

Достоинства:

- 1) выход из строя одной рабочей станции не отражается на работе всей сети в целом;
- 2) хорошая масштабируемость сети;
- 3) лёгкий поиск неисправностей и обрывов в сети;
- 4) высокая производительность сети (при условии правильного проектирования);
- 5) гибкие возможности администрирования.

Недостатки:

- 1) выход из строя центрального концентратора обернётся неработоспособностью сети (или сегмента сети) в целом;
- 2) Для прокладки сети зачастую требуется больше кабеля, чем для большинства других топологий;
- 3) конечное число рабочих станций в сети (или сегменте сети) ограничено количеством портов в центральном концентраторе.



Рис. 3.4 Топология сети

## Диаграмма развёртывания

Диаграммы развертывания показывает конфигурацию узлов, где производится обработка информации, и то, какие компоненты размещены на каждом узле.

Диаграммы развертывания обычно включают в себя:

- 1) узлы
- 2) отношения зависимости и ассоциации

Применение:

Диаграммы развертывания используются в трех случаях:

- 1) моделирование встроенных систем
- 2) моделирование клиент-серверных систем
- 3) моделирование полностью распределенных систем

Автоматизированная обучающая система для центра обучения населения компьютерной грамотности представляет собой клиент-серверную систему с "толстым" клиентом т.к. в стандартную поставку техники входят компьютеры с равными вычислительными мощностями и перекладывать всю нагрузку на 1 ПК нецелесообразно.

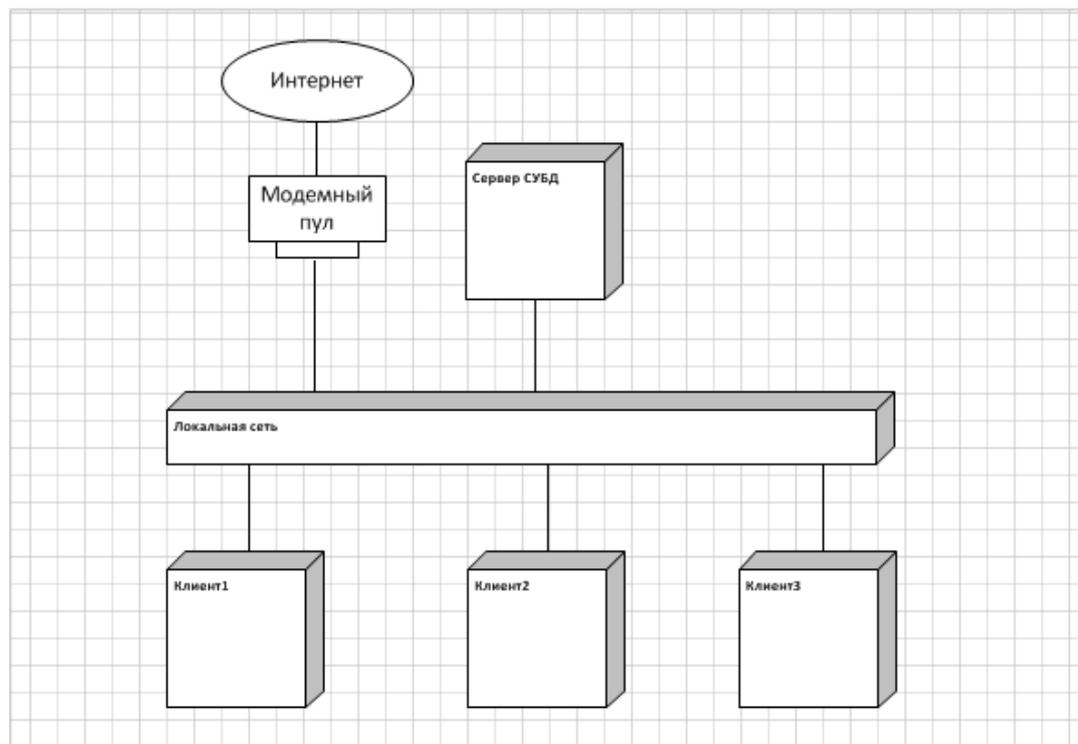


Рис. 2.5 Диаграмма развёртывания

## **Обоснование выбора платформы проектирования**

Основными платформами проектирования, позволяющими реализовать подобную систему, являются:

Delphi 7 — интегрированная среда разработки ПО на языке Delphi, созданная фирмой Embarcadero Technologies. Delphi 7 включает полностью интегрированный, быстрый компилятор и отладчик Windows, визуальную среду для разработки интерактивных приложений, платформу визуальных компонентов, содержащую свыше 250 стандартных классов и компонентов, а также обширные возможности для подключения к базам данных. Delphi является мощным и простым в использовании инструментом для создания автономных программ, обладающих графическим интерфейсом, или 32-битных консольных приложений. Delphi содержит встроенные средства, использующие небольшой объем ресурсов и обеспечивающие высокопроизводительный доступ ко всем популярным системам управления базами данных, включая Microsoft SQL Server, Oracle, MySQL, InterBase, Firebird и другие.

Delphi 7 поддерживает технологию ADO (ActiveX Data Object – объекты данных, построенные как объекты ActiveX) корпорации Microsoft. На основе этой технологии созданы соответствующие компоненты-наборы включенные в репозиторий среды разработки. Основным достоинством ADO является её естественная ориентация на создание «облегченного» клиента.

Преимущества:

1. Удобный интерфейс.
2. Высокая производительность вкуче с относительно небольшими размерами программ.
3. Возможность подключения дополнительных библиотек.
4. Широкий выбор средств взаимодействия с базами данных.
5. Недостатки:
6. Сложность реализации больших и сложносвязанных проектов.
7. Отсутствие проверки кода «на лету».

Microsoft Visual Studio 2010 — платформа разработки от компании Microsoft, обладает интегрированной поддержкой разработки через тестирование и новые инструменты отладки, что позволяет быстро и без труда находить и устранять ошибки. Visual Studio 2010 поставляется вместе с NET Framework 4 и поддерживает разработку приложений для Windows 7. Платформа также поддерживает IBM DB2, Oracle и Microsoft SQL Server [6].

Преимущества:

1. Большой выбор инструментов проектирования и программирования.
2. Возможность писать программы на нескольких языках программирования;

Недостатки:

1. Сложность интерфейса.
2. Проблемы с реализацией компоновки «база + программный интерфейс».

На основе данного анализа было принято решение разрабатывать систему на платформе Delphi 7. Платформа имеет множество компонентов для работы с базами данных, простую, наглядную реализацию приложений и может решить большинство задач предметной области за довольно короткий промежуток времени.

## Алгоритм работы программы

На рисунке рисунок 2.4 изображен алгоритм работы электронного учебника по дисциплине «Компьютерной грамотности».

Алгоритм — это точное предписание, которое задаёт вычислительный (алгоритмический) процесс, начинающийся с произвольного исходного данного и направленный на получение полностью определяемым этим исходным данным результата [8].

Пользователь запускает программу. Далее программа предоставляет возможность выбрать тему лекции. После выбора лекции появляется окошко с текстом лекции. После изучения материала лекции, пользователь может пройти тестирование по материалам лекции. После прохождения теста выдается результат решения теста. После объявления результата пользователь может завершить работу с программой или перейти к списку лекции.



Рисунок 3.5. Алгоритм работы программы в виде блок-схемы

Программный код реализации программы представлен в приложении.

## Проектирование интерфейса

При разработке интерфейса автоматизированной обучающей системы по дисциплине «Компьютерной грамотности» мы руководствовались принципом простоты и удобства использования программы. В программе задействовано четыре формы. Функции первой формы (рисунок 3.6): выбор темы лекции, из нескольких предложенных программой, и отображение на экране лекционного материала. Вторая форма – тестовая. Она необходима для реализации функции тестирования студентов (рисунок 3.7). Третья форма нужна для ввода с клавиатуры фамилии и имя пользователя перед началом тестирования. Четвертая форма служит для отображения результатов тестирования пользователей программы (рисунок 3.8).

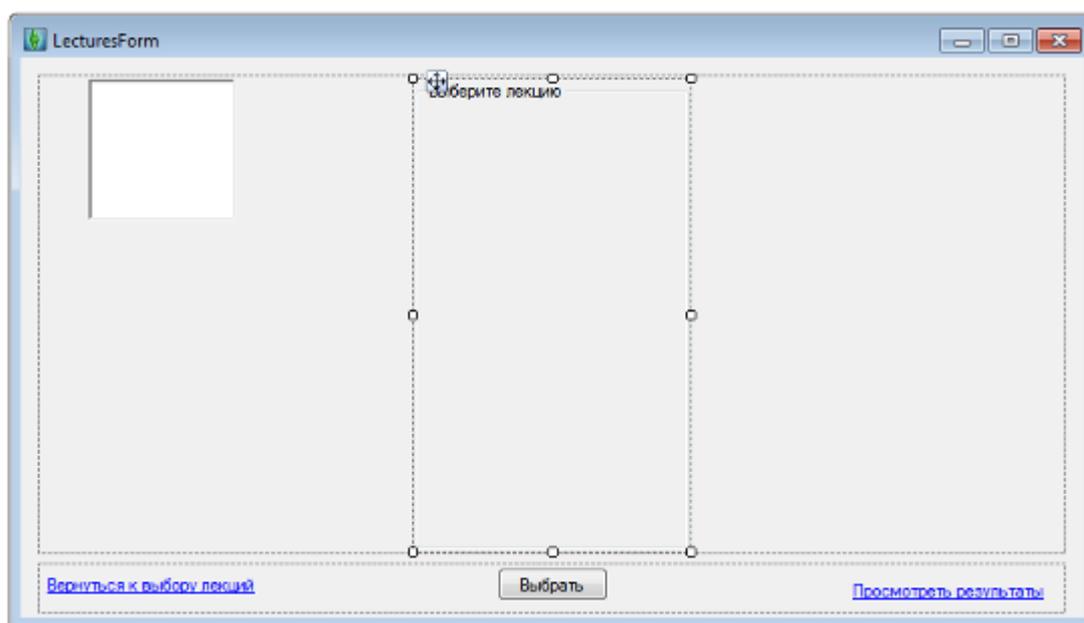


Рисунок 3.6. Формы выбора и чтения лекций в режиме конструктора



Рисунок 3.7. Форма тестирования студентов

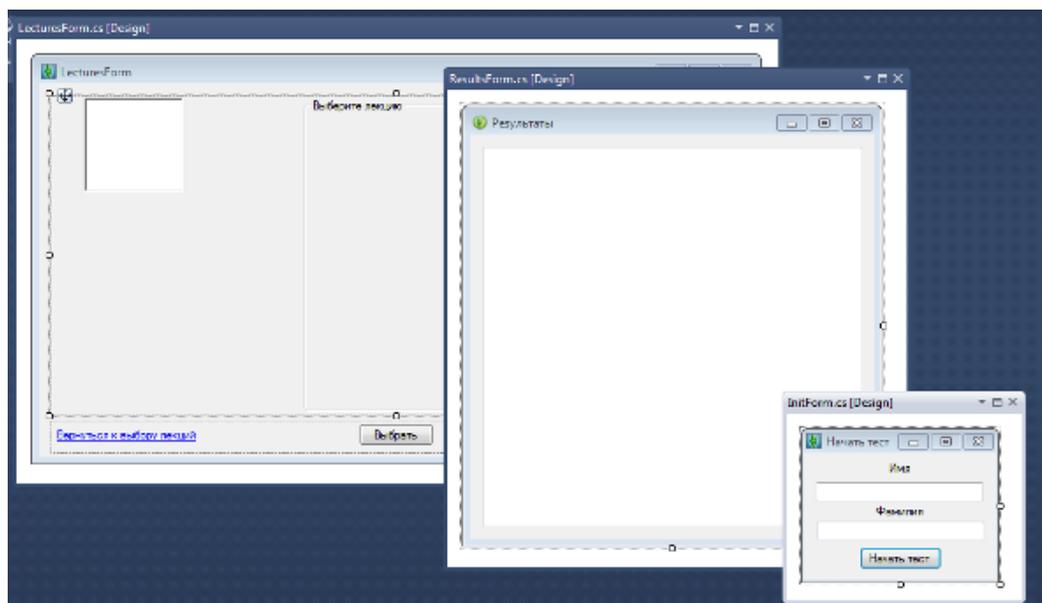


Рисунок 3.8. Формы программы в режиме конструктора

### **3. ЭКСПЛУАТАЦИОННАЯ И ЭКОНОМИЧЕСКАЯ ЧАСТЬ**

#### **3.1. Эксплуатационная часть**

##### **Требования к вычислительной системе**

1. Операционная система Windows XP/Vista/7 32/64 bit.
2. Наличие установленной программы Microsoft Net Framework 2.0 и выше.  
При отсутствии этой утилиты система работать не будет.
3. Тип процессора – Pentium Celeron 1,6 ГГц и выше.
4. Объем оперативного запоминающего устройства – 512 Мб и более.
5. Объем свободного места на жестком диске – 60 Мб.

##### **Установка программного продукта**

Установка данного программного продукта не требуется.

##### **Инструкция по эксплуатации**

Автоматизированная обучающая система по дисциплине «Компьютерной грамотности» поставляется на лицензионном диске.

1. Вставьте лицензионный диск в дисковод.
2. Откройте папку «Мой компьютер».
3. В папке «Мой компьютер» откройте диск с программой.
4. Запустите программу, кликнув по ярлыке запуска.
5. Откроется окно с выбором тем лекций.
6. Выберите тему лекции и нажмите кнопку «Выбрать».
7. В окне появится лекция по выбранной теме.
8. Далее вы можете вернуться к выбору лекций, нажав ссылку «Вернуться к выбору лекций» или пройти тест, нажав кнопку «Тест».
9. Перед началом теста введите имя и фамилию с клавиатуры и нажмите кнопку «Начать тест».
10. Запустится тест. Выберите правильный, на ваш взгляд вариант ответа и нажмите кнопку «Следующий вопрос». В тесте есть вопросы с

многовариантным ответом. Выберете несколько, правильных вариантов и нажмите кнопку «Следующий вопрос».

11. Когда тест будет пройден, появится диалоговое окно с результатами решения и вопросом: «Перейти к выбору лекций». Если нажать кнопку «Да» то появится окно с выбором лекции, если нажать кнопку «Нет», то программа завершит работу.

12. В окне с выбором лекции есть ссылка «Посмотреть результаты». Если перейти по этой ссылке можно увидеть все результаты решения тестов, а именно число прохождения теста, фамилию и имя тестируемого и его результат решения.

#### 4. Демонстрационная часть работы продукта.

Запускаем программу с лазерного диска при двойном щелчке по ярлыку запуска.

На рисунке 4.1 показано, что при запуске программы пользователю будет предложено выбрать тему для самостоятельного изучения.

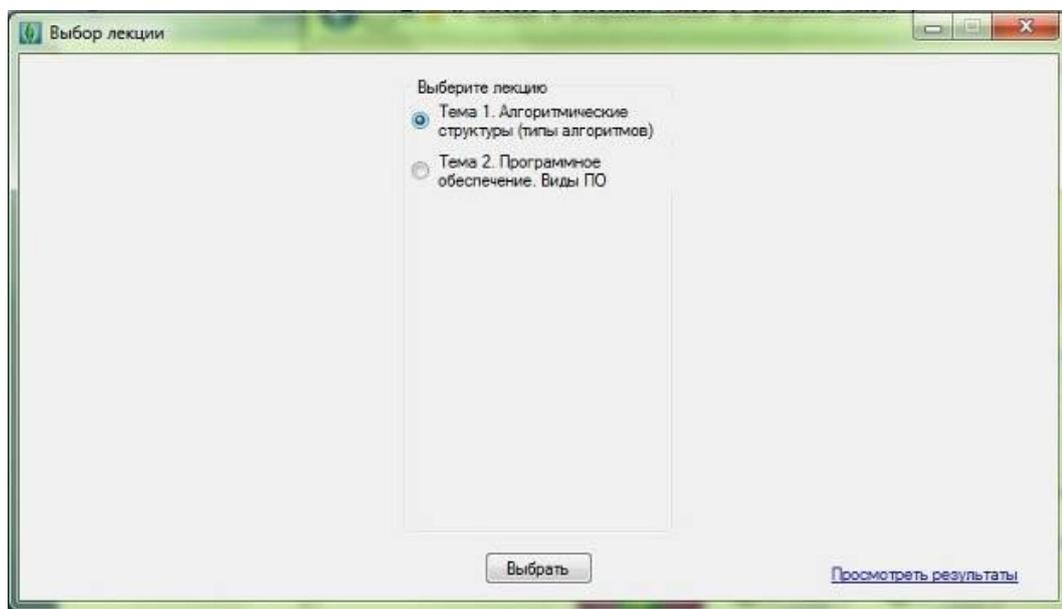


Рисунок 4.1. Выбор темы

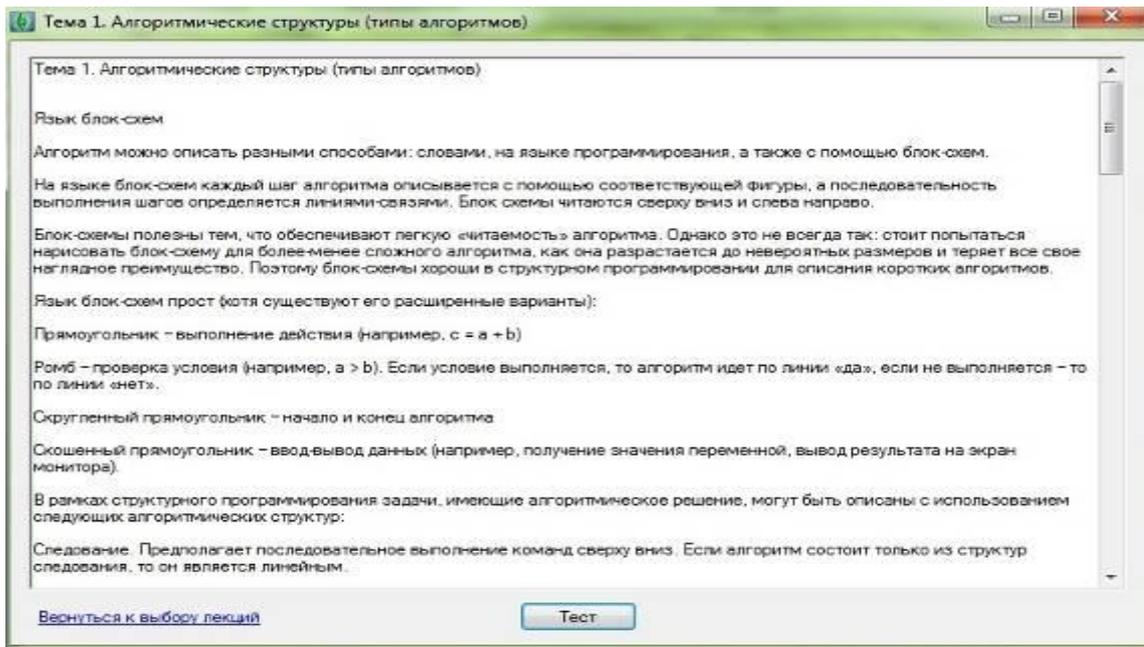


Рисунок 4.2. Лекция по теме.

На рисунке 4.2 приведен пример лекции. После ознакомления с лекционным материалом по теме, пользователь может пройти тест.

The image shows a small dialog box titled "Начать тест". It has two input fields: "Имя" and "Фамилия". Below the fields is a button labeled "Начать тест".

Рисунок 4.3. Ввод информации о студенте перед началом тестирования

Перед прохождением тестирования нужно ввести имя и фамилию пользователя системы. В каждом тесте по десять вопросов (рисунок 4.3).

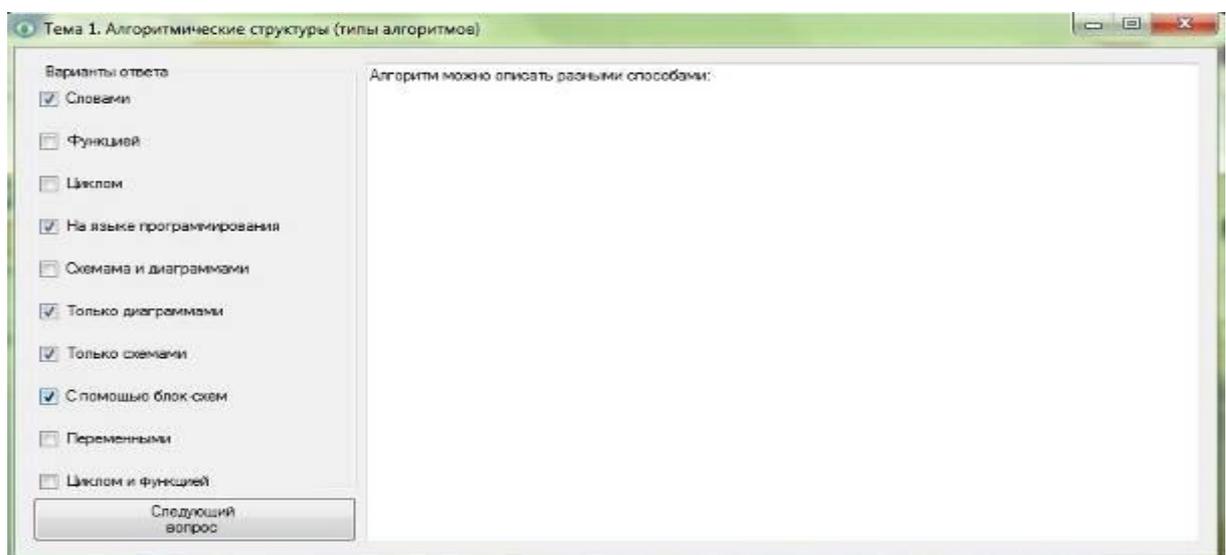


Рисунок 4.4. Вопрос с многовариантным ответом

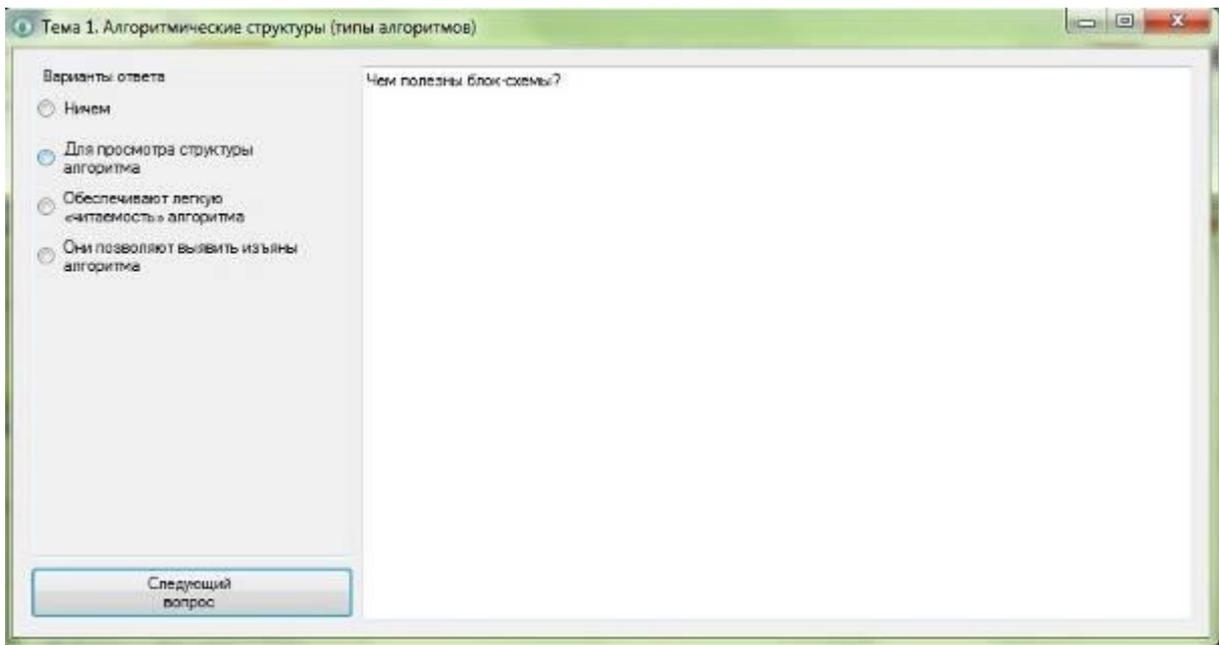


Рисунок 4.5. Вопрос с одним вариантом ответа

В тесте встречаются вопросы с многовариантными (рисунок 4.4) и одновариантным (рисунок 4.5) ответами.

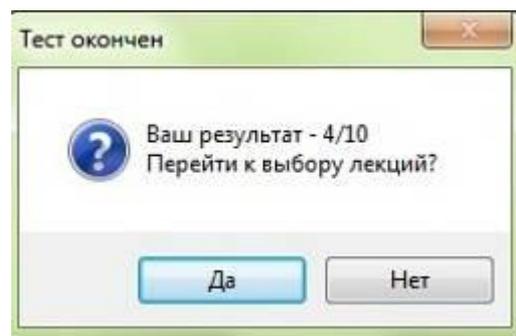


Рисунок 4.5. Результаты тестирования

После прохождения теста пользователь сможет увидеть результаты решения. На рисунке 4.5 показан результат решения одного из тестов. Количество правильных ответов – четыре из десяти возможных. Пользователю предложена дальнейшая работа с программой. Для этого на вопрос «Перейти к выбору лекций нажать кнопку «Да», или завершить работу с программой, нажав кнопку «Нет».

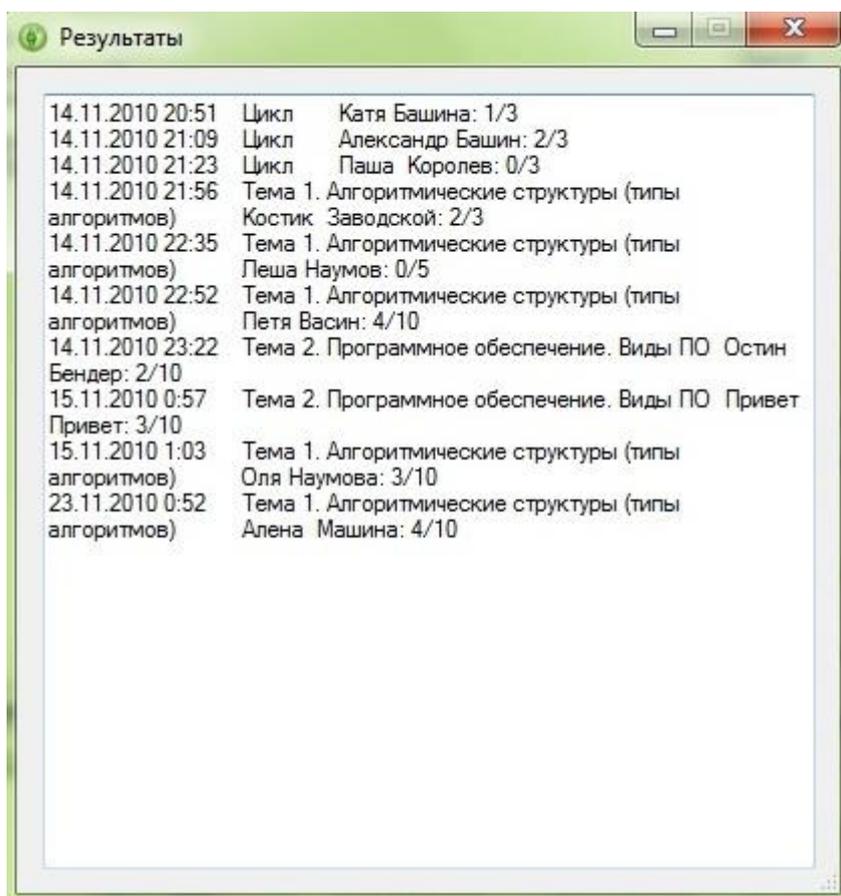


Рисунок 4.6. Результаты тестирования

Все результаты тестов можно посмотреть после тестирования (рисунок 4.6). В таблице результатов отражено дата прохождения теста, фамилия, имя студента и его результат.

### 3.2. Экспериментальная часть

Серьезные приложения должны надежным образом обрабатывать исключительные ситуации, сохранять, если возможно, выполнение программы или, если это невозможно, аккуратно ее завершать. Написание кода, обрабатывающего исключительные ситуации, всегда было непростой задачей, и являлось источником дополнительных ошибок [16].

Проведем эксперимент по созданию исключительной ситуации с электронным учебником по дисциплине «Программирование».

Пусть файл с результатами тестирования удалили студенты из папки с программой. Запустим программу (рисунок 5.1).

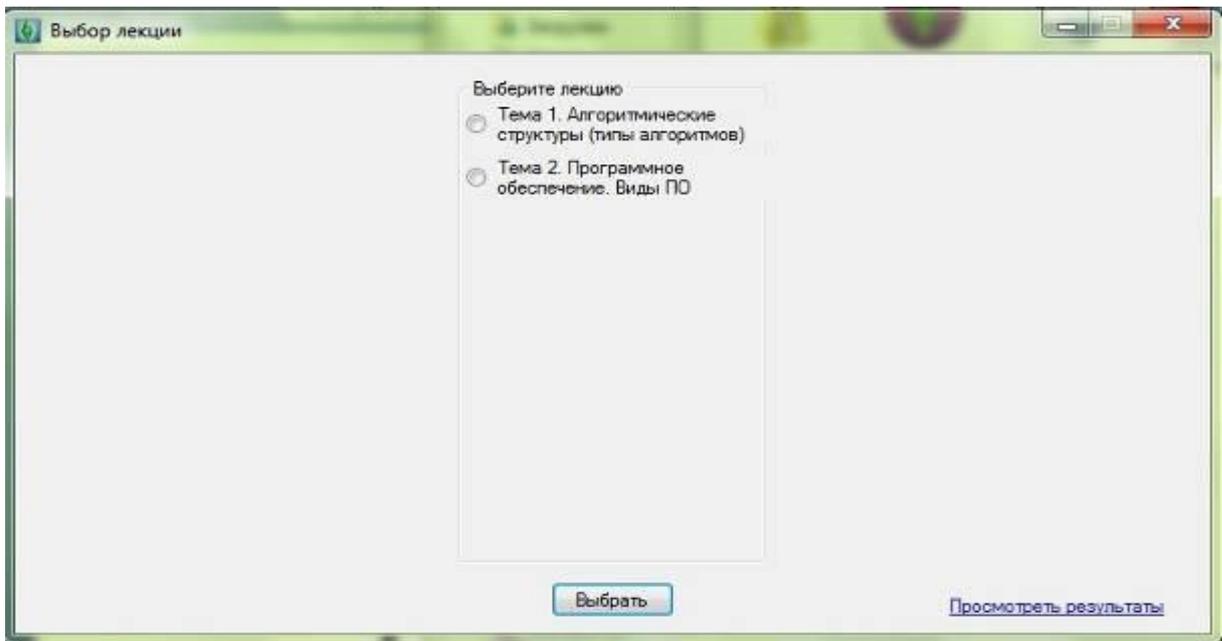


Рисунок 5.1. Запуск программы в исключительной ситуации

Программа запустилась без ошибок и системных сообщений. Теперь попробуем посмотреть результаты тестирования прошлых студентов (рисунок 5.2).

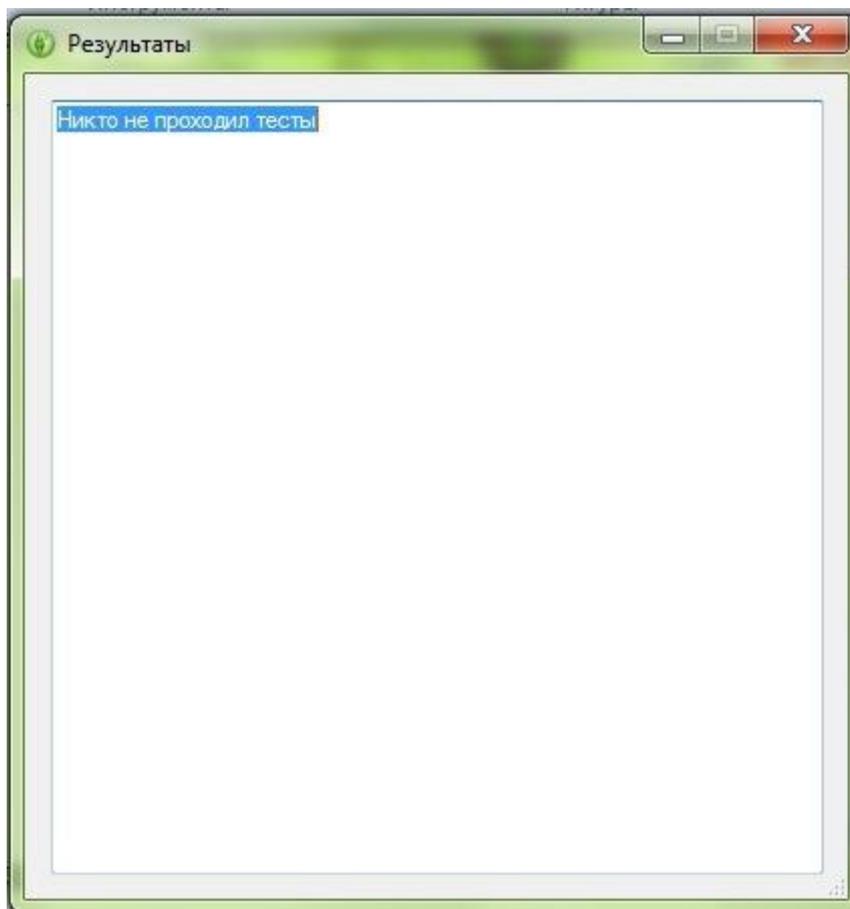


Рисунок 5.2. Поведение программы в исключительной ситуации

К сожалению, результаты прошлого тестирования студентов утеряны. Программа в корневом каталоге создала новый файл и при повторном тестировании будет произведена запись результатов в этот файл. Значит, тестирование можно провести заново и результаты тестирования будут фиксироваться и сохраняться.

Проведем еще один эксперимент по созданию исключительной ситуации с электронным учебником по дисциплине «Программирование». Теперь студенты удалили файлы с лекциями по темам. При попытке запуска лекций программа выдаст системную ошибку (рисунок 5.3).

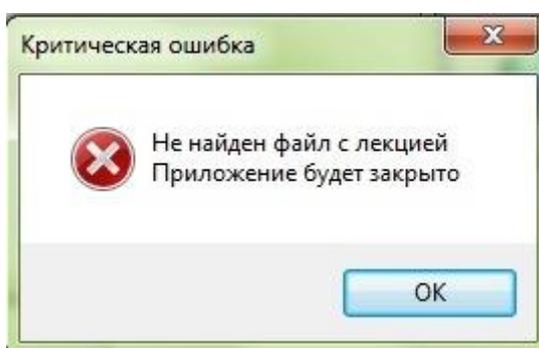


Рисунок 5.3. Критическая ошибка программы

### 3.3. Экономическая часть

Программный продукт в данной выпускной квалификационной работе будет выполняться индивидуальным разработчиком (студенткой). Решение данной задачи будет проводиться на ЭВМ. Общее ознакомление с предметной областью проходило до решения об изучении и возможности создания программного продукта для решения задач предметной области (определяется техническим заданием).

Предполагаемая продолжительность разработки программного комплекса была выбрана в количестве 45 дней. Это позволит не затягивать с выходом продукта и в тоже время не сильно сокращает сроки, что обеспечивает нормальную стоимость работ и желаемую прибыль.

#### Определение вида и длительности работ

Продолжительности этапов разработки программного комплекса сведены в таблицу 3.1.

Продолжительность первого этапа и его работ определяется по результатам предварительного исследования. Общая трудоемкость этого этапа составляет 18% от всего времени.  $T_1 = T_{\text{раб}}/100 * 18 = 8$  дней.

Продолжительность остальных этапов и работ определяется по методу удельных весов:

$$T_1 = \frac{T_2 * Y_1}{Y_2} \quad (3.1),$$

где  $T_1$ ,  $T_2$  – соответственно, определяемая трудоемкость этапа и трудоемкость известного этапа;  $Y_1$ ,  $Y_2$  – соответственно, удельный вес тех же этапов в общей трудоемкости разработки программного комплекса.

Этапы и трудоемкость работ показаны в таблице 3.1.

Таблица 3.1 Продолжительность этапов разработки программного комплекса

| № этапа | Относительная доля времени, % | Продолжительность, дни |
|---------|-------------------------------|------------------------|
| 1       | 18                            | 8                      |
| 2       | 40                            | 18                     |
| 3       | 30                            | 13,5                   |
| 4       | 12                            | 5,5                    |
| Всего:  | 100                           | 45                     |

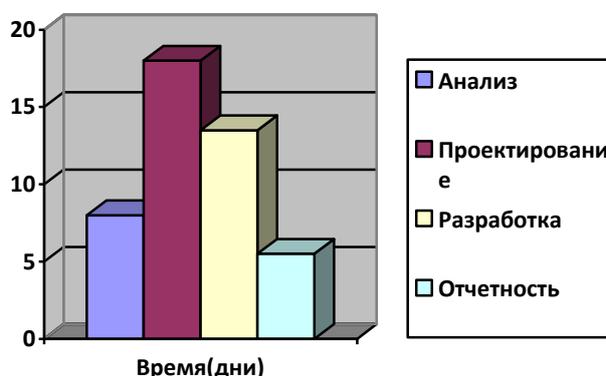


Рисунок 3.1. Этапы и трудоемкость работ

Оценивается наиболее вероятная продолжительность работ по формуле

$$t_{н.в.} = \frac{T}{P * K_{пер}} = 8 \quad (3.2),$$

где P – число исполнителей на данной работе.

Коэффициент перехода рабочих дней в календарные дни  $K_{пер} = 45 / 365 = 0.12$ .

Ожидаемое время рассчитывается по формуле:

$$t_{ожд} = \frac{t_{min} + 4 * t_{н.в.} + t_{max}}{6} = 4 \quad (3.3),$$

где  $t_{ожд}$  – трудоемкость выполнения отдельных видов работ, человеко-дни;  $t_{min}$  – минимально возможная трудоемкость выполнения отдельных видов работ, дни;  $t_{max}$  – максимально возможная трудоемкость выполнения отдельных видов работ, дни;  $t_{нв}$  – наиболее вероятная трудоемкость выполнения отдельных видов работ, дни.

$$t_{max} = 15$$

$$t_{min} = 10$$

Итог: ожидаемое время работы над проектом 37 дней не превышает изначально предполагаемую продолжительность работ.

### **Определение заработной платы исполнителей**

Было принято решение о том, что заработная плата не будет начисляться, так как исполнитель работы один. Т.е. доход исполнителя будет равен прибыли, полученной от продажи программного продукта.

Без начисления заработной платы затраты на создание программного продукта существенно сократятся.

### **Составление сметы затрат**

Сметная стоимость определяется статьями калькуляции.

1. Накладные расходы.
2. Прочие расходы: диски и упаковка (100 шт.). Диски: 100 шт. по 7 сума, полиграфические услуги: 100шт. упаковок по 10 сум. Итого 1700 сум.

Смета затрат на выполнение представлена в таблице 3.3.

Таблица 6.2 Смета затрат на выполнение

| Наименование статьи затрат | Сумма, сум. | Удельный вес статьи, % | Основание для расчета |
|----------------------------|-------------|------------------------|-----------------------|
| Накладные расходы          | 500         | 23                     | Анализ затрат         |
| Прочие расходы             | 1700        | 77                     | Анализ затрат         |
| Итого                      | 2200        | 100                    | —                     |

### Расчет предполагаемой прибыли

Итак, общие затраты на создания программного продукта составляют 2200 сумов. Исходя из 50% рентабельности, сумму, которую мы хотим выручить, рассчитаем по формуле

$$C = Z + \Pi = 2200 + 1100 \approx 3300 \text{ сум.} \quad (3.4)$$

Эта сумма без налога на добавленную стоимость (НДС). Планируется ее получить от продажи 100 программных продуктов. Анализ рынка говорит, что реализовать такое количество не составит особых проблем. Исходя из этого, посчитаем цену одного программного комплекса без учета НДС

$$Ц = C / 5 = 3300 / 100 = 33 \text{ руб.} \quad (3.5)$$

Теперь посчитаем окончательную цену одного программного продукта с учетом НДС 18%.

$$\text{НДС} = ((Z + \Pi) / 100) * 0.18 = 33 * 0.18 \approx 6 \text{ сум.} \quad (3.6)$$

С учетом НДС цена программного продукта составляет:

$$Ц_{\text{ит}} = Ц + \text{НДС} \approx 40 \text{ руб.} \quad (3.7)$$

Процентное соотношение себестоимости, прибыли и налогов для ста комплектов продукта представлены на рисунке 2.

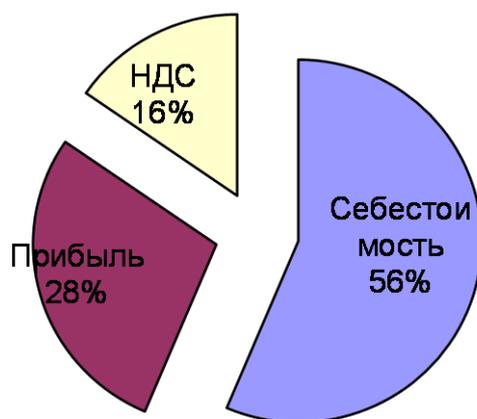


Рисунок 3.2. Процентное соотношение себестоимости, прибыли и налогов для 100 комплектов продукта

На рисунке 3.2 показана динамика роста прибыли от продажи программного продукта относительно числа экземпляров.

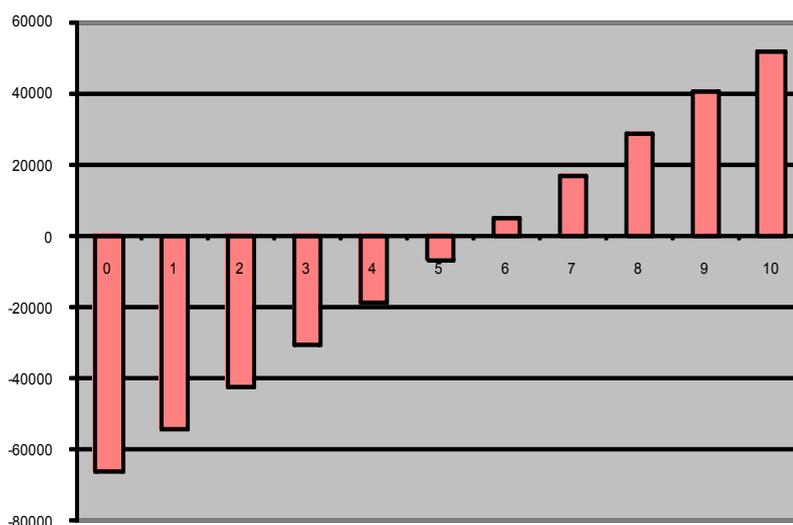


Рисунок 3.3. Динамика роста прибыли от продажи ПП

### **Экономическая целесообразность разработки**

При разработке программного продукта значительное внимание уделяется экономическому обоснованию необходимости проведения данного комплекса работ по его созданию.

Стоимость конечного продукта для потребителя не высока всего 40 руб. Разработка данного программного продукта ведется для образовательных учреждений. Поэтому цена одного лицензионного диска очень важна. Чем ниже цена, тем больше вероятность, что программный

продукт купят. При цене всего 40 рублей, диски с программой быстро купят. Покупателями станут преподаватели вузов и учителя других коммерческих учебных заведений. При продаже 100 лицензионных дисков, разработка программы не просто окупится, а принесет прибыль в размере 1100 рублей. Если при продаже всех лицензионных дисков с программой, на рынке все еще будет спрос, то при дополнительном выпуске лицензионных копий, при минимальных затратах, мы получим еще какой-то процент прибыли.

В итоге можно сделать вывод, что разработка данного программного продукта целесообразна.

## **4. БЕЗОПАСНОСТЬ ЖИЗНЕДЕЯТЕЛЬНОСТИ**

### **4.1. Эргономика**

Усложнение производственных процессов и оборудования изменили функции человека в современном производстве: возросла ответственность решаемых задач; увеличился объем информации, воспринимаемой работающим и быстрейшее действие оборудования. Работа человека стала сложнее, возросла нагрузка на нервную систему и снизилась нагрузка физическая. В ряде случаев человек стал наименее надежным звеном системы «человек-машина». Возникла задача обеспечения надежности и безопасности работы человека на производстве. Эту задачу решает эргономика и инженерная психология.

**Эргономика** (от греческого *ergon* - работа и *nomos* - закон) - научная дисциплина, изучающая человека в условиях его деятельности, связанной с использованием машин. Цель эргономики - оптимизация условий труда в системе "человек-машина" (СЧМ). Эргономика определяет требования человека к технике и условия ее функционирования. Эргономичность техники является наиболее обобщенным показателем свойств и других показателей техники.

Эргономика - наука о том, как люди с их различными физическими данными и способами функционирования взаимодействуют с окружающей рабочей средой (оборудованием и машинами, которыми они пользуются).

Цель эргономики состоит в том, чтобы обеспечить комфорт, эффективность и безопасность при пользовании компьютерами уже на этапе разработки клавиатур, компьютерных плат, рабочей мебели и др. для устранения физического дискомфорта и проблем со здоровьем на рабочем месте. В связи с тем, что все больше людей проводят много времени перед компьютерными мониторами, ученые многих областей, включая анатомию, психологию и охрану окружающей среды, вовлекаются в изучение правильных, с точки зрения эргономики, условий работы.

Большинство изготовителей мебели не учитывают индивидуальные особенности человеческого тела при проектировании автоматизированных рабочих мест. Обустройство эргономически оборудованного места может потребовать дополнительных расходов. Если позволяет Ваш бюджет, покупайте эргономически разработанную мебель типа стульев, полок и столов, которая может быть приспособлена к Вашим индивидуальным физическим данным.

**Инженерная психология** - научная дисциплина, изучающая закономерности информационного взаимодействия человека и техники для проектирования, создания и эксплуатации системы "человек-машина". Инженерная психология исследует процессы приема, хранения, переработки и реализации информации человеком. На основании закономерностей психических, психофизиологических процессов и свойств человека она определяет требования к техническим устройствам и построению системы "человек-машина", а также требования к свойствам человека-оператора.

В качестве обобщенных показателей деятельности оператора и системы "человек-машина" инженерная психология использует эффективность, надежность, точность, быстродействие.

Научную основу эргономики составляют анатомия, физиология и психология. Анатомия составляет теоретическую основу антропометрии и биомеханики.

**Антропометрия** - измерение человека позволяет получить данные, необходимые для правильного расположения органов управления и определения размеров рабочих пространств. Важным моментом при этом является определение границ колебаний размеров, в которых учитывается потребный объем выборки, выражаемый в перцентилях. Так, 90-й перцентиль представляет результаты измерений, показывающих, что 90% измеряемой группы имеют определенные размеры меньше, а 10% больше средних для данной группы. На практике любая конструкция рассчитывается на 90% населения.

**Биомеханика** - изучает приложение сил телом человека. Она дает рекомендации, как необходимо эффективно прилагать силы: усилие должно создаваться массой тела, а не мышц; наиболее полно должны использоваться мышцы, передвигающие сустав вокруг его центрального участка.

**Физиология в эргономике** дает закономерности процесса производства энергии организмом человека. Вырабатываемая энергия организма оценивается по потреблению кислорода. Психология вносит в эргономику теорию деятельности человека, основанную на информационной модели человека-оператора; теорию обучения и теорию организации, связанную с проектированием работы.

**Задачами эргономики** как прикладной дисциплины являются:

- проектирование системы "человек-машина", то есть распределение функций между человеком и машиной;
- проектирование рабочего пространства так, чтобы физическое окружение соответствовало характеристикам человека;
- проектирование окружающей среды в соответствии с требованиями оператора;
- проектирование рабочих ситуаций (продолжительность рабочего дня, перерывы для отдыха и т.п.).

**Инженерная психология,** как это следует из вышеизложенного, является практически составной частью эргономики, решающая задачи организации системы "человек-машина" путем:

- распределения функций между человеком и машиной;
- анализа функций, выполняемых человеком в системе "человек-машина";
- проектирования системы информации, выбора чувствительного канала;
- конструирования средств управления;
- проектирования рабочих мест;
- обеспечение удобства технического обслуживания машин;
- подбора кадров и их профессиональной подготовки.

Учет, эргономических требований должен осуществляться на всех этапах проектных решений и включает:

- Разработку профессиограммы, определяющей цели и задачи трудовой деятельности, ее психофизиологические характеристики, требования к человеку и технике.
- Анализ и уточнение назначения, принципов действия и конструкции техники, ее характеристик применительно к целям трудовой деятельности.
- Распределение функций между человеком и техникой на основе оценки качества выполнения задач человеком и машиной и общей эффективности системы.
- Установление последовательности выполняемых человеком операций и определение объема и формы представления информации.
- Ориентационную оценку надежных, временных и точностных требований к деятельности человека.

На основании рассмотренных работ определяется: состав специалистов, их функции и организация работы; состав средств отображения информации,

органов управления рабочих мест и пультов управления; компоновка средств отображения информации и органов управления, размещение рабочих мест в производственных помещениях.

### **Связь человека с окружающей средой и параметрами рабочего места.**

**Рабочее место** - это зона, в которой совершается трудовая деятельность исполнителя или группы исполнителей. Рабочие места могут быть индивидуальными и коллективными, универсальными, специализированными и специальными.

**Общие требования**, которые должны соблюдаться при проектировании рабочих мест, следующие:

- достаточное рабочее пространство для человека;
- оптимальное положение тела работающего;
- достаточные физические, зрительные и слуховые связи между человеком и машиной;
- оптимальное размещение рабочего места в помещении;
- допустимый уровень действия факторов производственных условий; оптимальное размещение информационного и моторного поля;
- наличие средств защиты от производственных опасностей.

Конструирование должно обеспечивать зоны оптимальной и легкой досягаемости моторного поля рабочего места и оптимальную зону информационного поля рабочего места. Угол обзора по отношению к горизонтали должен составлять 30-40°.

Выбор рабочего положения должен учитывать усилия, затрачиваемые человеком, размах движений, необходимость перемещений, темп операций. Выбор рабочей позы должен учитывать физиологию человека, а параметры рабочего места определяются выбором положения тела при работе (сидя, стоя, переменнo).

Рабочие места для выполнения работ «сидя» организуются при легкой работе и средней тяжести, а при тяжелой - рабочая поза - "стоя".

В конструкции оборудования и организации рабочего места необходимо предусматривать возможности регулирования отдельных элементов, чтобы обеспечить оптимальное положение работающего.

Проектирование оборудования должно обеспечить его соответствие антропометрическим и биомеханическим характеристикам человека на основе учета динамики изменения размеров тела при его перемещении, диапазона движений в суставах.

Для учета в конструкции оборудования антропометрических данных необходимо:

- определить контингент людей, для которых предназначено оборудование;
- выбрать группу антропометрических признаков;
- установить процент работающих, которому должно удовлетворять оборудование;
- определить границы интервала размеров (усилий), которые должны быть реализованы в оборудовании.

При проектировании используют антропометрические размеры тела, причем учитываются различия в размерах тела мужчин и женщин, национальные, возрастные, профессиональные. Для определения границ интервалов, в которых учитывается процент населения, используется система перцентелей. Конструкция оборудования должна обеспечивать возможность использования по меньшей мере для 90% потребителей.

Для работы в положении "сидя" используются различные рабочие сиденья. Различают рабочие сиденья для длительного и кратковременного пользования. Общие требования для сидений длительного пользования следующие: сидение должно обеспечивать позу, уменьшающую статистическую работу мышц; создавать условия для возможности изменения рабочей позы; не затруднять деятельность систем организма; обеспечивать свободное перемещение относительно

рабочей поверхности, иметь регулируемые параметры; иметь полумягкую обивку. Для кратковременного пользования рекомендуются жесткие стулья и различного типа табуреты.

В условиях растущей механизации и автоматизации производственных процессов особое значение приобретают средства отображения информации об объекте управления. Широкое использование получила информационная модель, то есть организованная по определенным правилам информация о состоянии объекта управления. К информационным моделям предъявляются следующие требования:

- содержание информационной модели должно адекватно отображать объект управления;
- информационная модель должна обеспечивать оптимальный информационный баланс;
- форма и композиция информационной модели должна соответствовать задачам трудового процесса и возможностям человека по приему информации.

Практика позволяет наметить последовательность разработки информационной модели: определение задач системы, очередность их решения и источников информации; составление перечня объектов управления и их признаков; распределение объектов по степени важности; распределение функции между автоматикой и человеком; выбор системы кодирования объектов и составление общей композиции модели; определение исполнительных действий человека.

В процессе конструирования информационной модели определяются места размещения средств информации на рабочем месте, выбираются размеры знаков и компоновка. Средства отображения размещаются в поле зрения наблюдателя с учетом оптимальных углов и зон наблюдения. Размеры знаков наблюдения определяются с учетом максимальной точности и скорости восприятия информации, а также яркости знаков, величины контраста, использования цвета. Оптимальной

яркостью считаются значения, при которых обеспечивается максимальная контрастная чувствительность. Величина ее будет тем больше, чем меньше размер объекта различения. Оптимальная зона величины контраста равна 60-90%.

В работе глаза имеет место определенная инерционность, что требует учета времени экспозиции зрительного сигнала и временных интервалов для ощущения раздельности сигналов следующих один за другим. В большинстве случаев время экспозиции сигнала должно быть не менее 50 мс. Каждая разновидность индикаторов имеет свою область использования: индикаторы с подсветкой применяются для отображения качественной информации, требующей немедленной реакции оператора; стрелочные индикаторы используются для чтения измеряемых параметров; интегральные индикаторы для совмещения информации сразу о нескольких параметрах.

Структуру и динамику управляемого объекта обычно представляют с помощью микросхемы. В ряде случаев используется табло для отображения информации и восприятия ее коллективом операторов.

При проектировании рабочего места должны учитываться правила экономики движений: при работе двумя руками движения их должны быть одновременными и симметричными; движения должны быть плавными и закругленными, ритмичными и привычными для работающего. Конструкция оборудования должна учитывать правила, касающиеся скорости и точности рабочих движений. Например, наиболее быстрое движение к себе; в горизонтальной плоскости скорость рук больше, чем в вертикальной; точность движений лучше в положении сидя, чем стоя и т.д. Органы управления, используемые на рабочем месте, должны соответствовать общим требованиям эргономики: направление движения органов управления должно соответствовать движению связанного с ним индикатора; соответствие расположения органов управления последовательности работы оператора; удобство использования; создание в

органах правления механического сопротивления и т.п. Помимо этого, к каждому виду органов давления соответствует своя область использования и особые требования к размерам, форме, усилию и т.п.

На автоматизированном рабочем месте оператора-связиста (оператор в диспетчерской) в общем случае используются:

- средства отображения информации индивидуального пользования (блоки отображения, устройства сигнализации и так далее);
- средства управления и ввода информации (пульт дисплея, клавиатура управления, отдельные органы управления и так далее);
- устройства связи и передачи информации (модемы, телеграфные и телефонные аппараты):
- устройства документирования и хранения информации (устройства печати, магнитной записи и так далее);
- вспомогательное оборудование (средства оргтехники, хранилища для носителей информации, устройства местного освещения).

На автоматизированном рабочем месте должна быть обеспечена информационная и конструктивная совместимость используемых технических средств, антропометрических и психофизиологических характеристик человека.

При организации рабочего места должны быть учтены не только факторы, отражающие опыт, уровень профессиональной подготовки, индивидуально-личностные свойства операторов-связистов, но и факторы, характеризующие соответствие форм, способов представления и ввода информации психофизиологическим возможностям человека.

При оптимизации процедур взаимодействия операторов-связистов с техническими средствами в условиях автоматизации эргономические факторы выступают в качестве основных, обуславливающих вероятностно-временные характеристики и напряженность работы. Эти факторы являются чувствительными к вариациям индивидуально-личностных свойств оператора.

Рабочая мебель должна быть удобной для выполнения планируемых рабочих операций. Конструкция рабочей мебели: стола, стула имеет огромное значение для создания здоровых условий и высокопроизводительного труда. Рабочая мебель конструируется с учетом антропометрических данных человека, технических, эстетических и экономических факторов.

В комплекте рабочей мебели большое значение имеет конструкция производственного стула, так как от него зависит поза работника, а следовательно, и затрата энергии и степень его утомляемости. Рабочее сиденье должно иметь требуемые размеры, соответствующие антропометрическим данным человека и быть подвижным. Наиболее удобны стулья и кресла с регулируемым наклоном спинки и высотой сиденья. Изменяя высоту сиденья от уровня пола и угол наклона спинки, можно найти положение, наиболее соответствующее трудовому процессу и индивидуальным особенностям работника.

Как правило, все поверхности письменных и рабочих столов должны быть на уровне локтя при рабочем положении человека. При выборе высоты стола необходимо учитывать сидит человек во время работы или стоит.

Неудобная высота стола снижает эффективность работы и вызывает быстрое утомление. Отсутствие достаточного пространства для коленей и ступней вызывает постоянное раздражение работника. Минимальная рабочая высота стола должна быть не менее 725 мм. Как показывает практика, для рабочего среднего роста высота рабочего стола принимается 800 мм. Для работника другого роста можно изменить высоту рабочего стула или положение его подножки так, чтобы расстояние от предмета обработки до глаз рабочего по высоте было равным примерно 450 мм.

Размещение технических средств и кресла оператора в рабочей зоне должно обеспечивать удобный доступ к основным функциональным

узлам и блокам аппаратуры для проведения технической диагностики, профилактического осмотра и ремонта; возможность быстро занимать и покидать рабочую зону; исключение случайного приведения в действие средств управления и ввода информации; удобную рабочую позу и позу отдыха. Кроме того, схема размещения должна удовлетворять требованиям целостности, компактности и технико-эстетической выразительности рабочей позы.

Дисплей должен размещаться на столе или подставке так, чтобы расстояние наблюдения на экране не превышало 700 мм (оптимальное расстояние 450 – 500 мм). Экран дисплея по высоте должен быть расположен так, чтобы угол между центром экрана и горизонтальной линией взгляда составлял 20°. В горизонтальной плоскости угол наблюдения экрана не должен превышать 60°. Пульт дисплея должен быть размещен на столе или подставке так, чтобы высота клавиатуры пульта по отношению к полу составляла 650 - 720 мм. При размещении пульта на стандартном столе высотой 750 мм необходимо использовать кресло с регулируемой высотой сиденья (450 - 380 мм) и подставку для ног.

Документ (бланк) для ввода оператором данных рекомендуется располагать на расстоянии 450 - 500 мм от глаза оператора, преимущественно слева, при этом угол между экраном дисплея и документом в горизонтальной плоскости должен составлять 30-40°. Угол наклона клавиатуры должен быть равен 15°.

Экран дисплея, документы и клавиатура пульта дисплея должны быть расположены так, чтобы перепад яркостей поверхностей, зависящий от их расположения относительно источника света, не превышал 1 : 10 (рекомендуемое значение

1 : 3). При номинальных значениях яркостей изображения на экране 50 - 100 кд/м<sup>2</sup> освещенность документа должна составлять 300 - 500 лк.

Рабочее место следует оборудовать таким образом, чтобы движения работника были бы наиболее рациональные, наименее утомительные.

Устройства документирования и другие, нечасто используемые технические средства, рекомендуется располагать справа от оператора в зоне максимальной досягаемости, а средства связи слева, чтобы освободить правую руку для записей.

#### **4.2. Обеспечение пожарной безопасности**

Пожар – неконтролируемое горение вне специального очага, наносящее материальный ущерб. Возможности создания условий для возникновения пожара или его быстрого развития представляют собой пожарную опасность.

Опасными поражающими факторами пожара являются:

- открытый огонь и искры;
- выделяющееся при горении тепло вызывает повышение температуры окружающей среды, и когда она доходит до критической для окружающих очаг пожара предметов и вещей, загораются и они. Очаг пожара разрастается;
- токсичные продукты горения, дым;
- падающие части строительных конструкций и агрегатов.

Основными факторами взрыва являются:

- воздушная взрывная волна, основным параметром которой является избыточное давление в ее фронте;
- осколочные поля, создаваемые летящими обломками взрывающихся объектов, поражающее действие которых определяется количеством летящих осколков, их кинетической энергией и радиусом разлета.

Согласно стандартам производственное здание относится к категории Г (производства, связанные с обработкой негорючих веществ и материалов в горячем, раскаленном или расплавленном состоянии, сопровождающееся выделением лучистого тепла, искр и пламени). Использование в компьютерах мощных радиоэлементов: понижающие высоковольтные трансформаторы, выпрямляющие диоды, транзисторы, микросхемы, токопроводящие линии работающих при относительно больших токах и напряжениях, предполагает их нагрев до высоких температур. Тепловое

действие электрического тока, проходящего по проводим, при неисправностях или перегрузка электроустановок или аппаратуры может быть причиной пожара.

Основными причинами пожара являются:

- повреждение изоляции проводов;
- попадание на незаизолированные провода токопроводящих предметов;
- воздействие на провода химически активных веществ, паров;
- неправильный монтаж установки прибора;

Защита электрооборудования осуществляется при помощи плавких предохранителей и специальных автоматов, включаемых последовательно в электрическую цепь. При конструировании аппаратуры, где возможно образование искры или электродуги, предусмотрены кожухи или дугогасящие решетки.

На производстве применяются методы и средства пожаротушения: применения углекислотных огнетушителей, так как  $\text{CO}_2$  не портит оборудование и не проводит электрический ток.

В случае пожара имеются эвакуационные пути:

- из помещений первого этажа – наружу непосредственно или через коридор, вестибюль, лестничную клетку;
- из помещений любого этажа – в коридор или проход, ведущий к лестничной клетке, или на лестничную клетку, имеющую выход наружу, отделенной от примыкающих коридоров перегородками с дверьми;
- из помещения – в соседние помещения на том же этаже, обеспеченные выходом непосредственно наружу, через коридор, лестничную клетку или вестибюль.

В производственном здании имеются планы эвакуации, в которых указаны пути эвакуации людей из здания в случаях возгорания

## ЗАКЛЮЧЕНИЕ

Данная специальные обучающие курсы по повышению компьютерной грамотности, разработанная в ходе выполнения данной работы, является актуальным на сегодняшний день, так как многие пользователи нуждаются в автоматизированном простом и быстром средстве обучения. Программа может быть успешно внедрена в учебный процесс, что соответственно повысит качество знаний. С экономической точки зрения внедрения целесообразно.

Программа соответствует все требованиям, предъявленным к ней. В нее входит: лекционный материал по дисциплине «Компьютерной грамотности», тестовые задания по темам лекций, результаты тестирования записываются в файл, который потом можно посмотреть в программе.

Разработанный учебник устойчиво выполняет все свои функции, что делает его применимым в процессе обучения. Но теперь стоит задача сделать учебник еще более совершенным и более расширенным.

## Список литературы

1. [www.256.ru/publish/elec-book.php](http://www.256.ru/publish/elec-book.php) Компьютерные технологии в обучении. Публикации. Электронный учебник. Тыщенко О.Б. Новое средство компьютерного обучения - электронный учебник // Компьютеры в учебном процессе, 1999, №10, стр.89-92.
2. Журнал "Компьютерные учебные программы".-2001- №2(25).М.:ИНИНФО.-72с. Стратегия информатизации образования в России. Д.Д.Аветисян. стр.47-50. CD-ROM издательство "Мультимедиа Технологии и Дистанционное Обучение". стр.45-46. Информация о регистрируемых программных средствах. Мультимедиа. стр.51-67.
3. Журнал СИР 09/2001. Новинки российского рынка CD. Стр.153-157.
4. <http://molod.mephi.ru/2002/Data/265.htm> Д.В.Гуров, В.В.Гуров. Электронный урок "Представление логических функций". Московский инженерно-физический институт (государственный университет).
5. <http://www.nsu.ru/archive/conf/nit/96/sect2/node4.html#SECTION00040000000000000000> Комплекс программ по математической логике. В.Я.Беляев, Д.А.Печкин, В.Н.Ремесленников. Омский государственный университет.
6. Автоматизированные обучающие системы [Электронный ресурс] URL: [http://www.tspu.tula.ru/ivt/old\\_site/umr/nit/lect/lect4.htm](http://www.tspu.tula.ru/ivt/old_site/umr/nit/lect/lect4.htm) (Дата обращения: 14.11.2010).
7. Википедия. Свободная энциклопедия. Официальный сайт. URL <http://ru.wikipedia.org/wiki/Алгоритм>, (Дата обращения: 14.11.2010).
8. Горюнова М.А., Горюхова Т.В., Кондратьева И.Н., Рубашкин Д.Д. Электронные образовательные издания. Учебно-методическое пособие. СПб.: ЛОИРО, 2003.
9. Интернет университет информационных технологий: Официальный сайт. URL: <http://www.intuit.ru> (Дата обращения: 14.11.2010).
10. Могилёв А. В. Информатика. – М.: «Академия», 1999.

11. Пак Н.И. Нелинейные технологии обучения в условиях информатизации: Учебное пособие. Красноярск: РИО КГПУ, 2004
12. Олейников Б.В. Проблемы и особенности преподавания информатики и программирования в вузе с учетом современных требований. Красноярский государственный университет, 1997. URL: <http://www.nsu.ru/archive/conf/nit/97/c5/node12.html> (Дата обращения: 14.11.2010).
13. Федеральный интернет-экзамен в сфере профессионального образования: Официальный сайт. URL: <http://www.fepo.ru> (Дата обращения: 14.11.2010).
14. Чистохвалов В.Н. Проблемы организации учебного процесса в вузе в системе зачетных единиц. Российский университет дружбы народов (РУДН), 2006. URL: <http://www.rudn.ru/?pagec=293> (Дата обращения: 14.11.2010).
15. Шилдт Г. С# Учебный курс. Программирование. СПб.: Питер, 2003.
16. Электронный словарь: Официальный сайт. URL: [http://chtotakoe.info/articles/elektronnyj\\_uchebnik\\_858.html](http://chtotakoe.info/articles/elektronnyj_uchebnik_858.html) (Дата обращения: 14.11.2010)
1. Экология и безопасность жизнедеятельности. Учебник для ВУЗов. Муравий А. 2002.
2. С.В. Белов. Безопасность жизнедеятельности. Высшая школа, М.: 2003.

## Приложение

### Приложение 1

#### Код формы с лекциями

```
namespace Kursach
{
    public enum Modes { Lecture, ChooseLecture }
    public partial class LecturesForm : Form
    {
        private ChoiseButtons buttons;
        private DirectoryInfo[] lecturesDirectories;
        private DirectoryInfo currentLectureDirectory;
        private Point startPosition = new Point(5, 15);
        private Size buttonSize = new Size(200, 30);
        private int distance = 5;
        private Modes currentMode;
        public LecturesForm(string folderPath)
        {
            InitializeComponent();
            buttons = new ChoiseButtons(ButtonTypes.RadioButton, startPosition.X,
startPosition.Y, buttonSize.Width, buttonSize.Height, distance);
            DirectoryInfo directory = new DirectoryInfo(folderPath);
            lecturesDirectories = directory.GetDirectories();
            foreach (DirectoryInfo lectureDirectory in lecturesDirectories)
            {
                groupBoxLectures.Controls.Add(buttons.AddNextButton(lectureDirectory.
Name));
            }
            SetChooseLectureMode();
        }
    }
}
```

```

private void button1_Click(object sender, EventArgs e)
{
    if (currentMode == Modes.ChooseLecture)
    {
        List<int> selected = buttons.GetSelected();
        if (selected.Count == 0)
        {
            MessageBox.Show("Выберите лекцию!", "Ошибка!",
                MessageBoxButtons.OK, MessageBoxIcon.Information);
            return;
        }
        else
        {
            currentLectureDirectory = lecturesDirectories[selected[0]];
            try
            {
                FileStream fileStream = new FileStream(currentLectureDirectory.FullName
                + "\\lecture", FileMode.Open);
                StreamReader streamReader = new StreamReader(fileStream);
                richTextBoxLecture.Text = streamReader.ReadToEnd();
                streamReader.Close();
                fileStream.Close();
            }
            catch
            {
                TestForm.ShowCriticalErrorMessage("Не найден файл с
                лекцией\nПриложение будет закрыто");
            }
        }
        SetLecturesMode();
    }
}

```

```

    }
    else
    {
        InitForm form = new InitForm();
        if (form.ShowDialog() == DialogResult.OK)
        {
            new
                TestForm(currentLectureDirectory.FullName,
currentLectureDirectory.Name, form.UserName, this).Show();
        }
    }
}

private void SetLecturesMode()
{
    currentMode = Modes.Lecture;
    buttonChoose.Text = "Тест";
    richTextBoxLecture.Visible = true;
    linkLabelResults.Visible = false;
    groupBoxLectures.Visible = false;
    richTextBoxLecture.Dock = DockStyle.Fill;
    linkLabelBack.Visible = true;
    this.MaximumSize = new Size();
    this.MinimumSize = new Size();
    this.Size = new Size(739, 419);
    this.MaximizeBox = true;
    this.Text = currentLectureDirectory.Name;
    buttonChoose.Select();
}

private void SetChooseLectureMode()
{
    currentMode = Modes.ChooseLecture;

```

```

linkLabelResults.Visible = true;
buttonChoose.Text = "Выбрать";
richTextBoxLecture.Visible = false;
groupBoxLectures.Visible = true;
linkLabelBack.Visible = false;
this.Text = "Выбор лекции";
this.MaximumSize = new Size(739, 419);
this.MinimumSize = new Size(739, 419);
this.Size = new Size(739, 419);
this.MaximizeBox = false;
}
private void linkLabel1_LinkClicked(object sender,
LinkLabelLinkClickedEventArgs e)
{
SetChooseLectureMode();
}
private void LecturesForm_VisibleChanged(object sender, EventArgs e)
{
if (Visible == true)
{
SetChooseLectureMode();
}
}
private void linkLabel1_LinkClicked_1(object sender,
LinkLabelLinkClickedEventArgs e)
{
new ResultsForm().ShowDialog();
}
private void panel2_Paint(object sender, PaintEventArgs e)
{

```

```
}  
}  
}
```

### **Код тестовой формы**

```
namespace Kursach  
{  
public partial class TestForm : Form  
{  
private ChoiseButtons buttons;  
private Test test;  
private string userName;  
private List<int> correctAnswers = new List<int>();  
private Point startPosition = new Point(5, 15);  
private Size buttonSize = new Size(200, 30);  
private int distance = 5;  
private Form parentForm;  
public TestForm(string path, string testName, string userName, Form  
formToClose)  
{  
parentForm = formToClose;  
formToClose.Hide();  
InitializeComponent();  
this.userName = userName;  
try  
{  
test = new Test(testName, path);  
}  
catch (Exception)  
{
```

```

        ShowCriticalErrorMessage("Один из файлов тестов имеет неверный
формат\nПриложение будет закрыто");
    }
    test.BeginTest();
    Question question = test.CurrentQuestion;
    buttons = new ChoiseButtons(ButtonTypes.CheckboxButton,
startPosition.X, startPosition.Y, buttonSize.Width, buttonSize.Height, distance);
    InitQuestionWindow(question);
    questionsGroupBox.Select();
}
private void InitQuestionWindow(Question question)
{
    ButtonTypes type = (question.CorrectAnswers.Count != 1) ?
ButtonTypes.CheckboxButton : ButtonTypes.RadioButton;
    buttons.Reset(type);
    questionsGroupBox.Controls.Clear();
    this.Text = question.QuestionName;
    this.textBoxTest.Text = question.QuestionText;
    for (int i = 0; i < question.Answers.Count; i++)
    {
        buttons.AddNextButton(question.Answers[i + 1]);
        questionsGroupBox.Controls.Add(buttons[i]);
    }
}
public static void ShowCriticalErrorMessage(string message)
{
    ShowErrorMessage(message, "Критическая ошибка");
    Environment.Exit(1);
}
private static void ShowInfoMessage(string info, string message)

```

```

    {
        ShowMessage(message,          info,          MessageBoxButtons.OK,
MessageBoxIcon.Information);
    }

    private static void ShowErrorMessage(string info, string message)
    {
        ShowMessage(message,          info,          MessageBoxButtons.OK,
MessageBoxIcon.Error);
    }

    private static void ShowMessage(string info, string message,
MessageBoxButtons buttons, MessageBoxIcon icon)
    {
        MessageBox.Show(message, info, buttons, icon);
    }

    private void buttonNextQuestion_Click(object sender, EventArgs e)
    {
        List<int> selected = buttons.GetSelected();
        bool isCorrect = false;
        if (selected.Count == test.CurrentQuestion.CorrectAnswers.Count)
        {
            isCorrect = true;
            foreach (int correctQuestionNumber in
test.CurrentQuestion.CorrectAnswers)
            {
                if (!selected.Contains(correctQuestionNumber - 1))
                {
                    isCorrect = false;
                }
            }
        }
    }
}

```

```

if (isCorrect) correctAnswers.Add(test.CurrentQuestion.QuestionNumber);
if (!test.IsTestEnded)
{
InitQuestionWindow(test.GoToNextQuestion());
}
else
{
EndTest();
}
}
private void EndTest()
{
FileStream testFileStream = new FileStream("data\\results.txt",
FileMode.Append);
StreamWriter testStreamWriter = new StreamWriter(testFileStream);
testStreamWriter.WriteLine(DateTime.Now.ToShortDateString() + " " +
DateTime.Now.ToShortTimeString() + "\t" + test.TestName + "\t" + userName +
": " + correctAnswers.Count + "/" + test.QuestionsCount);
testStreamWriter.Close();
testFileStream.Close();
DialogResult userDecision = MessageBox.Show("Ваш результат - " +
correctAnswers.Count + "/" + test.QuestionsCount + "\nПерейти к выбору
лекций?", "Тест окончен", MessageBoxButtons.YesNo,
MessageBoxIcon.Question);
if (userDecision == System.Windows.Forms.DialogResult.Yes)
{
parentForm.Show();
this.Close();
}
else

```

```

{
Environment.Exit(0);
}
}
private void SetChooseLectureMode()
{
questionsGroupBox.Visible = false;
buttonNextQuestion.Visible = false;
textBoxTest.Visible = false;
}
private void TestForm_FormClosing(object sender, FormClosingEventArgs

```

e)

```

{
if (parentForm.Visible == false)
{
parentForm.Close();
}
}
private void TestForm_Load(object sender, EventArgs e)
{
}
}
}

```

### **Код формы результатов**

```

namespace Kursach
{
public partial class ResultsForm : Form
{
public ResultsForm()
{

```

```

InitializeComponent();

try
{
    FileStream fileStream = new FileStream("data\\results.txt", FileMode.Open);
    StreamReader streamReader = new StreamReader(fileStream);
    textBox1.Text = streamReader.ReadToEnd();
    this.Select();
    streamReader.Close();
    fileStream.Close();
}
catch
{
    textBox1.Text = "Никто не проходил тесты";
}
}

private void textBox1_TextChanged(object sender, EventArgs e)
{
}
}
}

```

### **Код информационной формы**

```

namespace Kursach
{
    public partial class InitForm : Form
    {
        public InitForm()
        {
            InitializeComponent();
        }
        public string UserName

```

```
{
get { return textBoxName.Text + " " + textBoxSurname.Text; }
}
private void buttonBeginTest_Click(object sender, EventArgs e)
{
DialogResult = System.Windows.Forms.DialogResult.OK;
}
private void InitForm_Load(object sender, EventArgs e)
{
}
}
}
```