

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение
высшего профессионального образования
«ЧЕРЕПОВЕЦКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Институт информационных технологий

Кафедра программного обеспечения вычислительной техники
и информационных систем

МЕТОДИКА И ОРГАНИЗАЦИЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ СТУДЕНТОВ

Учебно-методическое пособие

Направление: 230100.62 Информатика и вычислительная техника,
230400.62 Информационные системы и технологии,
230700.62 Прикладная информатика,
231000.62 Программная инженерия,
221000.68 Мехатроника и робототехника,
230100.68 Информатика и вычислительная техника,
231000.68 Программная инженерия,

Специальность: 230105.65 Программное обеспечение вычислительной
техники и автоматизированных систем

Череповец
2012

Рассмотрено на заседании кафедры программного обеспечения вычислительной техники и информационных систем, протокол № 5 от 10.01.12 г.

Одобрено редакционно-издательской комиссией Института информационных технологий ФГБОУ ВПО ЧГУ, протокол № 23 от 22.02.12 г.

Составители: Е.В. Ершов, д-р техн. наук, проф.; Л.Н. Виноградова, канд. техн. наук; В.В. Селивановских, канд. техн. наук, доцент; О.Л. Селяничев, канд. техн. наук, доцент; О.В. Юдина, канд. техн. наук; Е.В. Майтама; Н.Е. Сивков

Рецензенты: *Е.Н. Бирюков*, канд. техн. наук, начальник УИТ ЧГУ; *Ю.В. Веселов*, канд. техн. наук, менеджер департамента по работе с клиентами ООО «Северсталь-инфоком Софт»

Научный редактор: Е.В. Ершов, д-р техн. наук, проф.

© Коллектив авторов, 2012

© ФГБОУ ВПО «Череповецкий государственный университет», 2012

Введение

Самостоятельная работа студентов в вузе является важным видом учебной и научной деятельности студента.

К современному специалисту общество предъявляет достаточно широкий перечень требований, среди которых немаловажное значение имеет наличие у выпускников определенных способностей и умения самостоятельно добывать знания из различных источников, систематизировать полученную информацию, давать оценку конкретной ситуации. Формирование такого умения происходит в течение всего периода обучения через участие студентов в практических занятиях, выполнение контрольных заданий и тестов, написание курсовых и выпускных квалификационных работ. При этом самостоятельная работа студентов играет решающую роль в ходе всего учебного процесса.

Самостоятельная работа – планируемая учебная, учебно-исследовательская, научно-исследовательская работа студентов, выполняемая во внеаудиторное (аудиторное) время по заданию и при методическом руководстве преподавателя, но без его непосредственного участия (при частичном непосредственном участии преподавателя, оставляющем ведущую роль за работой студентов). Данный вид деятельности является обязательным для каждого студента и определяется основной образовательной программой (учебным планом, программами дисциплин, программами практик, итоговой государственной аттестации и т.д.) по направлению подготовки (специальности).

Содержание самостоятельной работы студента определяется конкретными заданиями, которые он должен выполнить по каждой дисциплине учебного плана.

В качестве конкретных видов самостоятельной работы при освоении той или иной дисциплины выступают:

- самостоятельное изучение отдельных тем, модулей, разделов курса, которые не вошли в аудиторные часы;
- самостоятельное выполнение заданий преподавателя или заданий, сформулированных по инициативе самих студентов, связан-

ных с подготовкой к занятиям и более глубоким освоением содержания той или иной дисциплины;

- самостоятельная работа во время практики;

- учебно-исследовательская и научно-исследовательская самостоятельная работа, связанная с овладением исследовательскими умениями, важнейшими из которых являются: теоретический анализ научных текстов, определение методологии исследования.

Конкретные формы самостоятельной работы разнообразны и определяются видом учебной деятельности и поставленными задачами.

Результаты самостоятельной работы студентов, обучающихся по очной форме с полным сроком освоения образовательных программ в соответствии с рейтинговой формой контроля успеваемости, учитываются в рейтинге студента по соответствующим дисциплинам.

Данное учебно-методическое пособие содержит особенности организации самостоятельной работы и методические рекомендации студентам кафедры ПО ВТ и ИС по отдельным формам самостоятельной работы.

Раздел 1

**ПОДГОТОВКА К ЛЕКЦИЯМ
И СЕМИНАРСКИМ ЗАНЯТИЯМ**

Лекция является ведущей формой организации учебного процесса в вузе. Основными организационными вопросами при этом являются:

- подготовка к восприятию лекции;
- запись лекций.

Особое значение лекции состоит в том, что она знакомит студента с наукой, расширяет, углубляет и совершенствует ранее полученные знания, формирует научное мировоззрение, учит методике и технике лекционной работы. Преподаватель в процессе изложения курса умело связывает теоретические положения своей науки с практикой. Вместе с тем на лекции мобилизуется внимание, вырабатываются навыки слушания, восприятия, осмысления и записывания информации.

Лекция не только дает студенту глубокие, всесторонние сведения по предмету, но и направляет на дальнейшее самостоятельное, более углубленное, изучение материала. Это основа самостоятельной учебной работы студента и получения им знаний. Следовательно, студенту необходимо в совершенстве овладеть умениями и навыками самостоятельной работы над лекцией, которая включает в себя: подготовку к восприятию ее содержания, непосредственное восприятие и осмысливание знаний, записывание их, последующую работу над материалом лекций.

Подготовка к лекции включает в себя проработку материала предыдущей лекции: чтение конспекта, просмотр дополнительных источников, устранение возникших пробелов. При подготовке также рекомендуется кратко ознакомиться по учебной литературе с содержанием следующей лекции. Немаловажную роль играет психологический настрой на работу на лекции, готовность слушать, воспринимать, «пропускать через себя», анализировать, записывать и воспринимать информацию. Только при выполнении этих пунктов студента ждет высокий уровень усвоения знаний и успех в его учебной деятельности.

Запись лекции

Прежде всего, необходимо записать название темы, план лекции, рекомендованную литературу. После этого можно приступать к записи содержания лекции. Придерживайтесь следующих рекомендаций:

- оставляйте поля для различных пометок, замечаний, вопросов;
- не старайтесь дословно конспектировать лекции, выделяйте основные положения, старайтесь понять логику лектора;
- точно записывайте определения, законы, понятия, формулы, теоремы и т.д.;
- передавайте излагаемый лектором материал своими словами;
- наиболее важные положения лекции выделяйте подчеркиванием;
- создайте свою систему сокращения слов.

Форма записи очень индивидуальна, поэтому трудно пользоваться чужими лекциями.

Слушание и конспектирование лекций – это не единственный метод изучения материала дисциплин учебного плана. Важную роль в обучении играет умение студентов работать с различными источниками: книгами, журналами и сетью Internet. Лекции не дают полного объема информации по дисциплинам, так как время, отведенное в соответствии с рабочим учебным планом на лекции сильно ограничено. Темпы роста количества информации, необходимой современному специалисту настолько велики, что обучаемому крайне трудно сориентироваться в ее объемах и местах расположения. Возникает необходимость в выработке у студентов способности самим находить нужные для получения знаний сведения. Очень важно развивать умение самостоятельно пополнять свои знания, ориентироваться в стремительном потоке научной информации. Здесь важны следующие формы самостоятельной работы:

- анализ учебных и научных текстов (составление плана, аннотирование, конспектирование, перевод);
- изучение тематических разделов учебников и учебных пособий;
- подготовка докладов, рефератов.

1.1 Работа с учебной литературой

Самостоятельная работа с учебниками и книгами (а также самостоятельное теоретическое исследование проблем, обозначенных преподавателем на лекциях) – это важнейшее условие формирования научного способа познания.

При работе с книгой необходимо подобрать литературу, научиться правильно ее читать, вести записи. Для подбора литературы в библиотеке используются алфавитный и систематический каталоги.

Важно помнить, что рациональные навыки работы с книгой – это всегда большая экономия времени и сил.

Правильный подбор учебников рекомендуется преподавателем, читающим лекционный курс. Необходимая литература может быть также указана в методических разработках по данному курсу.

Изучая материал по учебнику, следует переходить к следующему вопросу только после правильного уяснения предыдущего, описывая на бумаге все выкладки и вычисления (в том числе те, которые в учебнике опущены или на лекции даны для самостоятельного вывода).

Особое внимание следует обратить на определение основных понятий курса. Необходимо подробно разбирать примеры, которые поясняют такие определения, и уметь строить аналогичные примеры самостоятельно. Нужно добиваться точного представления о том, что изучаешь. Полезно составлять опорные конспекты. При изучении материала по учебнику полезно в тетради (на специально отведенных полях) дополнять конспект лекций. Там же следует отмечать вопросы, выделенные студентом для консультации с преподавателем.

Выводы, полученные в результате изучения, рекомендуется в конспекте выделять, чтобы они при повторном чтении записи лучше запоминались.

Опыт показывает, что многим студентам помогает составление листа опорных сигналов, содержащего важнейшие и наиболее часто употребляемые формулы и понятия. Такой лист помогает запом-

нить формулы, основные положения лекции, а также может служить постоянным справочником для студента.

Различают два вида чтения: первичное и вторичное.

Первичное – это внимательное, неторопливое чтение, при котором можно остановиться на трудных местах. После него не должно остаться ни одного непонятого слова. Содержание не всегда может быть понятно после первичного чтения.

Задача *вторичного* чтения – полное усвоение смысла.

Основные советы по работе с литературой

– Составить перечень книг, с которыми Вам следует познакомиться.

– Сам такой перечень должен быть систематизированным (что необходимо для семинаров, что для экзаменов, что пригодится для написания курсовых и дипломных работ, а что Вас интересует за рамками официальной учебной деятельности, т.е. что может расширить Вашу общую культуру).

– Обязательно выписывать все выходные данные по каждой книге (при написании курсовых и дипломных работ это позволит очень сэкономить время).

– Разобраться для себя, какие книги (или какие главы книг) следует прочитать более внимательно, а какие – просто просмотреть.

– Все прочитанные книги, учебники и статьи следует конспектировать (выписывать кратко основные идеи автора и иногда приводить наиболее яркие и показательные цитаты (с указанием страниц)).

– Если книга – Ваша собственная, то допускается делать на полях книги краткие пометки или же в конце книги, на пустых страницах просто сделать свой «предметный указатель», где отмечаются наиболее интересные для Вас мысли и обязательно указываются страницы в тексте автора.

– Если Вы раньше мало работали с научной литературой, то следует выработать в себе способность «воспринимать» сложные тексты; для этого лучший прием – научиться «читать медленно», когда Вам понятно каждое прочитанное слово (а если слово незна-

комое, то либо с помощью словаря, либо с помощью преподавателя обязательно его узнать), и это может занять немалое время (у кого-то – до нескольких недель и даже месяцев); опыт показывает, что после этого студент каким-то «чудом» начинает буквально заглаживать книги и чуть ли не видеть «сквозь обложку», стоящая это работа или нет.

– Либо читайте, либо перелистывайте материал, но не пытайтесь читать быстро.

– Есть еще один эффективный способ оптимизировать знакомство с научной литературой – следует увлечься какой-то идеей и все книги просматривать с точки зрения данной идеи. В этом случае студент (или молодой ученый) будет как бы искать аргументы «за» или «против» интересующей его идеи, и одновременно он будет как бы общаться с авторами этих книг по поводу своих идей и размышлений. Проблема лишь в том, как найти «свою» идею.

Чтение научного текста является частью познавательной деятельности. Ее цель – извлечение из текста необходимой информации. От того на сколько осознанна читающим собственная внутренняя установка при обращении к печатному слову (найти нужные сведения, усвоить информацию полностью или частично, критически проанализировать материал и т.п.) во многом зависит эффективность осуществляемого действия.

Выделяют *четыре основные установки при чтении научного текста*:

1. Информационно-поисковая (задача – найти, выделить искомую информацию).

2. Усваивающая (усилия читателя направлены на то, чтобы как можно полнее осознать и запомнить как сами сведения излагаемые автором, так и всю логику его рассуждений).

3. Аналитико-критическая (читатель стремится критически осмыслить материал, проанализировав его, определив свое отношение к нему).

4. Творческая (создает у читателя готовность в том или ином виде – как отправной пункт для своих рассуждений, как образ для действия по аналогии и т.п. – использовать суждения автора, ход

его мыслей, результат наблюдения, разработанную методику, дополнить их, подвергнуть новой проверке).

С наличием различных установок обращения к научному тексту связано существование нескольких *видов чтения*:

1. *Библиографическое* – просматривание карточек каталога, рекомендательных списков, сводных списков журналов и статей за год и т.п.;

2. *Просмотровое* – используется для поиска материалов, содержащих нужную информацию, обычно к нему прибегают сразу после работы со списками литературы и каталогами, в результате такого просмотра читатель устанавливает, какие из источников будут использованы в дальнейшей работе;

3. *Ознакомительное* – подразумевает сплошное, достаточно подробное прочтение отобранных статей, глав, отдельных страниц, цель – познакомиться с характером информации, узнать, какие вопросы вынесены автором на рассмотрение, провести сортировку материала;

4. *Изучающее* – предполагает доскональное освоение материала; в ходе такого чтения проявляется доверие читателя к автору, готовность принять изложенную информацию, реализуется установка на предельно полное понимание материала;

5. *Аналитико-критическое и творческое чтение* – два вида чтения близкие между собой тем, что участвуют в решении исследовательских задач. Первый из них предполагает направленный критический анализ, как самой информации, так и способов ее получения и подачи автором; второе – поиск тех суждений, фактов, по которым или в связи с которыми, читатель считает нужным высказать собственные мысли.

Из всех рассмотренных видов чтения основным для студентов является изучающее – именно оно позволяет в работе с учебной литературой накапливать знания в различных областях. Вот почему именно этот вид чтения в рамках учебной деятельности должен быть освоен в первую очередь. Кроме того, при овладении данным видом чтения формируются основные приемы, повышающие эффективность работы с научным текстом.

Основные виды систематизированной записи прочитанного

1. *Аннотирование* – предельно краткое связное описание просмотренной или прочитанной книги (статьи), ее содержания, источников, характера и назначения;

2. *Планирование* – краткая логическая организация текста, раскрывающая содержание и структуру изучаемого материала;

3. *Тезирование* – лаконичное воспроизведение основных утверждений автора без привлечения фактического материала;

4. *Цитирование* – дословное выписывание из текста выдержек, извлечений, наиболее существенно отражающих ту или иную мысль автора;

5. *Конспектирование* – краткое и последовательное изложение содержания прочитанного.

Конспект – сложный способ изложения содержания книги или статьи в логической последовательности. Конспект аккумулирует в себе предыдущие виды записи, позволяет всесторонне охватить содержание книги, статьи. Поэтому умение составлять план, тезисы, делать выписки и другие записи определяет и технологию составления конспекта.

Методические рекомендации по составлению конспекта

1. Внимательно прочитайте текст. Уточните в справочной литературе непонятные слова. При записи не забудьте вынести справочные данные на поля конспекта.

2. Выделите главное, составьте план.

3. Кратко сформулируйте основные положения текста, отметьте аргументацию автора.

4. Законспектируйте материал, четко следуя пунктам плана. При конспектировании старайтесь выразить мысль своими словами. Записи следует вести четко, ясно.

5. Грамотно записывайте цитаты. Цитируя, учитывайте лаконичность, значимость мысли.

В тексте конспекта желательно приводить не только тезисные положения, но и их доказательства. При оформлении конспекта необходимо стремиться к емкости каждого предложения. Мысли автора книги следует излагать кратко, заботясь о стиле и выразительности написанного. Число дополнительных элементов конспекта должно быть логически обоснованным, записи должны распределяться в определенной последовательности, отвечающей логической структуре произведения. Для уточнения и дополнения необходимо оставлять поля.

Овладение навыками конспектирования требует от студента целеустремленности, повседневной самостоятельной работы.

1.2 Подготовка докладов и рефератов

Реферат (от лат. *refero* – сообщаю) – краткое изложение в письменном виде или форме публичного доклада содержания научного труда (трудов), литературы по теме.

Это самостоятельная научно-исследовательская работа студента, где он раскрывает суть исследуемой проблемы, приводит различные точки зрения, а также собственные взгляды на нее. Содержание реферата должно быть логичным. Изложение материала носит проблемно-тематический характер. Тематика рефератов обычно определяется преподавателем, но в определении темы инициативу может проявить и студент.

Этапы работы над рефератом

– *Формулирование темы*, причем она должна быть не только актуальной по своему значению, но и оригинальной, интересной по содержанию. Тема в концентрированном виде выражает содержание будущего текста, фиксируя как предмет исследования, так и его ожидаемый результат.

– *Подбор и изучение основных источников по теме* (как правило, при разработке реферата используется не менее 8-10 различных

источников). На этом этапе необходимо вспомнить, как работать с энциклопедиями и энциклопедическими словарями (обращать особое внимание на список литературы, приведенный в конце тематической статьи); как работать с систематическими и алфавитными каталогами библиотек; как оформлять список литературы (выписывая выходные данные книги и отмечая библиотечный шифр).

– *Составление библиографии.* При написании реферата должна использоваться не только специальная литература, но также и монографии, периодическая литература.

– *Обработка и систематизация информации.* Работу с источниками надо начинать с ознакомительного чтения, т.е. просмотреть текст, выделяя его структурные единицы. При ознакомительном чтении закладками отмечаются те страницы, которые требуют более внимательного изучения. В зависимости от результатов ознакомительного чтения выбирается дальнейший способ работы с источником. Если для разрешения поставленной задачи требуется изучение некоторых фрагментов текста, то используется метод выборочного чтения. Если в книге нет подробного оглавления, следует обратить внимание на предметные и именные указатели. Избранные фрагменты или весь текст (если он целиком имеет отношение к теме) требуют вдумчивого, неторопливого чтения с «мысленной проработкой» материала. Такое чтение предполагает выделение: 1) главного в тексте; 2) основных аргументов; 3) выводов. Особое внимание следует обратить на то, вытекает тезис из аргументов или нет. Необходимо также проанализировать, какие из утверждений автора носят проблематичный, гипотетический характер и уловить скрытые вопросы.

– *Разработка плана реферата.* Универсальный план научного текста, помимо формулировки темы, предполагает изложение вводного материала, основного текста и заключения. Все научные работы – от реферата до докторской диссертации – строятся по этому плану, поэтому важно с самого начала научиться придерживаться данной схемы.

– *Написание реферата.* Раскрытие темы предполагает, что в тексте реферата излагается относящийся к теме материал, и предла-

гаются пути решения содержащейся в теме проблемы; связность текста предполагает смысловую соотносительность отдельных компонентов, а цельность – смысловую законченность текста. С точки зрения связности все тексты делятся на тексты – констатации и тексты – рассуждения. Тексты-констатации содержат результаты ознакомления с предметом и фиксируют устойчивые и несомненные суждения. В текстах-рассуждениях одни мысли извлекаются из других, некоторые ставятся под сомнение, дается им оценка, выдвигаются различные предположения

– *Публичное выступление с результатами исследования.*

Примерная структура реферата

1. *Титульный лист* (заполняется по единой форме, см. приложение 3)

2. *Оглавление* (план, содержание), в котором указаны названия всех разделов (пунктов плана) реферата и номера страниц, указывающие начало этих разделов в тексте реферата.

3. *Введение* – начальная часть текста. Оно имеет своей целью сориентировать читателя в дальнейшем изложении. Во введении аргументируется актуальность исследования, т.е. выявляется практическое и теоретическое значение данного исследования. Далее констатируется, что сделано в данной области предшественниками; перечисляются положения, которые должны быть обоснованы. Введение может также содержать обзор источников или экспериментальных данных, уточнение исходных понятий и терминов, сведения о методах исследования. Во введении обязательно формулируются цель и задачи реферата. Объем введения – в среднем около 10 % от общего объема реферата.

4. *Основная часть* реферата может иметь одну или несколько глав, состоящих из 2 – 3 параграфов (подпунктов, разделов), и предполагает осмысленное и логичное изложение главных положений и идей, содержащихся в изученной литературе. В тексте обязательны ссылки на первоисточники. В том случае, если цитируется или используется чья-либо неординарная мысль, идея, вывод, при-

водится какой-либо цифрой материал, таблица – обязательно приводится ссылка на автора.

5. *Заключение* – последняя часть научного текста. В нем в краткой и сжатой форме излагаются полученные результаты, представляющие собой ответ на главный вопрос исследования. Здесь же могут намечаться и дальнейшие перспективы развития темы. Небольшое по объему сообщение также не может обойтись без заключительной части – пусть это будут две-три фразы. Но в них должен подводиться итог проделанной работы.

6. *Библиография (список литературы)* обязательно сопровождает реферат любого уровня сложности и содержит реально использованную для написания реферата литературу. Список составляется согласно правилам библиографического описания: названия книг в списке располагают по алфавиту с указанием выходных данных.

7. *Приложение* может включать графики, таблицы, блок-схемы алгоритмов, тексты программ, расчеты.

Требования к оформлению реферата

Оформление текстового и графического материала должно соответствовать действующим стандартам. Подробные правила оформления приведены в разделе 10.

Объем реферата может колебаться в пределах 10 – 18 печатных страниц. Все приложения к работе не входят в ее объем.

Реферат должен быть выполнен грамотно, с соблюдением культуры изложения. Обязательно должны иметься ссылки на используемую литературу.

Критерии оценки реферата

1. Знания и умения на уровне требований стандарта конкретной дисциплины: знание фактического материала, усвоение общих представлений, понятий, идей.

2. Характеристика реализации цели и задач исследования (новизна и актуальность поставленных в реферате проблем, правильность формулирования цели, определения задач исследования, пра-

тельность выбора методов решения задач и реализации цели; соответствие выводов решаемым задачам, поставленной цели, убедительность выводов).

3. Степень обоснованности аргументов и обобщений (полнота, глубина, всесторонность раскрытия темы, логичность и последовательность изложения материала, корректность аргументации и системы доказательств, характер и достоверность примеров, иллюстративного материала, широта кругозора автора, наличие знаний интегрированного характера, способность к обобщению).

4. Качество и ценность полученных результатов (степень завершенности реферативного исследования, спорность или однозначность выводов).

5. Использование литературных источников.

6. Культура письменного изложения материала.

7. Культура оформления материалов работы.

По усмотрению преподавателя рефераты могут быть представлены на семинарах, научно-практических конференциях и сопровождаться презентациями.

1.3 Подготовка презентации доклада

Компьютерную презентацию, сопровождающую выступление докладчика, удобнее всего подготовить в программе MS PowerPoint.

Презентация, как документ, представляет собой последовательность сменяющих друг друга слайдов, т.е. электронных страничек, занимающих весь экран монитора (без присутствия панелей программы). Демонстрация презентации, чаще всего, проецируется на большом экране, реже – раздается собравшимся как печатный материал. Количество слайдов адекватно содержанию и продолжительности выступления (например, для 5-минутного выступления рекомендуется использовать не более 10 слайдов).

Перед созданием презентации на компьютере важно определить:

- назначение презентации, ее тему – следует самому понять то, о чем Вы собираетесь рассказывать;

- примерное количество слайдов – слайдов не должно быть много, иначе они будут слишком быстро меняться;
- наиболее удачный способ представления информации;
- содержание слайдов;
- графическое оформление каждого слайда.

Этапы создания презентации

Планирование презентации – определение целей, изучение аудитории, формирование структуры и логики подачи материала.

Составление сценария – логика, содержание.

Разработка дизайна презентации – определение соотношения текстовой и графической информации.

Проверка и отладка презентации.

Требования к оформлению презентаций

1. Требования к содержанию информации:

- на первом слайде обязательно представляется тема выступления и сведения об авторах;
- заголовки должны привлекать внимание аудитории;
- слова и предложения – короткие;
- временная форма глаголов – одинаковая;
- минимум предлогов, наречий, прилагательных.

2. Требования к расположению информации:

- горизонтальное расположение информации;
- наиболее важная информация в центре экрана;
- комментарии к картинке располагаются внизу.

3. Требования к шрифтам:

- размер заголовка не менее 24 пунктов, остальной информации не менее 18 пунктов;
- не более двух – трех типов шрифтов в одной презентации;
- для выделения информации использовать начертание: полужирный шрифт, курсив или подчеркивание.

Необходимо использовать так называемые рубленые шрифты (например, различные варианты Arial или Tahoma), причем размер шрифта должен быть довольно крупный. Предпочтительно не пользоваться курсивом или шрифтами с засечками, так как при этом иногда восприятие текста ухудшается. В некоторых случаях лучше писать большими (заглавными) буквами (тогда можно использовать меньший размер шрифта). Иногда хорошо смотрится жирный шрифт.

Стоит учитывать, что на большом экране текст и рисунки будет видно также (не лучше и не крупнее), чем на экране компьютера. Часто для подписей к рисункам или таблицам выставляется мелкий шрифт (менее 10 пунктов) с оговоркой: «на большом экране все будет видно». Это заблуждение: конечно шрифт будет проецироваться крупнее, но и расстояние до зрителя будет значительно больше.

4. Способы выделения информации:

- рамки, границы, заливка;
- различный цвет шрифта, ячейки, блока;
- рисунки, диаграммы, стрелки, схемы для иллюстрации наиболее важных фактов.

Важно подобрать правильное сочетание цветов для фона и шрифта. Они должны контрастировать, например, фон – светлый, а шрифт – темный, или наоборот. Первый вариант предпочтительнее, так как текст читается лучше. Черный текст – белый фон не всегда можно назвать удачным сочетанием для презентаций, так как при этом глаза устают (особенно если шрифт мелкий), кроме того, иногда не достигается тот визуальный эффект, который необходим для эффективного восприятия материала. Использование фотографий в качестве фона также не всегда удачно, из-за трудностей с подбором шрифта. В этом случае надо либо использовать однотонные чуть размытые фотографии, либо располагать текст не на самой фотографии, а на цветной подложке. Иногда целесообразно использование «тематического» фона: сочетания цветов, несущих смысловую нагрузку.

5. Объем информации и требования к содержанию:

- на одном слайде не более трех фактов, выводов, определений;
- ключевые пункты отражаются по одному на каждом отдельном слайде.

Слайды не надо перегружать ни текстом, ни картинками. Лучше избегать дословного «перепечатывания» текста на слайды – слайды, перегруженные текстом, вообще не смотрятся. Лучше не располагать на одном слайде более 2 – 3 рисунков, так как иначе внимание слушателей будет рассеиваться.

Не стоит вставлять в презентации большие таблицы – они трудны для восприятия – лучше заменять их графиками, построенными на основе этих таблиц. Если все же таблицу показать необходимо, то лучше оставить как можно меньше строк и столбцов, привести только самые необходимые данные. Это также позволит сохранить необходимый размер шрифта. При той легкости, с которой презентации позволяют показывать иллюстративный материал, конечно же, хочется продемонстрировать как можно больше картинок. Однако не стоит злоупотреблять этим.

В презентации не стоит использовать музыкальное сопровождение, если конечно оно не несет смысловую нагрузку, так как музыка будет сильно отвлекать и рассеивать внимание – трудно одновременно слушать лектора и музыку.

В принципе тоже относится и к анимационным эффектам: они не должны использоваться как самоцель. Не стоит думать, что чем больше различных эффектов – тем лучше. Чаще всего неудобочитаемые, быстро появляющиеся и сразу исчезающие надписи не вызывают ничего, кроме раздражения. Анимация допустима либо для создания определенного настроения или атмосферы презентации (в этом случае анимация тем более должна быть сдержанна и хорошо продумана), либо для демонстрации динамичных процессов, изобразить которые иначе просто невозможно (например, для поэтапного вывода на экран рисунка).

Если презентация предназначена только для показа (не для печати), то целесообразно «сжимать» картинки до экранного разрешения (76 точек на дюйм), а также использовать рисунки в формате

«jprg». Это уменьшит объем презентации и значительно ускорит и упростит работу.

Диаграммы готовятся с использованием мастера диаграмм табличного процессора MS Excel. Для ввода числовых данных используется числовой формат с разделителем групп разрядов. Если данные (подписи данных) являются дробными числами, то число отображаемых десятичных знаков должно быть одинаково для всей группы этих данных (всего ряда подписей данных). Данные и подписи не должны накладываться друг на друга и сливаться с графическими элементами диаграммы. Структурные диаграммы готовятся при помощи стандартных средств рисования пакета MS Office. Если при форматировании слайда есть необходимость пропорционально уменьшить размер диаграммы, то размер шрифтов реквизитов должен быть увеличен с таким расчетом, чтобы реальное отображение объектов диаграммы соответствовало значениям, указанным в таблице. В таблицах не должно быть более 4 строк и 4 столбцов – в противном случае данные в таблице будут просто невозможно увидеть. Ячейки с названиями строк и столбцов и наиболее значимые данные рекомендуется выделять цветом.

Табличная информация вставляется в материалы как таблица текстового процессора MS Word или табличного процессора MS Excel. При вставке таблицы как объекта и пропорциональном изменении ее размера реальный отображаемый размер шрифта должен быть не менее 18 пунктов. Таблицы и диаграммы размещаются на светлом или белом фоне.

Заключительный слайд презентации, содержащий текст «Спасибо за внимание» или «Конец», вряд ли приемлем для презентации, сопровождающей публичное выступление, поскольку завершение показа слайдов еще не является завершением выступления. Кроме того, такие слайды, так же как и слайд «Вопросы?», дублируют устное сообщение. Оптимальным вариантом представляется **повторение первого слайда в конце** презентации, поскольку это дает возможность еще раз напомнить слушателям тему выступления и имя докладчика и либо перейти к вопросам, либо завершить выступление.

Для показа файл презентации необходимо сохранить в формате «Демонстрация PowerPoint» (**Файл – Сохранить как – Тип файла – Демонстрация PowerPoint**). В этом случае презентация автоматически открывается в режиме полноэкранного показа (slideshow) и слушатели избавлены как от вида рабочего окна программы PowerPoint, так и от потерь времени в начале показа презентации.

После подготовки презентации необходима репетиция выступления.

Критерии оценки выступления докладчика

Критерии оценки	Содержание оценки
1. Содержательный критерий	Правильный выбор темы, знание предмета и свободное владение текстом, грамотное использование научной терминологии, импровизация, речевой этикет
2. Логический критерий	Стройное логико-композиционное построение речи, доказательность, аргументированность
3. Речевой критерий	Использование языковых (метафоры, фразеологизмы, пословицы, поговорки и т.д.) и неязыковых (поза, манеры и пр.), средств выразительности: фонетическая организация речи, правильность ударения, четкая дикция, логические ударения и пр.
4. Психологический критерий	Взаимодействие с аудиторией (прямая и обратная связь), знание и учет законов восприятия речи, использование различных приемов привлечения и активизации внимания
5. Критерий соблюдения дизайн-эргономических требований к компьютерной презентации	Соблюдены требования к первому и последним слайдам, прослеживается обоснованная последовательность слайдов и информации на слайдах, необходимое и достаточное количество фото- и видеоматериалов, учет особенностей восприятия графической (иллюстративной) информации, корректное сочетание фона и графики, дизайн презентации не противоречит ее содержанию, грамотное соотнесение устного выступления и компьютерного сопровождения, общее впечатление от мультимедийной презентации

1.4 Рекомендации по использованию поисковых систем

С каждым годом объемы интернета увеличиваются в разы, поэтому вероятность найти необходимую информацию резко возрастает. Интернет объединяет миллионы компьютеров, множество разных сетей, число его пользователей быстро увеличивается. И, тем не менее, все чаще при обращении к интернету основной проблемой оказывается не отсутствие искомой информации, а возможность ее найти. Как правило, обычный человек в силу разных обстоятельств не может, или не хочет тратить на поиск нужного ему ответа больше 15 – 20 минут. Поэтому особенно актуально правильно и грамотно научиться, казалось бы, простой вещи – где и как искать, чтобы получать желаемые ответы.

Чтобы найти нужную информацию, необходимо найти ее адрес. Для этого существуют специализированные **поисковые инструменты**:

- Поисковые машины (search engines)
- Каталоги (directories)
- Подборки ссылок
- Базы данных адресов (addresses database)
- Поиск в архивах Gopher (Gopher archives)
- Система поиска FTP файлов (FTP Search)
- Система поиска в конференциях Usenet News
- Системы мета-поиска
- Системы поиска людей

Популярнее всего среди пользователей интернета поисковые машины или другими словами поисковые системы.

Говоря о системах поиска информации, употребляют термины запрос и объект запроса.

Запрос – это формализованный способ выражения информационных потребностей пользователем системы. Для выражения информационной потребности используется язык поисковых запросов, синтаксис которого варьируется от системы к системе. Кроме специального языка запросов, современные поисковые системы позволяют вводить запрос на естественном языке.

Объект запроса – это информационная сущность, которая хранится в базе автоматизированной системы поиска. Несмотря на то, что наиболее распространенным объектом запроса является текстовый документ, не существует никаких принципиальных ограничений. В частности, возможен поиск изображений, музыки и другой мультимедиа информации.

Процесс занесения объектов поиска в информационно-поисковые системы называется *индексацией*.

Правильный запрос состоит из нескольких слов, потому что по одному слову обычно трудно понять, что вы хотите найти. Например, запрос «дизайн» может означать желание найти информацию о дизайне интерьера, или о веб-дизайне, а может – о ландшафтном дизайне. По такому запросу поисковая система найдет страницы обо всех известных ему видах дизайна.

Сейчас поисковые системы хорошо понимают как односложные запросы, так и запросы из нескольких слов и способны самостоятельно их интерпретировать. Они учитывают морфологию языка, знают, в каких случаях надо искать страницы, в которых слова запроса расположены рядом, а когда достаточно словам встретиться в одном документе, чтобы страница хорошо соответствовала запросу.

Большинство поисковых инструментов предлагают два способа поиска – *simple search* (**простой поиск**) и *advanced search* (**расширенный поиск**) с использованием специальной формы запроса и без нее.

В первом случае поисковая система сама интерпретирует запрос и отображает все документы, в которых встречается заданная последовательность слов. Можно также руководить действиями поисковой системы. Для этого и существуют расширенный поиск и **язык запросов**, который состоит из команд-операторов. С помощью языка запросов можно дать поиску дополнительные указания, что и где нужно искать. Освоение критериев уточнения запроса и приемов расширенного поиска, позволяет увеличивать эффективность поиска и достаточно быстро найти необходимую информацию. Многие поисковые машины позволяют решать сложные поисковые задачи, не пользуясь языком запросов. Для этого они приме-

няют формы расширенного поиска, где сложные поисковые условия задаются в простой и наглядной форме.

Советы по поиску

Придерживаясь некоторых правил и советов можно сделать поиск более эффективным.

- **Определитесь с инструментами поиска**

Для обнаружения в текстах фрагментов, аналогичных заданному, используются инструменты линейного поиска информации. К таким инструментам относятся прежде всего общедоступные поисковые машины. Для нахождения данных о связях между объектами используйте системы, позволяющие выполнить запрос по связям.

- **Проверяйте орфографию**

Если поиск не нашел ни одного документа, то вы, возможно, допустили орфографическую ошибку в написании слова. Проверьте правильность написания. Если вы использовали при поиске несколько слов, то посмотрите на количество каждого из слов в найденных документах (перед их списком после фразы «Результат поиска»). Какое-то из слов не встречается ни разу? Скорее всего, его вы и написали неверно.

- **Используйте синонимы**

Если список найденных страниц слишком мал или не содержит полезных страниц, попробуйте изменить слово. Например, вместо «рефераты» возможно больше подойдет «курсовые работы» или «сочинения». Попробуйте задать для поиска три-четыре слова-синонима сразу. Для этого перечислите их через вертикальную черту (|). Тогда будут найдены страницы, где встречается хотя бы одно из них. Например, вместо «фотографии» попробуйте «фотографии | фото | фотоснимки».

- **Ищите больше, чем по одному слову**

Слово «психология» или «продукты» дадут при поиске поодиночке большое число бессмысленных ссылок. Добавьте одно или два ключевых слова, связанных с искомой темой. Например, «пси-

хология Юнга» или «продажа и покупка продовольствия». Также рекомендуется сужать область вопроса.

- **Используйте знаки «+» и «-»**

Чтобы исключить документы, где встречается определенное слово, поставьте перед ним знак минуса. И наоборот, чтобы определенное слово обязательно присутствовало в документе, поставьте перед ним плюс. Обратите внимание, что между словом и знаком плюс-минус не должно быть пробела. Например, если вам нужно описание Парижа, а не предложения многочисленных турагентств, имеет смысл задать такой запрос «путеводитель по Парижу – агентство – тур». Плюс стоит использовать в том случае, когда нужно найти так называемые стоп-слова (наиболее частотные слова русского языка, в основном это местоимения, предлоги, частицы). Чтобы найти цитату из Гамлета, надо задать запрос «+быть или +не быть».

- **Искать без морфологии**

Можно указать поисковой системе, чтобы она не перебирала все словоформы слов из запроса при поиске. Например, запрос «!лукоморья» найдет только страницы, цитирующие строчку из стихотворения Пушкина («У лукоморья дуб зеленый»).

- **Поиск картинок и фотографий**

Сейчас популярные поисковые системы умеют искать не только в тексте документа, но и отыскивать картинки по названию файла или подписи. Для этого обычно предусмотрена соответствующая ссылка на форму, в которой можно указать название картинки и/или подпись к картинке.

Часто возникающие вопросы

- **Как найти точную фразу или форму слова?**

С помощью оператора " ". Заключите фразу или слово в кавычки, и поисковая система будет искать веб-страницы, где есть ровно такая фраза (форма слова).

Пример: ["я к вам пишу"]

- **Как найти цитату, в которой пропущено слово?**

Забыли слово в цитате? Возьмите всю цитату в кавычки, а вместо пропущенного слова поставьте звездочку *. Цитата найдется вместе с забытым словом.

Пример: ["ночь улица * аптека"]

- **Как найти любые из нескольких слов?**

Просто перечислите все подходящие варианты через вертикальный слеш: |. Поисковая система будет искать документы с любым из этих слов.

Пример: [ананасы | рябчики | шампанское]

- **Как найти слова в пределах одного предложения?**

Используйте оператор &. Если соединить слова амперсандом, то будут найдены документы, где эти слова стоят в одном предложении.

Пример: [памятник Пушкину & Псков]

- **Как исключить слово из поиска?**

Поставьте минус перед словом, которое вы не хотите видеть в ответах. Так можно исключить даже несколько слов:

Пример: [мумий тролль мультфильм - рок - Лагутенко]

- **Как искать на определенном сайте?**

Для этого подойдет оператор site. Он позволяет прямо в запросе указать сайт, на котором нужно искать. Только обязательно поставьте после site двоеточие.

Пример: [конституция рф site:consultant.ru]

- **Как искать документы определенного типа?**

Для этого подойдет оператор mime. Поставьте в запросе mime, двоеточие и затем тип документа, который вам нужен. Например, pdf или doc.

Пример: [заявление на загранпаспорт mime:pdf]

- **Как искать на сайтах на определенном языке?**

С помощью оператора lang. После lang нужно поставить двоеточие и написать, на каком языке вам нужны документы. Если это русский язык, то нужно указать ru.

Пример: [void glColor lang:ru]

Что такое язык запросов

Язык запросов, позволяет наиболее точно формулировать свой запрос поисковой системе, учитывая малейшие нюансы ее поведения. Наиболее популярные критерии поиска можно задавать с помощью расширенного поиска, знание же языка запросов дает возможность решать самые сложные поисковые задачи.

Памятка по использованию языка запросов

Пример	Значение
1	2
К нам на утренний рассол"	Слова идут подряд в точной форме
"Прибыл * посол"	Пропущено слово в цитате
полгорбушки & мосол	Слова в пределах одного предложения
снаряжайся && добудь	Слова в пределах одного документа
глухаря куропатку кого-нибудь	Поиск любого из слов
не сможешь << винить	Неранжирующее "и": выражение после оператора не влияет на позицию документа в выдаче
я должен /2 казнить	Расстояние в пределах двух слов в любую сторону (т.е. между заданными словами может встречаться одно слово)
государственное дело && /3 улавливаешь нить	Расстояние в 3 предложения в любую сторону
нешто я ~~ пойму	Исключение слова пойму из поиска
при моем /+2 уму	Расстояние в пределах двух слов в прямом порядке
чай ~ лаптем	Поиск предложения, где слово чай встречается без слова лаптем
щи /(-1 +2) хлебаю	Расстояние от одного слова в обратном порядке до двух слов в прямом

1	2
!Соображаю !что !чему	Слова в точной форме с заданным регистром
получается && (+на !мне)	Скобки формируют группы в сложных запросах
!!политика	Словарная форма слова
title:(в стране)	Поиск по заголовкам документов
url:ptici.narod.ru/ptici/kuropatka.htm	Поиск по URL
беспрерывно inurl:vojne	Поиск с учетом фрагмента URL
host:lib.ru	Поиск по хосту
rhost:ru.lib.*	Поиск по хосту в обратной записи
site:http://www.lib.ru/PXESY/FILATOW	Поиск по всем поддоменам и страницам заданного сайта
mime:pdf	Поиск по одному типу файлов
lang:en	Поиск с ограничением по языку
domain:ru	Поиск с ограничением по домену
date:200712*	Поиск с ограничением по дате
date:20071215..20080101, date:>20091231	Поиск с ограничением по интервалу дат

Что такое Каталог

Каталог – это коллекция аннотированных ссылок на сайты, которую вручную собрали редакторы. Цель каталога – предложить пользователю представительную подборку сайтов в каждой рубрике и облегчить поиск информации в интернете. Для этого в каталоге используется не только обычная тематическая рубрикация (Развлечения, Спорт, Работа и пр.), но и типовая (Товары и услуги, Советы и пр.) и географическая.

Аннотации сайтов

Каталог содержит большое количество информации о сайтах, которая позволяет сделать выбор в пользу того или иного ресурса.

Для того чтобы дать полное представление о ресурсе, для каждого сайта может приводиться следующая информация:

- Название и описание сайта – формируются редакторами вручную и дают представление о содержании сайта.
- Скриншот сайта
- Адрес организации
- Регион
- Значимые подразделы сайта. Кроме ссылки на сам ресурс, на странице могут показываться ссылки на другие описанные в Каталоге страницы с этого хоста близкие по тематике к этому сайту.
- Цитируемость – специальный параметр, позволяющий отсортировать сайты в рубрике по их популярности.
- Пиктограмма (favicon) – небольшая картинка, которая показывается рядом с сайтом на страницах каталога.

Поиск по каталогу

В каталоге существует два типа поиска.

1. Поиск по сайтам

Поиск осуществляется по страницам сайтов, находящихся в базе каталога. Туда относятся или опубликованные в каталоге сайты или те, которым присвоены только классифицирующие признаки (рубрика и регион).

2. Поиск по описаниям сайтов

В каталоге можно искать по описаниям сайтов, составленных редакторами. Для этого необходимо установить флажок «поиск по описаниям сайтов» под поисковой строкой. Поскольку заголовки отражают основной профиль сайта, сайты, найденные по заголовку, стоят в результатах поиска выше, чем сайты, найденные по описанию.

Популярные каталоги интернет-ресурсов

- **Глобальные каталоги**

Open Directory Project, также известный как DMOZ.

(<http://www.dmoz.org>)

Каталог Yahoo! (<http://dir.yahoo.com>)

- **Русскоязычные каталоги**

Rambler Top100 (<http://top100.rambler.ru>)

Каталог@mail.ru (<http://list.mail.ru/index.html>)

Яндекс.Каталог (<http://catalog.yandex.ru>, <http://yaca.yandex.ru>)

Aport.Каталог (<http://catalog.aport.ru>,
<http://aport.ru/search/?That=cat>)

Раздел 2

ПОДГОТОВКА К ПРАКТИЧЕСКИМ И ЛАБОРАТОРНЫМ ЗАНЯТИЯМ

Лабораторно-практические занятия сопровождают практически все курсы вузовской подготовки специалиста. Их целью является закрепление лекционного материала, выработка практических умений и навыков.

Для того чтобы практические занятия приносили максимальную пользу, необходимо помнить, что решение упражнений проводится по вычитанному на лекциях материалу и связано, как правило, с детальным разбором отдельных вопросов лекционного курса. Подготовка к занятиям состоит из теоретической подготовки с использованием лекций и методических пособий, и выполнения практических заданий (решения задач, ответов на вопросы и т.д.). Лабораторные работы проводятся в специально оборудованных учебных аудиториях (как правило, в компьютерных классах) и имеют продолжительность не менее 4-х часов. Практические занятия проводятся в учебных аудиториях. Их продолжительность не менее 2-х часов.

Конкретными формами самостоятельной работы при подготовке к лабораторным и практическим занятиям могут быть:

- теоретическая подготовка к выполнению заданий, которые будут на практическом занятии;
- выполнение упражнений, решение задач;
- составление алгоритмов, написание и отладка программ;
- оформление отчетов по лабораторным работам (образец титульного листа приведен в прил. 4);
- ответы на контрольные вопросы;
- выполнение расчетно-графических заданий, домашних лабораторных работ;
- выполнение индивидуальных или коллективных учебных проектов;
- выполнение домашних контрольных работ и тестовых заданий и др.

2.1 Выполнение расчетно-графических заданий

Расчетно-графическое задание (РГЗ) – вид самостоятельной работы студентов, цель которого – развитие профессиональных компетенций и практических навыков через самостоятельное освоение изучаемого материала.

Задание, как правило, выдается каждому обучающемуся индивидуально и включает теоретическую и практическую части.

Теоретическая часть предполагает самостоятельное освоение некоторого теоретического материала, либо расширение знаний, полученных во время аудиторной работы. Ее выполнение способствует формированию компетенций, связанных с развитием аналитических способностей, способности к самостоятельному освоению новых методов, а также навыков поиска необходимой информации.

Практическая часть зависит от специфики курса и может включать выполнение расчетов, составление алгоритмов, написание программ. Ее выполнение формирует практические навыки в профессиональной области, использовании инструментальных средств и технологий, а также представления информации, составление отчетов.

Преподаватель в письменной (электронной) форме формулирует задание. Формулировка включает: задачу, требования к выполнению практической части, содержанию пояснительной записки, критерии оценки, ссылки на теоретический материал, необходимый для решения задачи, либо рекомендации для его поиска.

Задание выдается не позднее, чем за 1,5 месяца до установленного графиком срока сдачи для студентов очной формы обучения и в соответствии с учебным графиком для студентов заочной формы обучения.

Выдача задания, как правило, предполагает индивидуальные пояснения каждому учащемуся в соответствии с его вариантом, а также возможность консультации по выполнению задания.

Результат выполнения РГЗ представляется студентом в форме пояснительной записки.

Пояснительная записка оформляется в соответствии с общими требованиями к оформлению текстовых и графических материалов

согласно ЕСПД. Подробные правила оформления текстовых и графических материалов приведены в разделе 10. Рекомендуемый объем пояснительной записки 17 – 25 страниц без приложений. Она должна включать следующие разделы:

- титульный лист (образец в прил. 3),
- введение,
- глава 1 (теоретическая часть),
- глава 2 (практическое задание),
- заключение,
- список литературы,
- приложения.

Во введении студентом представляется в сжатой форме понимание постановки задачи, план ее решения, возможные подходы к решению. Рекомендуемый объем – 1 – 2 страницы.

Первая глава представляет собой изложение теоретического материала, необходимого для выполнения задания. Это может быть изложение метода, алгоритма, либо подборка материалов, выполненная студентом в соответствии с заданием преподавателя. Теоретический материал должен представляться в форме обзора, содержать выводы и пояснения. Прямое копирование необходимо оформлять как цитаты, со ссылками на первоисточник. Наличие заимствованных включений текстов других авторов, не отмеченных как цитаты, является основанием для отрицательной оценки.

Во второй главе необходимо перейти к практическому выполнению задания. В зависимости от специфики курса это может быть подробное решение с необходимыми пояснениями, описание алгоритма, либо программной реализации с требуемыми схемами, обоснованиями. Если задание предполагает, что учащимся должны быть приведены алгоритмы, диаграммы, схемы и т.д., то они должны быть выполнены в соответствии с требованиями ГОСТ.

В заключении необходимо перечислить полученные результаты выполнения задания, дать ответы на поставленные вопросы, сделать выводы по итогам проведенной работы.

В списке литературы указываются все источники (книги, статьи, интернет-источники), использованные при выполнении зада-

ния. В тексте записки необходимо указать ссылки на каждый из источников.

В приложения выносятся: текст индивидуального задания, тексты программ, объемные схемы и диаграммы, которые необходимы для иллюстрации решения или выполнения задания.

Пояснительная записка проверяется преподавателем в течение 3 – 10 дней, результаты проверки доводятся до сведения студента. В случае отрицательной оценки пояснительная записка возвращается с указанием недочетов (замечаний). Для положительно оцененных работ проводится защита.

Защита РГЗ может проводиться как в форме публичного выступления студента перед аудиторией, так и в форме индивидуальной беседы с преподавателем. Итоговая оценка выставляется преподавателем с учетом качества выполнения задания (решения задачи), представленных письменных материалов и результатов защиты.

2.2 Отладка программного обеспечения

Отладка программы – один из самых сложных этапов разработки программного обеспечения, требующий глубокого знания:

- специфики управления используемыми техническими средствами,
- операционной системы,
- среды и языка программирования,
- реализуемых процессов,
- природы и специфики различных ошибок,
- методик отладки и соответствующих программных средств.

Отладка – это процесс локализации и исправления ошибок, обнаруженных при тестировании программного обеспечения. *Локализацией* называют процесс определения оператора программы, выполнение которого вызвало нарушение нормального вычислительного процесса. Для исправления ошибки необходимо определить ее *причину*, т.е. определить оператор или фрагмент, содержащие ошибку. Причины ошибок могут быть как очевидны, так и очень глубоко скрыты.

В целом, сложность отладки обусловлена следующими причинами:

- требует от программиста глубоких знаний специфики управления используемыми техническими средствами, операционной системы, среды и языка программирования, реализуемых процессов, природы и специфики различных ошибок, методик отладки и соответствующих программных средств;

- психологически дискомфортна, так как необходимо искать собственные ошибки, как правило, в условиях ограниченного времени;

- возможно взаимовлияние ошибок в разных частях программы, например, за счет затирания области памяти одного модуля другим из-за ошибок адресации;

- отсутствуют четко сформулированные методики отладки.

В соответствии с этапом обработки, на котором проявляются ошибки, различают (рис. 2.1):

- *синтаксические ошибки* – ошибки, фиксируемые компилятором (транслятором, интерпретатором) при выполнении синтаксического и, частично, семантического анализа программы;

- *ошибки компоновки* – ошибки, обнаруженные компоновщиком (редактором связей) при объединении модулей программы;

- *ошибки выполнения* – ошибки, обнаруженные операционной системой, аппаратными средствами или пользователем при выполнении программы.

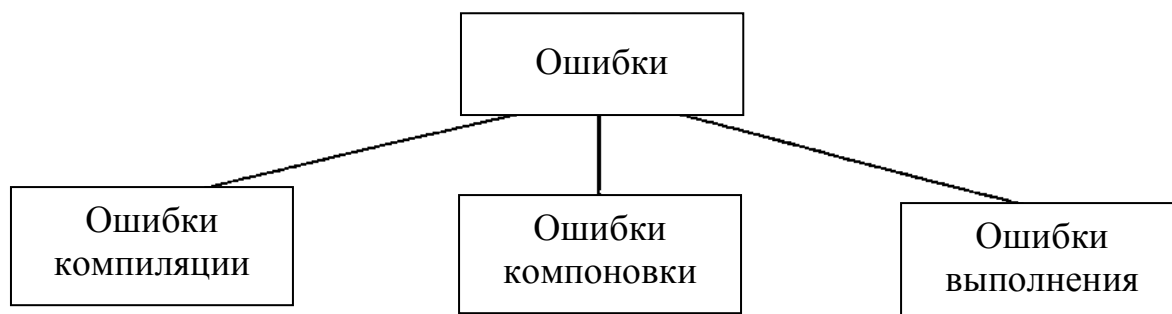


Рис. 2.1. Классификация ошибок по этапу обработки программы

Синтаксические ошибки. Синтаксические ошибки относят к группе самых простых, так как синтаксис языка, как правило, строго формализован, и ошибки сопровождаются развернутым комментарием с указанием их местоположения. Определение причин таких ошибок, как правило, труда не составляет, и даже при нечетком знании правил языка за несколько прогонов удастся удалить все ошибки данного типа.

Следует иметь в виду, что чем лучше формализованы правила синтаксиса языка, тем больше ошибок из общего количества может обнаружить компилятор и, соответственно, меньше ошибок будет обнаруживаться на следующих этапах. В связи с этим, говорят о языках программирования с защищенным синтаксисом и с незащищенным синтаксисом. К первым, безусловно, можно отнести Pascal, имеющий очень простой и четко определенный синтаксис, хорошо проверяемый при компиляции программы, ко вторым – Си со всеми его модификациями. Чего стоит хотя бы возможность выполнения присваивания в условном операторе в Си, например:

$$\text{if } (c = n) \ x = 0;$$

В данном случае не проверятся равенство c и n , а выполняется присваивание c значения n , после чего результат операции сравнивается с нулем, если программист хотел выполнить не присваивание, а сравнение, то эта ошибка будет обнаружена только на этапе выполнения при получении результатов, отличающихся от ожидаемых.

Ошибки компоновки. Ошибки компоновки, как следует из названия, связаны с проблемами, обнаруженными при разрешении внешних ссылок. Например, предусмотрено обращение к подпрограмме другого модуля, а при объединении модулей данная подпрограмма не найдена или не стыкуются списки параметров. В большинстве случаев ошибки такого рода также удастся быстро локализовать и устранить.

Ошибки выполнения. К самой непредсказуемой группе относятся ошибки выполнения. Прежде всего, они могут иметь разную природу, и соответственно по-разному проявляться. Часть ошибок

обнаруживается и документируется операционной системой. Выделяют четыре способа проявления таких ошибок:

- появление сообщения об ошибке, зафиксированной схемами контроля выполнения машинных команд, например, переполнении разрядной сетки, ситуации «деление на ноль», нарушении адресации и т.п.;
- появление сообщения об ошибке, обнаруженной операционной системой, например, нарушении защиты памяти, попытке записи на устройства, защищенные от записи, отсутствии файла с заданным именем и т.п.;
- «зависание» компьютера: как простое, когда удастся завершить программу без перезагрузки операционной системы, так и «тяжелое», когда для продолжения работы необходима перезагрузка;
- несовпадение полученных результатов с ожидаемыми.

Примечание. Отметим, что, если ошибки этапа выполнения обнаруживает пользователь, то в двух первых случаях, получив соответствующее сообщение, пользователь в зависимости от своего характера, степени необходимости и опыта работы за компьютером, либо попытается понять, что произошло, ища свою вину, либо обратится за помощью, либо постарается никогда больше не иметь дела с этим продуктом. При «зависании» компьютера пользователь может даже не сразу понять, что происходит что-то не то, хотя его печальный опыт и заставляет волноваться каждый раз, когда компьютер не выдает быстрой реакции на введенную команду, что также целесообразно иметь в виду. Также опасны могут быть ситуации, при которых пользователь получает неправильные результаты и использует их в своей работе.

Причины ошибок выполнения очень разнообразны, а потому и локализация может оказаться крайне сложной. Все возможные причины ошибок можно разделить на следующие группы:

- неверное определение исходных данных,
- логические ошибки,
- накопление погрешностей результатов вычислений (рис. 2.2).

Неверное определение исходных данных происходит, если возникают любые ошибки при выполнении операций ввода-вывода: ошибки передачи, ошибки преобразования, ошибки перезаписи и

ошибки данных. Причем использование специальных технических средств и программирование с защитой от ошибок позволяет обнаружить и предотвратить только часть этих ошибок, о чем безусловно не следует забывать.



Рис. 2.2. Классификация ошибок этапа выполнения по возможным причинам

Логические ошибки имеют разную природу. Так, они могут следовать из ошибок, допущенных при проектировании, например, при выборе методов, разработке алгоритмов или определении структуры классов, а могут быть непосредственно внесены при кодировании модуля. К последней группе относят:

- *ошибки некорректного использования переменных*, например, неудачный выбор типов данных, использование переменных до их инициализации, использование индексов, выходящих за границы определения массивов, нарушения соответствия типов данных при использовании явного или неявного переопределения типа данных, расположенных в памяти при использовании нетипизированных переменных, открытых массивов, объединений, динамической памяти, адресной арифметики и т.п.;

- *ошибки вычислений*, например, некорректные вычисления над неарифметическими переменными, некорректное использование целочисленной арифметики, некорректное преобразование типов данных в процессе вычислений, ошибки, связанные с незнанием приоритетов выполнения операций для арифметических и логических выражений и т.п.;

- *ошибки межмодульного интерфейса*, например, игнорирование системных соглашений, нарушение типов и последовательности при передаче параметров, несоблюдение единства единиц измерения формальных и фактических параметров, нарушение области действия локальных и глобальных переменных;

- *другие ошибки кодирования*, например, неправильная реализация логики программы при кодировании, игнорирование особенностей или ограничений конкретного языка программирования.

Накопление погрешностей результатов числовых вычислений возникает, например, при некорректном отбрасывании дробных цифр чисел, некорректном использовании приближенных методов вычислений, игнорировании ограничения разрядной сетки представления вещественных чисел в ЭВМ и т.п.

Все указанные выше причины возникновения ошибок следует иметь в виду в процессе отладки. Кроме того, сложность отладки увеличивается также вследствие влияния следующих факторов:

- опосредованного проявления ошибок;
- возможности взаимовлияния ошибок;
- возможности получения внешне одинаковых проявлений разных ошибок;
- отсутствия повторяемости проявлений некоторых ошибок от запуска к запуску – так называемые стохастические ошибки;

- возможности устранения внешних проявлений ошибок в исследуемой ситуации при внесении некоторых изменений в программу, например, при включении в программу диагностических фрагментов может аннулироваться или измениться внешнее проявление ошибок;
- написания отдельных частей программы разными программистами.

Методы отладки программного обеспечения

Отладка программы в любом случае предполагает обдумывание и логическое осмысление всей имеющейся информации об ошибке. Большинство ошибок можно обнаружить по косвенным признакам посредством тщательного анализа текстов программ и результатов тестирования без получения дополнительной информации. При этом используют различные методы:

- ручного тестирования;
- индукции;
- дедукции;
- обратного прослеживания.

Метод ручного тестирования. Это – самый простой и естественный способ данной группы. При обнаружении ошибки необходимо выполнить тестируемую программу вручную, используя тестовый набор, при работе с которым была обнаружена ошибка.

Метод очень эффективен, но не применим для больших программ, программ со сложными вычислениями и в тех случаях, когда ошибка связана с неверным представлением программиста о выполнении некоторых операций.

Данный метод часто используют как составную часть других методов отладки.

Метод индукции. Метод основан на тщательном анализе симптомов ошибки, которые могут проявляться как неверные результаты вычислений или как сообщение об ошибке. Если компьютер просто «зависает», то фрагмент проявления ошибки вычисляют, исходя из последних полученных результатов и действий пользова-

теля. Полученную таким образом информацию организуют и тщательно изучают, просматривая соответствующий фрагмент программы. В результате этих действий выдвигают гипотезы об ошибках, каждую из которых проверяют. Если гипотеза верна, то детализируют информацию об ошибке, иначе – выдвигают другую гипотезу. Последовательность выполнения отладки методом индукции показана на рис. 2.3 в виде схемы алгоритма.

Самый ответственный этап – выявление симптомов ошибки. Организуя данные об ошибке, целесообразно записать все, что известно о ее проявлениях, причем фиксируют, как ситуации, в которых фрагмент с ошибкой выполняется нормально, так и ситуации, в которых ошибка проявляется. Если в результате изучения данных никаких гипотез не появляется, то необходима дополнительная информация об ошибке. Дополнительную информацию можно получить, например, в результате выполнения схожих тестов.

В процессе доказательства пытаются выяснить, все ли проявления ошибки объясняет данная гипотеза, если не все, то либо гипотеза не верна, либо ошибок несколько.

Метод дедукции. По методу дедукции вначале формируют множество причин, которые могли бы вызвать данное проявление ошибки. Затем анализируя причины, исключают те, которые противоречат имеющимся данным. Если все причины исключены, то следует выполнить дополнительное тестирование исследуемого

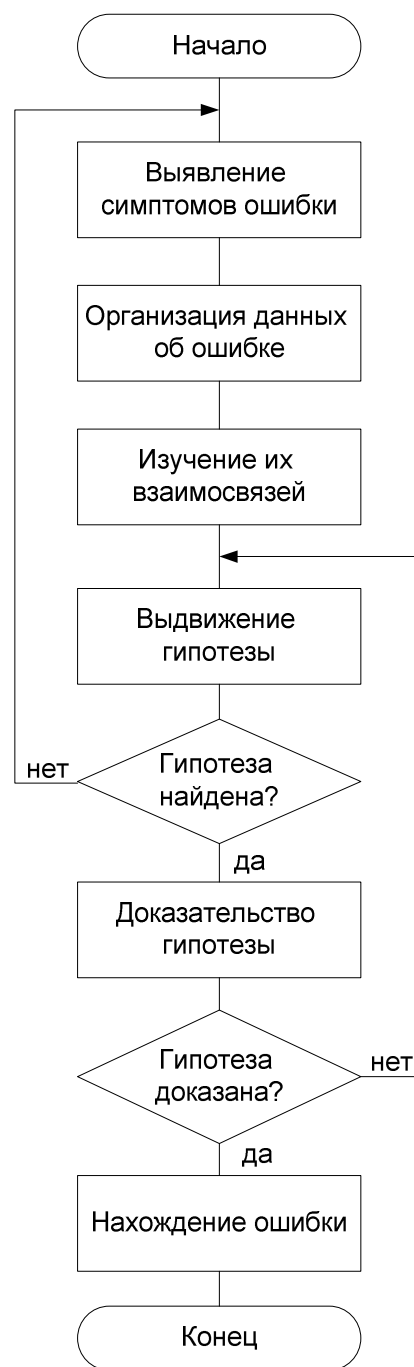


Рис. 2.3. Схема процесса отладки методом индукции

фрагмента. В противном случае наиболее вероятную гипотезу пытаются доказать. Если гипотеза объясняет полученные признаки ошибки, то ошибка найдена, иначе – проверяют следующую причину (рис. 2.4).

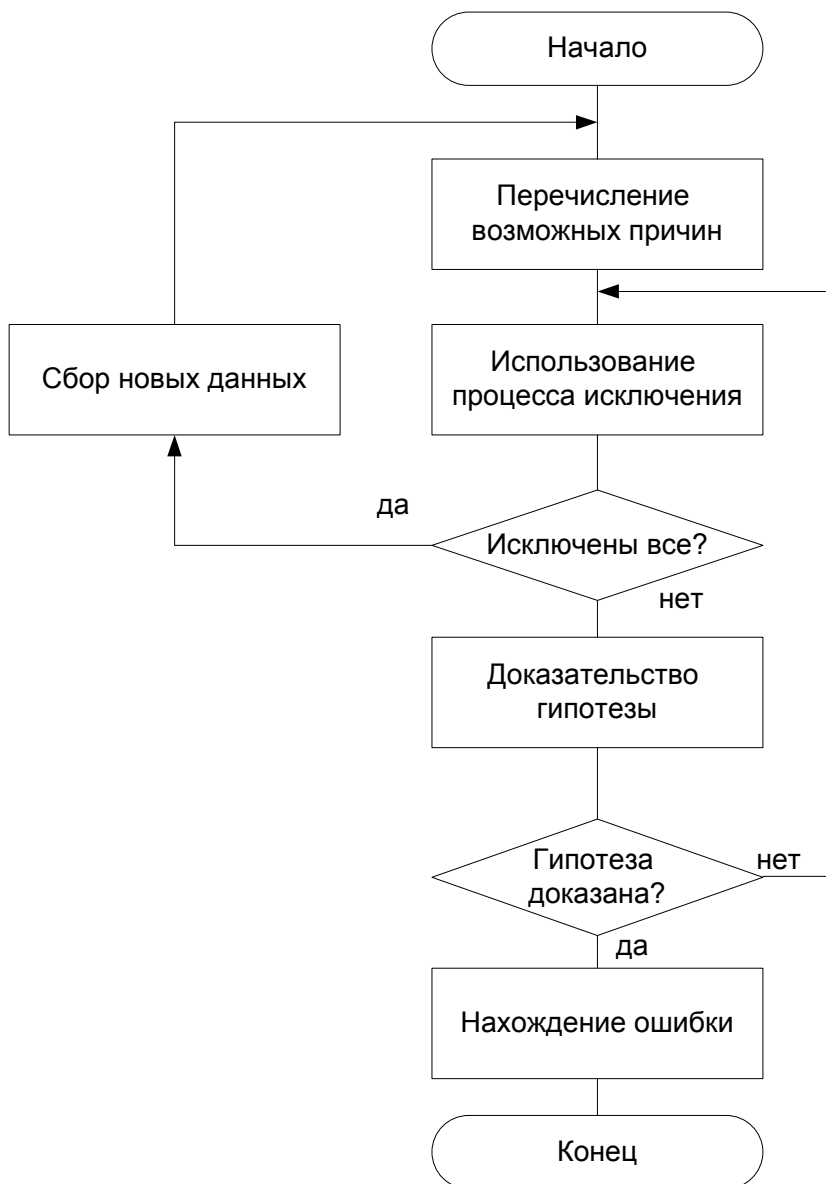


Рис. 2.4. Схема процесса отладки методом дедукции

Метод обратного прослеживания. Для небольших программ эффективно применение метода обратного прослеживания. Начинают с точки вывода неправильного результата. Для этой точки строится гипотеза о значениях основных переменных, которые могли бы привести к получению имеющегося результата. Далее, исходя из этой гипотезы, делают предложения о значениях пере-

менных в предыдущей точке. Процесс продолжают, пока не обнаружат причину ошибки.

Методы и средства получения дополнительной информации

Для получения дополнительной информации об ошибке можно выполнить добавочные тесты или использовать специальные методы и средства:

- отладочный вывод;
- интегрированные средства отладки;
- независимые отладчики.

Отладочный вывод. Метод требует включения в программу дополнительного отладочного вывода в *узловых* точках. Узловыми считают точки алгоритма, в которых основные переменные программы меняют свои значения. Например, отладочный вывод следует предусмотреть до и после завершения цикла изменения некоторого массива значений. (Если отладочный вывод предусмотреть в цикле, то будет выведено слишком много значений, в которых, как правило, сложно разбираться.) При этом предполагается, что, выполнив анализ выведенных значений, программист уточнит момент, когда были получены неправильные значения, и сможет сделать вывод о причине ошибки.

Данный метод не очень эффективен и в настоящее время практически не используется, так как в сложных случаях в процессе отладки может потребоваться вывод большого количества – «трассы» значений многих переменных, которые выводятся при каждом изменении. Кроме того, внесение в программы дополнительных операторов может привести к изменению проявления ошибки, что нежелательно, хотя и позволяет сделать определенный вывод о ее природе.

Примечание. Ошибки, исчезающие при включении в программу или удалении из нее каких-либо «безобидных» операторов, как правило, связаны с «затиранием» памяти. В результате добавления или удаления операторов область затирания может сместиться в другое место и ошибка либо перестанет проявляться, либо будет проявляться по-другому.

Интегрированные средства отладки. Большинство современных сред программирования (Delphi, Builder C++, Visual Studio и т.д.) включают средства отладки, которые обеспечивают максимально эффективную отладку. Они позволяют:

- выполнять программу по шагам, причем как с заходом в подпрограммы, так и выполняя их целиком;
- предусматривать точки останова;
- выполнять программу до оператора, указанного курсором;
- отображать содержимое любых переменных при пошаговом выполнении;
- отслеживать поток сообщений и т. п.

На рис. 2.5 показан вид программы в момент перехода в режим пошагового выполнения по достижении точки останова в Delphi. В этот момент программист имеет возможность просмотреть значения интересующих его переменных.

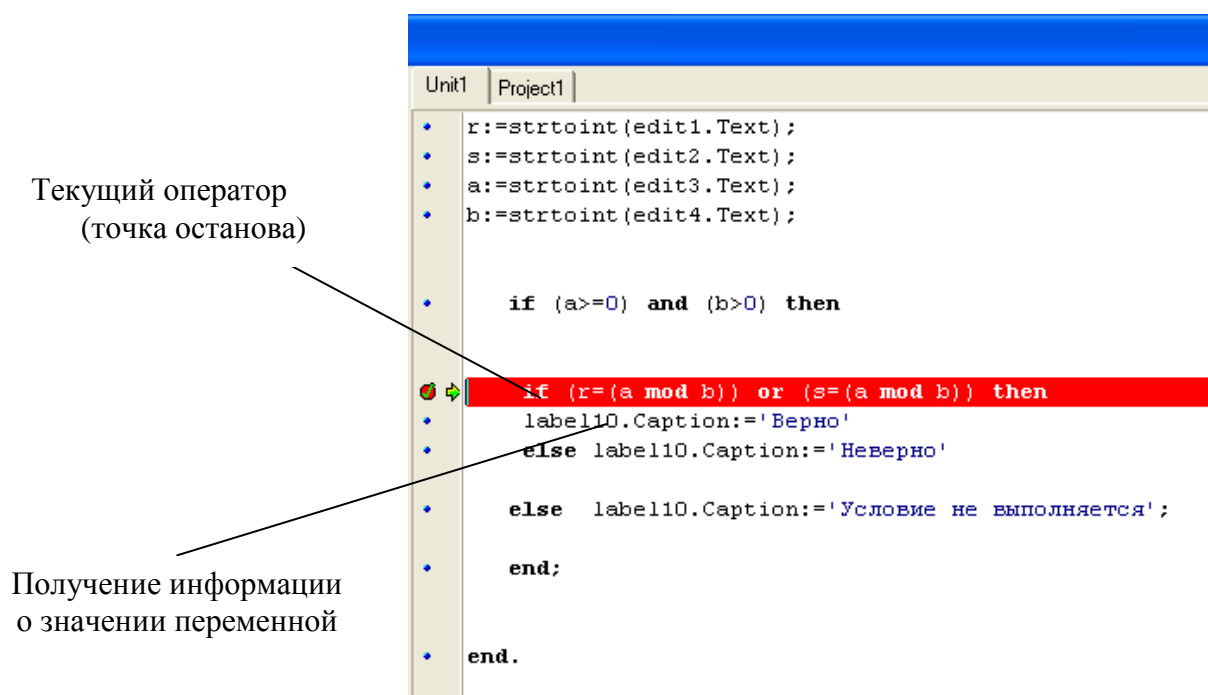


Рис. 2.5. Переход в режим пошагового выполнения по достижении точки останова в Delphi

Если получены неправильные результаты, то локализовать ошибку обычно существенно сложнее. В этом случае сначала определяют фрагмент, при выполнении которого получаются

неправильные результаты. Для этого последовательно проверяют интересные значения в узловых точках. Обнаружив значения, отличающиеся от ожидаемых, по шагам трассируют соответствующий фрагмент до выявления оператора, выполнение которого дает неверный результат.

Для уточнения природы ошибки возможен анализ машинных кодов, флагов и представления программы и значений памяти в 16-ричном виде (рис. 2.6).

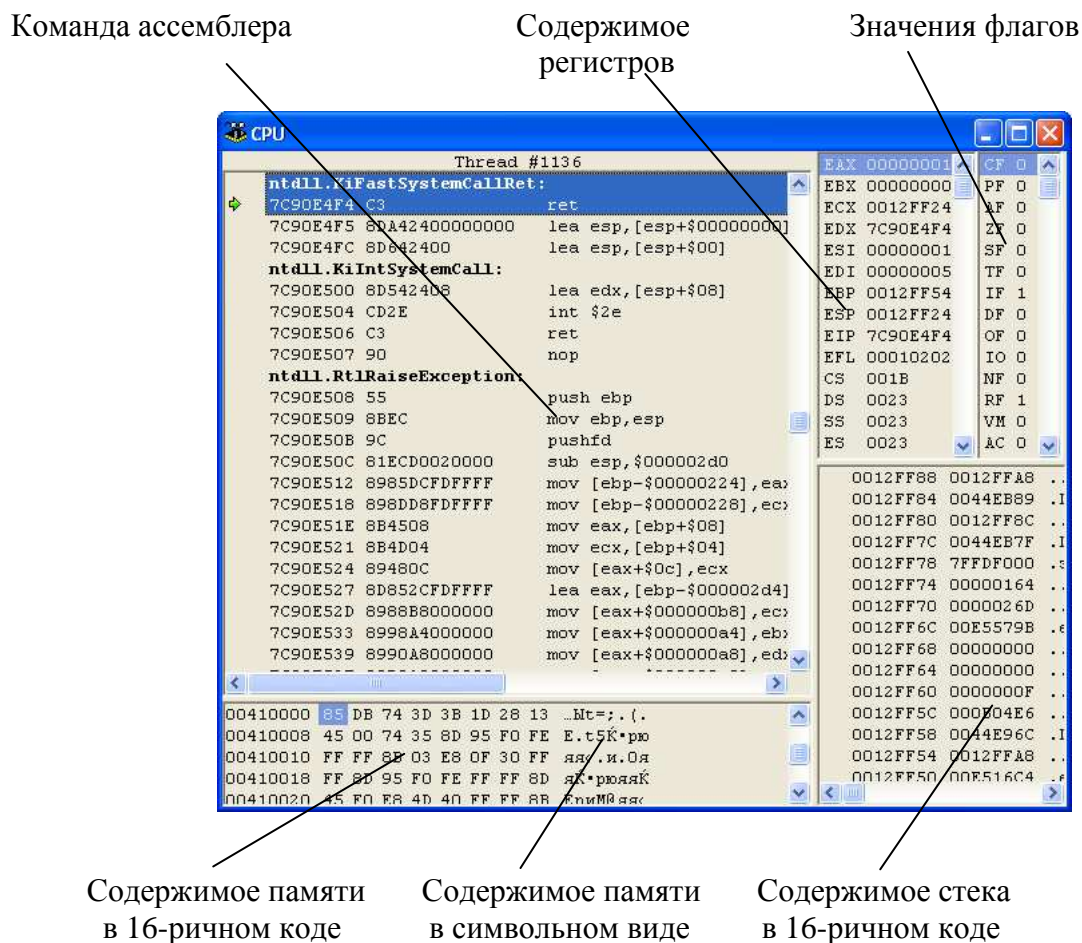


Рис. 2.6. Вид экрана при отладке программы в 16-ном коде (режим CPU)

Причину ошибки определяют, используя один из рассмотренных методов. При этом для проверки гипотез также можно использовать интегрированные средства отладки.

Отладка с использованием независимых отладчиков. При отладке программ иногда используют специальные программы –

отладчики, которые позволяют выполнить любой фрагмент программы в пошаговом режиме и проверить содержимое интересующих программиста переменных. Как правило такие отладчики позволяют отлаживать программу только в машинных командах, представленных в 16-ричном коде.

Общая методика отладки программного обеспечения

Суммируя все сказанное выше, можно предложить следующую методику отладки программного обеспечения, написанного на универсальных языках программирования для выполнения в операционных системах MS DOS и Win32:

1 этап – изучение проявления ошибки – если выдано какое-либо сообщение или выданы неправильные или неполные результаты, то необходимо их изучить и попытаться понять, какая ошибка могла так проявиться. При этом используют индуктивные и дедуктивные методы отладки. В результате выдвигают версии о характере ошибки, которые необходимо проверить. Для этого можно применить методы и средства получения дополнительной информации об ошибке.

Если ошибка не найдена или система просто «зависла», переходят ко второму этапу.

2 этап – локализация ошибки – определение конкретного фрагмента, при выполнении которого произошло отклонение от предполагаемого вычислительного процесса. Локализация может выполняться:

- путем отсечения частей программы, причем, если при отсечении некоторой части программы ошибка пропадает, то это может означать как то, что ошибка связана с этой частью, так и то, что внесенное изменение изменило проявление ошибки;
- с использованием отладочных средств, позволяющих выполнить интересующих нас фрагмент программы в пошаговом режиме и получить дополнительную информацию о месте проявления и характере ошибки, например, уточнить содержимое указанных переменных.

При этом, если были получены неправильные результаты, то в пошаговом режиме проверяют ключевые точки процесса формирования данного результата.

Как подчеркивалось выше, ошибка не обязательно допущена в том месте, где она проявилась. Если в конкретном случае это так, то переходят к следующему этапу.

3 этап – определение причины ошибки – изучение результатов второго этапа и формирование версий возможных причин ошибки. Эти версии необходимо проверить, возможно, используя отладочные средства для просмотра последовательности операторов или значений переменных.

4 этап – исправление ошибки – внесение соответствующих изменений во все операторы, совместное выполнение которых привело к ошибке.

5 этап – повторное тестирование – повторение всех тестов с начала, так как при исправлении обнаруженных ошибок часто вносят в программу новые.

Следует иметь в виду, что процесс отладки можно существенно упростить, если следовать основным рекомендациям структурного подхода к программированию:

- программу наращивать «сверху-вниз», от интерфейса к обрабатывающим подпрограммам, тестируя ее по ходу добавления подпрограмм;
- выводить пользователю вводимые им данные для контроля и проверять их на допустимость сразу после ввода;
- предусматривать вывод основных данных во всех узловых точках алгоритма (ветвлениях, вызовах подпрограмм).

Кроме того, следует более тщательно проверять фрагменты программного обеспечения, где уже были обнаружены ошибки, так как вероятность ошибок в этих местах по статистике выше. Это вызвано следующими причинами. Во-первых, ошибки чаще допускают в сложных местах или в тех случаях, если спецификации на реализуемые операции недостаточно проработаны. Во-вторых, ошибки могут быть результатом того, что программист устал, отвлекся или плохо себя чувствует. В-третьих, как уже упо-

миналось выше, ошибки часто появляются в результате исправления уже найденных ошибок.

Возвращаясь к рис. 2.2, можно отметить, что проще всего обычно искать ошибки определения данных и ошибки накопления погрешностей: их причины локализованы в месте проявления. Логические ошибки искать существенно сложнее. Причем обнаружение ошибок проектирования требует возврата на предыдущие этапы и внесения соответствующих изменений в проект. Ошибки кодирования бывают как простые, например, использование неинициализированной переменной, так и очень сложные, например, ошибки, связанные с затиранием памяти.

Затиранием памяти называют ошибки, приводящие к тому, что в результате записи некоторой информации не на свое место в оперативной памяти, затираются фрагменты данных или даже команд программы. Ошибки подобного рода обычно вызывают появление сообщения об ошибке. Поэтому определить фрагмент, при выполнении которого ошибка проявляется, несложно. А вот определение фрагмента программы, который затирает память – сложная задача, причем, чем длиннее программа, тем сложнее искать ошибки такого рода. Именно в этом случае часто прибегают к удалению из программы частей, хотя это и не обеспечивает однозначного ответа на вопрос, в какой из частей программы находится ошибка. Эффективнее попытаться определить операторы, которые записывают данные в память не по имени, а по адресу, и последовательно их проверить. Особое внимание при этом следует обращать на корректное распределение памяти под данные.

Раздел 3
**ПОДГОТОВКА К КОЛЛОКВИУМАМ,
ЗАЧЕТАМ И ЭКЗАМЕНАМ**

Проверка знаний студента, оценка глубины понимания им теоретического материала и умение применять его к конкретным задачам, осуществляется посредством коллоквиумов, зачетов и экзаменов.

В результате подготовки к коллоквиумам, зачетам и экзаменам из отдельных сведений и деталей у студентов должно сформироваться представление об общем содержании соответствующей дисциплины, стать понятной методика предмета, его система.

Формы самостоятельной работы студентов при подготовке к коллоквиумам, зачетам, экзаменам могут быть следующими:

- повторение пройденного материала по всем разделам программы, используя конспекты лекций, учебно-методическую и учебную литературу, компьютерные обучающие программы;
- самостоятельное определение пробелов в освоении той или иной темы, раздела, и их восполнение;
- работа с тест-тренажерами с целью определения уровня усвоения теоретического материала, проверки практических умений и навыков.

Коллоквиум – форма проверки текущей успеваемости студента по законченной теме курса. Он проводится в течение семестра один, два или большее число раз, в зависимости от характера дисциплины, и проходит в форме собеседования преподавателя со студентом. Объем выносимого материала ограничивается одной или несколькими темами.

В конце каждого семестра, в соответствии с учебным планом, в сроки, установленные учебным графиком, проводится зачетная и экзаменационная сессия.

Зачет – форма проверки успешного выполнения лабораторных и расчетно-графических заданий, курсовых проектов (работ), усвоения учебного материала практических и семинарских занятий, а также форма проверки прохождения учебной и профессиональной

практик и выполнения в процессе этих практик всех учебных поручений в соответствии с утвержденной программой.

Форма проведения зачета может быть разной. Он может проводиться в устной и письменной форме, преподаватель может включать в него вопросы как лабораторно-практических занятий, так и лекционных (что особенно уместно, когда по данному предмету не сдается экзамен). Главное отличие зачета от экзамена – почти всегда не пяти, а двухбалльная система оценки (сдал – не сдал).

Для успешной сдачи зачета необходимо, прежде всего, выполнить все требования преподавателя, что предполагает знание этих требований. Нужно как можно раньше выяснить, какие вопросы предстоит готовить и каковы правила самой процедуры (учитывается ли посещаемость, надо ли пропущенные занятия отрабатывать, а если надо, то каким образом и т.д.). Практика показывает, что хорошее посещение занятий является почти полной гарантией получения зачета, так как тогда можно быть в курсе всех требований преподавателя. И, напротив, большое количество пропусков может осложнить жизнь даже сильному студенту. Кроме того, необходимо учитывать, что проблемы могут появиться при распространенном подходе студента к практическим занятиям, когда многие работают первые месяцы вполсилы, накапливая задолженности по выполнению рефератов, практических заданий, конспектов и пр., а перед сессией пытаются все это сделать за одну неделю. Старайтесь распределять силы равномерно по всей дистанции семестра, и тогда зачетная неделя перед сессией будет не самой напряженной, а самой разгрузочной;

Экзамен – итог работы студента за семестр. Экзамены по всей дисциплине или ее части преследуют цель оценить работу обучающегося за курс (семестр), полученные теоретические знания, прочность усвояемости их, развитие творческого мышления, приобретение навыков самостоятельной работы, умение синтезировать полученные знания и применять их для решения практических задач.

Экзамены в вузе проводятся в соответствии с утвержденной программой того курса (или его части), который изучался в текущем семестре или нескольких семестрах. Являясь государственным руководящим документом, программа определяет объем требова-

ний, предъявляемых студентам на экзаменах. Поэтому необходимо пользоваться ею не только во время экзаменов, но и при подготовке к ним. Экзамены могут проводиться по билетам в устной или письменной форме, а также в виде теста.

Подготовка к экзаменам – один из самых ответственных видов самостоятельной работы. Можно сэкономить значительную часть времени, если начинать подготовку заблаговременно. Одно из главных правил – представлять себе общую логику предмета, что достигается проработкой планов лекций, составлением опорных конспектов, схем, таблиц. Фактически основной вид подготовки к экзамену – «свертывание» большого объема информации в компактный вид, а также тренировка в ее «развертывании» (примеры к теории, выведение одних закономерностей из других и т.д.). Владение этими технологиями обеспечивает, пожалуй, более половины успеха. Надо также правильно распределить силы, не только готовясь к самому экзамену, но и позаботившись о допуске к нему (часто это хорошее посещение занятий, выполнение в назначенный срок практических заданий, активность на семинарах). Наконец, необходимо выяснить условия проведения самого экзаменационного испытания, используя для этой цели, прежде всего, консультацию. Преподаватель обычно заранее определяет условия сдачи экзамена: количество и характер вопросов, форму проведения (устно или письменно), возможность использования при подготовке различных материалов и пособий.

В практике высших учебных заведений установился следующий порядок проведения экзаменов и зачетов. На каждую сессию выносятся не более пяти экзаменов и шести зачетов. По желанию студентов могут также сдаваться экзамены и зачеты по факультативным курсам и практикумам, с тем, чтобы результаты были занесены в приложение к диплому, где дается перечень дисциплин, изучавшихся студентом, и проставляются итоговые оценки по ним. Экзамены принимаются, как правило, лектором данного академического потока. Зачет принимает преподаватель, проводивший практические, семинарские или лабораторные занятия, или читавший лекции по данной дисциплине.

Перед экзаменом назначается *консультация*. Цель ее – дать ответы на вопросы, возникшие в ходе самостоятельной подготовки. Хотелось бы обратить особое внимание на важность предэкзаменационных консультаций. Здесь студент имеет полную возможность получить ответ на все неясные ему вопросы. А для этого он должен проработать до консультации весь курс. Кроме того, преподаватель будет отвечать на вопросы других студентов, что будет для вас повторением и закреплением знаний. И еще очень важное обстоятельство: лектор на консультации, как правило, обращает внимание на те разделы, по которым на предыдущих экзаменах ответы были неудовлетворительными, а также фиксирует внимание на наиболее трудных разделах курса.

Некоторые студенты не приходят на консультации либо потому, что считают, что у них нет вопросов к лектору, либо полагают, что у них и так мало времени и лучше самому почитать материал по конспекту или в учебнике. Это глубокое заблуждение. Никакая другая работа не сможет принести столь значительного эффекта накануне экзамена, как консультация преподавателя.

При оценке знаний на экзамене учитываются следующие стороны подготовки студента:

- 1) понимание и степень усвоения теории;
- 2) методическая подготовка;
- 3) знание фактического материала;
- 4) знакомство с обязательной литературой, и, может быть, с современными публикациями по данному курсу
- 5) умение приложить теорию к практике, решать задачи, делать расчеты и т.д.;
- 6) знакомство с историей науки;
- 7) логика, структура и стиль ответа, умение защищать выдвигаемые положения.

ПРИМЕНЕНИЕ ИНФОРМАЦИОННЫХ КОММУНИКАЦИОННЫХ ТЕХНОЛОГИЙ

Благодаря своим конструктивным и функциональным особенностям и возможностям современный персональный компьютер является уникальным техническим средством обучения и получения знаний. Он находит применение в обучении самым разнообразным дисциплинам и служит базой для создания большого числа новых информационных технологий обучения.

Применение компьютера в сфере образования идет достаточно успешно и имеет следующие направления:

- видеоэнциклопедии;
- интерактивные путеводители;
- тренажеры;
- ситуационно-ролевые игры;
- электронные лектории;
- персональные интеллектуальные гиды по различным научным дисциплинам, являющиеся обучающими системами с использованием искусственного интеллекта;
- исследовательское обучение при моделировании изучаемого процесса в аналоговой или абстрактной форме;
- системы самотестирования знаний обучающегося;
- моделирование ситуации до уровня полного погружения – виртуальная реальность.

Современные студенты – в основном сетевое поколение, для которых электронный способ получения информации (в данном случае именно учебной) является нормальной составляющей жизни. Высокие технологии в образовании приветствуются студентами, а полученные знания, умения, навыки пригодятся в самосовершенствовании и карьерном росте. Информационные коммуникационные технологии становятся рабочим инструментом.

Компьютеризированное обучение не может заменить человека-преподавателя, но оно не только может дополнить и усовершенствовать традиционное обучение, а в некоторых областях, в которых

развиваются самостоятельность, творческое мышление, оно играет уникальную роль. Традиционное обучение является преимущественно теоретическим. Если же вести обучение с помощью компьютера, оно приобретает практический уклон: диалоговый характер работы с компьютером, его вычислительные моделирующие возможности predispose к обучению в форме решения задач, и к тому же, задач практической направленности.

Компьютерные обучающие программы

Компьютерные обучающие программы предназначены для самостоятельной работы студентов с учебным материалом при различных формах обучения. Они не подменяют собой традиционные учебные материалы, имеющиеся в наборе студента, а дополняют их, используя возможности современных компьютерных технологий. Компьютерные обучающие программы зачастую включают в себя теоретический материал, разбор решения типовых задач и поясняющие примеры (в том числе интерактивные), графические и анимационные материалы, тесты для самоконтроля и контроля знаний, необходимые дополнительные и сервисные средства.

Понятие компьютерных обучающих программ тесно связано с понятием E-learning (сокращение от англ. Electronic Learning) – система электронного обучения, синоним таких терминов, как электронное обучение, дистанционное обучение, обучение с применением компьютеров, сетевое обучение, виртуальное обучение, обучение при помощи информационных, электронных технологий. К E-learning относятся электронные учебники, образовательные услуги и технологии.

Электронные библиотечные ресурсы ЧГУ

[\(<http://www.chsu.ru/biblioteka/e-library>\)](http://www.chsu.ru/biblioteka/e-library)

- *Электронный каталог библиотеки университета*

Позволяет получить информацию о том, где находится та или иная книга библиотеки. Электронный каталог содержит информацию обо всех книгах, поступивших в библиотеку после 1999 года,

кроме того, частично представлен старый фонд. На сегодняшний день в электронный каталог занесено около 30 тысяч библиографических записей.

- *Электронно-библиотечная система «Библиотех»*

Электронно-библиотечная система (ЭБС) «Библиотех» – ресурс, позволяющий читать книги в электронном виде по различным темам в pdf-формате. В ЭБС представлено более 3 400 книг, в том числе около 500 учебников и учебно-методических пособий. Все студенты и сотрудники Череповецкого университета могут бесплатно пользоваться системой. К ней можно обратиться с любого компьютера, подключенного к сети Интернет. Вход в ЭБС осуществляется под индивидуальными логинами и паролями.

- *Учебная электронная библиотека*

Учебная электронная библиотека ЧГУ содержит учебную, учебно-методическую, научную литературу, изданную преподавателями Череповецкого университета.

**Федеральный интернет-экзамен
в сфере профессионального образования
(<http://www.fepo.ru>)**

ФЭПО – федеральный экзамен в сфере высшего профессионального образования, предполагает тестирование студентов совокупности образовательных программ или одной образовательной программы всех вузов Российской Федерации с использованием Интернет среды в режиме offline или в режиме online. По сути ФЭПО является компьютерным тестом, по результатам которого дается внешняя оценка уровня подготовки студентов на соответствие требованиям государственных образовательных стандартов.

Для того, чтобы отслеживать уровень подготовки студентов, выявлять глубину и актуальность знаний, университет вошел в проект Национального аккредитационного агентства, предполагающий проведение регулярных тестов.

Ссылки на популярные ресурсы

Интернет-университеты

<http://www.intuit.ru>,

<http://www.wikiversity.org>,

<http://lms.chsu.ru>

Веб-конференции (online семинары, лекции)

<http://www.techdays.ru>

Электронные библиотеки

<http://rsl.ru>,

<http://www.elibrary.ru>,

<http://www.chsu.ru/biblioteka>

Видеолекции

<http://univertv.ru>

<http://www.intuit.ru>,

лекционные каналы на <http://youtube.com>

Системы тестирования

<http://www.ast-centre.ru>

<http://www.fepo.ru>

Раздел 5

**САМОСТОЯТЕЛЬНАЯ РАБОТА
В ПРОЦЕССЕ ПРОХОЖДЕНИЯ ПРАКТИКИ**

Практика является составной частью основной образовательной программы высшего профессионального образования.

Основной целью и задачами практики являются: закрепление, расширение, углубление и систематизация теоретических и практических знаний, полученных студентами при изучении общепрофессиональных и специальных дисциплин; приобретение и совершенствование навыков работы для их применения в будущей профессиональной деятельности на предприятиях информационного профиля, в научно-исследовательских лабораториях и т.д.; изучение источников информации, системы обработки и хранения информации, технологических приемов оценки состояния информационных ресурсов предприятия; формирование профессиональных знаний, умений и навыков при принятии самостоятельных решений в реальных производственных условиях путем выполнения работ, входящих в сферу будущей профессиональной деятельности.

Каждый студент во время практики индивидуально должен решить конкретную учебную или научную задачу по предложенной теме, которая может состоять из ряда заданий. Предлагаемые задачи не должны вызывать у студентов особых затруднений при воплощении их на конкретном лабораторном оборудовании, типе ПЭВМ и т.д. Содержание заданий предусматривает выполнение следующих обязательных этапов:

- ознакомление с имеющимися ПЭВМ и периферийным оборудованием, прикладными программами, дополнительной научной литературой и отбор из них тех, которые необходимы для решения данной задачи;
- модернизация или разработка новой математической модели для решения предлагаемой задачи;
- разработка алгоритмов решения поставленной задачи;
- реализация решения поставленной задачи на выбранном языке программирования и проведение тестирования результатов;

– составление отчета, содержащего основные этапы решения задачи с приложением основных научных результатов и выводов.

Практике предшествует организационное собрание студентов, на которое приглашаются все руководители практики от кафедры.

Студенты знакомятся с задачами и программой практики, условиями работы, приказом по практике. Проводится инструктаж о правилах поведения на практике, о ведении дневника, о требованиях к отчету по практике. Руководители практики уточняют со студентами время и место встречи на базе практики.

В период прохождения практики студенты находятся под контролем руководителя практики от предприятия и подчиняются правилам внутреннего распорядка, установленным на базе практики.

Ежедневная производственная работа по программе практики отмечается в дневнике практики, в нем же отражается порядок прохождения практики и накапливаются материалы для отчета. Ежедневно руководитель практики от предприятия просматривает дневники.

Студент самостоятельно планирует выполнение индивидуально-го задания и программы практики и по мере накопления материала оформляет отчет. В соответствии с договором, предприятие предоставляет студентам возможность пользоваться консультациями руководителей, библиотекой, технической и другой документацией.

Отчетная документация

Дневник практики

Во время практики студент обязан ежедневно вести дневник, в котором должен отражать порядок прохождения практики, сведения по технологическим процессам, устройству и характеристикам оборудования, по охране труда и технике безопасности, по организации и экономике производства, сведения, получаемые на консультациях со специалистами предприятия и руководителями практики.

В дневнике должен содержаться отзыв руководителя практики от предприятия, в котором должна даваться предварительная оценка работы студента.

Материалы дневника являются основой для подготовки отчета по практике.

Отчет по практике

Отчет является основным документом, характеризующим работу студента в период практики.

Отчет по практике составляется каждым студентом индивидуально на основе материалов практики и индивидуального задания. Работа над отчетом должна вестись систематически в течение всего периода практики. Содержание излагается с соблюдением правил ЕСПД, сопровождается схемами, рисунками, таблицами. Отчет должен быть сброшюрован. Примерный объем отчета 20 – 30 страниц машинописного текста.

Аттестация по итогам практики проводится на основании оформленного в соответствии с указанными требованиями письменного отчета и отзыва руководителя практики от предприятия. По итогам практики выставляется оценка (отлично, хорошо, удовлетворительно).

Для успешного завершения практики студенту необходимо:

- в течение последних трех дней (для производственно-технологической практики) и в течение последней недели практики (для преддипломной практики) закончить отчет и подписать его у руководителя практики от предприятия;
- в дневнике подписать отзыв у руководителя практики от предприятия;
- сдать зачет с оценкой по программе практики руководителю от кафедры.

Основными показателями при оценке работы студента на практике служит устный ответ при защите отчета по практике и качество выполнения индивидуального задания.

Результаты практики студентов оцениваются по 5-балльной системе.

Оценка 5 (отлично) ставится, если студент полностью выполнил все задания практики, не имел никаких замечаний по режиму работы во время практики.

Оценка 4 (хорошо) ставится, если студент полностью выполнил все задания практики, однако имелись некоторые замечания.

Оценка 3 (удовлетворительно) ставится, если студент не полностью выполнил задания практики, не имел никаких замечаний по режиму работы во время практики.

Оценка 2 (неудовлетворительно) ставится, если студент не выполнил задания практики, имел замечания по режиму работы во время практики.

5.1 Учебная практика бакалавра

Учебная практика проводится в компьютерных классах университета во втором семестре.

В результате прохождения учебной практики студент должен приобрести следующие практические умения и навыки:

- подбора и классификации информации по заданной теме;
- оценки источников информации, качества представленных материалов;
- практической работы с рассмотренным прикладным программным обеспечением;
- использования ППО для решения поставленных задач;
- составления отчетов по результатам выполненной работы.

Задания выполняются с использованием средств обеспечения освоения дисциплины. Для выполнения индивидуальных практических заданий студент должен самостоятельно осуществить поиск недостающей информации, используя список основной и дополнительной литературы, рекомендации руководителя. Самостоятельная работа студентов предусматривает проработку дополнительных источников в читальных залах и на абонементе библиотеки ЧГУ, а также поиск необходимой информации в Интернет.

Структура отчета о прохождении учебной практики

Титульный лист должен содержать: наименование учебного заведения и структурного подразделения (институт, кафедра), в кото-

ром обучается студент, шифр и наименование направления, название практики, место прохождения практики, Ф.И.О. студента и его подпись, подпись руководителя практики от кафедры, год прохождения практики. Форма титульного листа представлена в прил. 5.

Во *введении* приводится формулировка индивидуального задания, определяются цель и задачи практики. Примерный объем введения – 1 – 2 страницы.

В *первом разделе* дается общая характеристика базы практики: название организации, характеристика подразделения, технологические процессы, производственное оборудование, программное обеспечение.

Во *втором разделе* рассматривается прикладное программное обеспечение, необходимое для выполнения индивидуального задания, дается его характеристика, рассматриваются приемы работы, приводятся системные требования.

В *третьем разделе* «Выполнение индивидуального задания» приводится подробное описание выполнения полученных заданий и полученные результаты.

В *заключении* делаются выводы о возможности и целесообразности использования вышерассмотренного ППО для решения практических задач данного класса по профилю направления подготовки.

5.2 Научно-исследовательская практика бакалавра

Научно-исследовательская практика проводится на кафедрах ВУЗов, на базе научно-исследовательских лабораторий учебных заведений и подразделений предприятий в течение четвертого семестра.

В результате прохождения научно-исследовательской практики обучающийся должен приобрести практические навыки, умения:

- подбора и классификации информации по теме исследования согласно заданной структуре;
- оценки источников информации, качества представленных материалов;

- составления докладов, отчетов по результатам своего микроисследования.

Под научно-исследовательскими задачами принято понимать задачи, решение которых актуально для потребителя (производственных подразделений, коллективов разработчиков и т.д.), но в данный момент оно отсутствует, или его качество не удовлетворяет заказчика.

Для решения научной задачи необходимо понять ее актуальность, проблемы исследования, сформулировать научную гипотезу, подобрать методы ее подтверждения, определить цель и задачи исследования. Классическими методами научного решения являются: аналитический метод и методы моделирования (математического, имитационного, аналогового, физического). Для многих научно-технических задач эффективным является подход, сочетающий научный и творческие методы: эвристические приемы, морфологический анализ и синтез решений.

Задания выполняются с использованием средств обеспечения освоения дисциплины: Microsoft Visio, BPWin v.4.0 и Rational Rose. Для выполнения домашних работ и практических заданий студент должен осуществить поиск недостающей информации самостоятельно, используя список основной и дополнительной литературы, рекомендации руководителя. Самостоятельная работа студентов предусматривает проработку дополнительных источников в читальных залах и на абонементе библиотеки ЧГУ, а также поиск необходимой информации в Интернет.

Структура отчета о прохождении научно-исследовательской практики

Титульный лист должен содержать: наименование учебного заведения и структурного подразделения (институт, кафедра), в котором обучается студент, шифр и наименование направления, название практики, место прохождения практики, Ф.И.О. студента и его подпись, подпись руководителя практики от предприятия, подпись руководителя практики от кафедры, год прохождения практики. Форма титульного листа представлена в прил. 5.

Во *введении* приводится формулировка индивидуального или группового задания, определяются цель и задачи практики. Примерный объем введения – 1 – 2 страницы.

В *первом разделе* дается общая характеристика базы практики: название организации (предприятия), кратко характеризуется организация и управление деятельностью научной лаборатории, приводится подробная информация о научной деятельности структурного подразделения (научной лаборатории), где студент проходил научно-исследовательскую практику.

Во *втором разделе* «Постановка задачи» показывается актуальность поставленной задачи, формулируется научная гипотеза (гипотезы), приводятся возможные методы исследования, их анализ, обоснование выбранного подхода.

В *третьем разделе* «Сравнительный анализ научных разработок по теме исследования» проводится обзор и анализ известных научных решений по заданной тематике с указанием их достоинств и недостатков.

В *четвертом разделе* «Выполнение исследовательского задания» делаются выводы о правильности и ошибочности предположений, целесообразности разработки, осуществляет выбор оптимального метода или программы исследований.

5.3 Производственная практика бакалавра

Производственная практика проводится в отделах информационных технологий предприятий и организаций, цехах и производствах, оснащенных автоматизированными системами и комплексами, с которыми заключен договор о прохождении производственной практики студентами данного направления в течении 6 семестра.

В результате прохождения практики студент должен приобрести следующие практические умения и навыки:

– пользования периодическими реферативными и справочно-информационными изданиями по профилю направления подготовки;

- оценки источников информации, качества представленных материалов;
- практической работы с отдельными пакетами программ компьютерного моделирования и проектирования объектов профессиональной деятельности;
- анализа технического уровня изучаемого аппаратного и программного обеспечения средств ВТ для определения их соответствия действующим техническим условиям и стандартам;
- применения измерительной техники для контроля и изучения отдельных характеристик используемых средств ВТ.

Индивидуальные задания выполняются с использованием средств обеспечения освоения предметной области, в том числе и программного обеспечения, используемого в данной организации. Темы индивидуальных заданий связаны с вопросами разработки, эксплуатации и сопровождения программных и аппаратно-программных средств вычислительной техники и автоматизированных систем. Для выполнения индивидуальных практических заданий студент должен осуществить поиск недостающей информации самостоятельно, используя список основной и дополнительной литературы, рекомендации руководителя. Самостоятельная работа студентов предусматривает проработку дополнительных источников в читальных залах библиотеки базы практики, а также поиск необходимой информации в Интернет.

Структура отчета о прохождении производственной практики

Титульный лист должен содержать: наименование учебного заведения и структурного подразделения (институт, кафедра), в котором обучается студент, шифр и наименование направления, название практики, место прохождения практики, Ф.И.О. студента, Ф.И.О. руководителя практики от кафедры, Ф.И.О. руководителя практики от предприятия, год прохождения практики. Форма титульного листа представлена в прил. 5.

Во *введении* приводится формулировка индивидуального задания, определяются цели производственной практики и задачи для ее достижения, указываются используемые методы разработки и

источники информации Примерный объем введения – 1 – 2 страницы.

В *первом разделе* дается общая характеристика базы практики: название организации (предприятия), кратко характеризуется организация и управление деятельностью подразделения, технологические процессы и соответствующее производственное оборудование.

Во *втором разделе* «Организация производства» описывается технологический процесс предприятия, методы управления базой практики.

В *третьем разделе* «Структура и функции подразделения базы практики» описывается структура подразделения базы практики, приводятся функции, возложенные на данное подразделение.

В *четвертом разделе* «Реализация индивидуального задания» приводится формулировка индивидуального задания. Далее происходит ознакомление с имеющимися ПЭВМ и периферийным оборудованием, прикладными программами, дополнительной научной литературой, модернизация или разработка новой математической модели, разработка алгоритмов, реализация решения поставленной задачи на выбранном языке программирования и проведение тестирования результатов.

В *пятом разделе* «Мероприятия по охране труда и технике безопасности» дается краткая характеристика безопасности производственного процесса и микроклимата в помещении, приводятся опасные и вредные производственные факторы, присутствующие в подразделении, описываются мероприятия, принятые по электро- и пожаробезопасности. Рекомендации по проведению анализа факторов безопасности жизнедеятельности приведены в разделе 8.

В *заключении* приводятся теоретические и практические выводы и результаты работы. Они должны излагаться в краткой форме и давать представление о степени выполнения поставленной перед студентом задачи. Заключение также должно содержать предполагаемую тему выпускной квалификационной работы. Примерный объем заключения – 1 страница.

В *списке литературы* приводятся все книги, журналы, использованные студентом в ходе прохождения производственной практики, а также ссылки на сайты Интернета.

5.4 Производственно-технологическая и преддипломная практики специалистов

Производственно-технологический этап направлен на закрепление связи теоретического обучения с практической деятельностью, обеспечивающей прикладную направленность и адаптацию к рынку труда, на повышение уровня практической подготовки студентов, на формирование профессиональных качеств у будущих инженеров и бакалавров, на приобретение ими опыта работы в производственном коллективе.

Производственно-технологическая практика делится на бакалаврскую производственно-технологическую практику (шестой семестр) и инженерную производственно-технологическую практику (восьмой семестр).

При прохождении *бакалаврской производственно-технологической практики* студенты должны изучить и освоить:

- организацию и управление деятельностью подразделения;
- технологические процессы и соответствующее производственное оборудование в подразделениях предприятия-базы практики;
- действующие стандарты, технические условия, положения и инструкции по эксплуатации аппаратных и программных средств вычислительной техники, периферийного и связанного оборудования, по программам испытаний и оформлению технической документации;
- пакеты прикладного программного обеспечения, используемые при проектировании аппаратных и программных средств;
- порядок пользования периодическими реферативными и справочно-информационными изданиями по профилю работы подразделения.

При прохождении *инженерной производственно-технологической практики* студенты должны изучить и освоить:

- обеспечение планирования и финансирования разработок;
- методы определения экономической эффективности исследований и разработок аппаратных и программных средств;

- правила эксплуатации средств вычислительной техники, измерительных приборов или технологического оборудования, имеющегося в подразделении, а также их обслуживание;
- вопросы обеспечения безопасности жизнедеятельности и экологической чистоты;
- методы анализа технического уровня изучаемого аппаратного и программного обеспечения средств вычислительной техники для определения их соответствия действующим техническим условиям и стандартам;
- методики применения измерительной техники для контроля и изучения отдельных характеристик используемых средств вычислительной техники;
- порядок и методы проведения и оформления патентных исследований;
- порядок пользования периодическими реферативными и справочно-информационными изданиями по профилю работы подразделения.

Структура отчета по производственно-технологической практике

Титульный лист должен содержать наименование учебного заведения и структурного подразделения (институт, кафедра), в котором обучается студент, шифр и наименование специальности, название практики, место прохождения практики, Ф.И.О. студента и его подпись, подпись руководителя практики от предприятия, подпись руководителя практики от кафедры, год прохождения практики. Форма титульного листа представлена в прил. 5.

Во *введении* приводится формулировка индивидуального задания, определяются цели производственно-технологической практики и задачи для ее достижения, указываются используемые методы разработки и источники информации. Примерный объем введения – 1 – 2 страницы.

В *первом разделе* дается общая характеристика базы практики: название организации (предприятия), кратко характеризуется организация и управление деятельностью подразделения, технологические процессы и соответствующее производственное оборудование.

Во *втором разделе* «Организация производства» описывается технологический процесс предприятия, методы управления базой практики.

В *третьем разделе* «Структура и функции подразделения базы практики» описывается структура подразделения базы практики, приводятся функции, возложенные на данное подразделение.

В *четвертом разделе* «Реализация индивидуального задания» приводится формулировка индивидуального задания. Далее происходит ознакомление с имеющимися ПЭВМ и периферийным оборудованием, прикладными программами, дополнительной научной литературой, модернизация или разработка новой математической модели, разработка алгоритмов, реализация решения поставленной задачи на выбранном языке программирования и проведение тестирования результатов.

В *пятом разделе* «Мероприятия по охране труда и технике безопасности» дается краткая характеристика безопасности производственного процесса и микроклимата в помещении, приводятся опасные и вредные производственные факторы, присутствующие в подразделении, описываются мероприятия, принятые по электро- и пожаробезопасности.

В *заключении* приводятся теоретические и практические выводы и результаты работы. Они должны излагаться в краткой форме и давать представление о степени выполнения поставленной перед студентом задачи. Заключение также должно содержать предполагаемую тему дипломной работы (проекта). Примерный объем заключения – 1 страница.

В *списке литературы* приводятся все книги, журналы, использованные студентом в ходе прохождения преддипломной практики, а также ссылки на сайты Интернета.

Преддипломный этап направлен на приобретение студентом опыта в исследовании актуальной научной проблемы или решении реальной инженерной задачи, а также на закрепление связи теоретического обучения с практической деятельностью, обеспечивающей прикладную направленность и адаптацию к рынку труда.

При прохождении преддипломной практики студенты должны:

изучить:

- проектно-технологическую документацию, патентные и литературные источники в целях их использования при выполнении выпускной квалификационной работы;
- назначение, состав, принцип функционирования или организации проектируемого объекта (программы или программной системы);
- отечественные и зарубежные аналоги проектируемого объекта;

выполнить:

- разработку технического задания на дипломный проект (работу) по установленной стандартом форме;
- сравнительный анализ возможных вариантов реализации научно-технической информации по теме исследования;
- реализацию некоторых из возможных путей решения поставленной в техническом задании на дипломный проект (работу) задачи;
- технико-экономическое обоснование выполняемой разработки;
- анализ мероприятий по безопасности жизнедеятельности и обеспечению экологической чистоты.

Структура отчета по преддипломной (профессионально-ориентированной) практике:

Титульный лист должен содержать наименование учебного заведения и структурного подразделения (институт, кафедра), в котором обучается студент, шифр и наименование специальности, название практики, место прохождения практики, Ф.И.О. студента и его подпись, подпись руководителя практики от предприятия, подпись руководителя практики от кафедры, год прохождения практики. Форма титульного листа представлена в прил. 5.

Во *введении* приводится формулировка индивидуального задания (тема ВКР), определяются цель практики и задачи для ее достижения, указываются используемые методы разработки и источники информации. Примерный объем введения – 1 – 2 страницы.

В *первом разделе* дается общая характеристика базы практики: название организации (предприятия), кратко характеризуется организация и управление деятельностью подразделения, технологические процессы и соответствующее производственное оборудование.

Во *втором разделе* приводится техническое задание – документ, в котором сформулированы основные цели разработки, требования к ПО, сроки и этапы разработки, а также регламентирован процесс приемо-сдаточных испытаний. Требования к содержанию и оформлению технического задания согласно ГОСТ 19.201-78 приведены в разделе 6.2 данного методического указания.

В *третьем разделе* «Сравнительный анализ возможных вариантов реализации научно-технической информации по теме исследования» проводится обзор и анализ известных проектных решений по тематике ВКР с указанием их достоинств и недостатков, делаются выводы о целесообразности и необходимости разработки.

В *четвертом разделе* «Реализация некоторых из возможных путей решения поставленной в техническом задании на дипломный проект задачи» приводится модульная структура ПО с описанием 1 – 2 модулей системы с точки зрения их назначения, алгоритмы методов и описание межмодульных интерфейсов.

В *пятом разделе* «Технико-экономическое обоснование выполняемой разработки» студент рассчитывает следующие экономические показатели:

- себестоимость программного продукта;
- цену программного продукта;
- годовой экономический эффект;
- срок окупаемости.

Этот раздел завершается выводами о целесообразности и экономической эффективности применения разрабатываемого программного продукта. Подробная методика расчета технико-экономических показателей приведена в разделе 7.

В *шестом разделе* «Анализ мероприятий по безопасности жизнедеятельности и обеспечению экологической чистоты» содержатся следующие подразделы:

- анализ мероприятий по безопасности жизнедеятельности (общая характеристика безопасности производственного процесса; анализ вредных производственных факторов и мероприятия по их снижению; анализ опасных производственных факторов и мероприятия по их снижению; оценка электробезопасности помещения и мероприятия по повышению электробезопасности; оценка поме-

щения по пожарной безопасности и мероприятия по повышению пожарной безопасности; анализ неполадок и аварийных ситуаций в рабочем помещении);

– анализ мероприятий по обеспечению экологической чистоты.

Подробные рекомендации по анализу факторов безопасности жизнедеятельности приведены в разделе 8.

При написании данного раздела студент должен использовать ГОСТы и СНиПы, например:

1. ГОСТ 12.0.003-74. Классификация опасных и вредных производственных факторов.

2. ГОСТ 12.3.002-75. Процессы производственные. Общие требования безопасности.

3. ГОСТ 12.2.003-91. Оборудование производственное. Общие требования безопасности.

4. ГОСТ 12.1.005-88. Общие санитарно-гигиенические требования к воздуху санитарной зоны.

5. ГОСТ 12.1.012-90. Вибрационная безопасность. Общие требования.

6. ГОСТ 12.1.003-83. Шум. Общие требования безопасности.

7. ГОСТ 12.1.019-79 изменение 01.86. Электробезопасность. Общие требования и номенклатура видов защиты.

8. ГОСТ 12.1.004-91. Пожарная безопасность. Общие требования.

9. СНиП 21-01-97. Пожарная безопасность зданий и сооружений.

10. СНиП 230595. Естественное и искусственное освещение.

11. НПБ 88-01. Установки пожаротушения и сигнализации.

В *заключении* приводятся теоретические и практические выводы и результаты работы. Они должны излагаться в краткой форме и давать представление о степени выполнения поставленной перед студентом задачи. В заключении делается акцент на актуальность, степень новизны и практическую значимость разрабатываемого программного продукта. Здесь же приводятся данные об экономической эффективности при внедрении данного программного продукта. Примерный объем заключения – 1 – 2 страницы.

В *списке литературы* приводятся все книги, журналы, использованные студентом в ходе прохождения преддипломной практики, а также ссылки на сайты Интернета.

5.5 Педагогическая практика магистрантов

Педагогическая практика направлена на закрепление связи теоретического обучения с научной деятельностью, обеспечивающую прикладную направленность и адаптацию к педагогической деятельности, формирование навыков ведения самостоятельной учебной работы, планирования учебных занятий, общения со студентами в аудитории. Проводится в высших и средних профессиональных учебных заведениях города и региона, в частности в Череповецком государственном университете, которые осуществляют ведение учебного процесса с использованием лекций, практических и лабораторных занятий.

Соискатель магистерской степени проходит педагогическую практику на выпускающей кафедре под руководством своего научного руководителя. Все виды педагогической деятельности в учебном процессе магистрант проходит в рамках обязанностей ассистента кафедры.

Во время практики магистрантом самостоятельно выполняются следующие работы:

- ознакомление с нормативной учебной документацией (ФГОС ВПО, ГОСТы, ТУ, программы учебных курсов, методические разработки и т.п.);
- разработка задач для практических занятий;
- разработка технических средств обучения;
- подготовка лекционного материала по учебному курсу;
- разработка учебно-методического пособия и подготовка его к изданию;
- участие в разработке и формировании электронных учебно-методических комплексов;
- разработка новых компьютерных программ и инновационных технологий в обучении;

- проведение совместно с руководителем учебных занятий (лабораторный практикум, практические занятия);
- изложение подготовленного лекционного материала на кафедральном семинаре.

Структура отчета о прохождении педагогической практики

Титульный лист должен содержать: наименование учебного заведения и структурного подразделения (институт, кафедра), в котором обучается магистрант, шифр и наименование направления, название практики, место прохождения практики, Ф.И.О. магистранта и его подпись, подпись руководителя практики от кафедры, год прохождения практики. Форма титульного листа представлена в прил. 5.

Во *введении* определяются цели педагогической практики и задачи для их достижения, указываются используемые методы разработки и источники информации. Примерный объем введения – 1 – 2 страницы.

В *первом разделе* «Общая характеристика базы практики» дается название организации (предприятия), кратко характеризуется организация и управление деятельностью подразделения, нормативная учебная документация и соответствующее учебное оборудование.

Во *втором разделе* «Организация учебного процесса» описывается учебный процесс кафедры, виды учебной нагрузки на кафедре.

В *третьем разделе* «Разработка технических средств обучения, новых компьютерных программ и инновационных технологий в обучении» описываются предложения по внедрению инновационных технологий в учебный процесс, разработанные компьютерные программы для организации учебного процесса.

В *четвертом разделе* «Разработка лекционного материала, заданий к практическим и лабораторным работам» приводятся тексты лекций, задания к практическим и лабораторным занятиям с подробным описанием их выполнения и анализом результатов.

В *пятом разделе* «Методика проведения лекций, практических и лабораторных занятий» дается характеристика специфики проведения учебных занятий по данной дисциплине, описываются методы подхода к студентам и бакалаврам, приводятся технические средства, используемые магистрантом при проведении занятий.

В *заключении* приводятся теоретические и практические выводы и результаты работы. Они должны излагаться в краткой форме и давать представление о степени выполнения поставленной перед магистрантом задачи. Примерный объем заключения – 1 страница.

В *списке литературы* приводятся все первоисточники: книги, журналы, использованные магистрантом в ходе прохождения педагогической практики, а также ссылки на сайты в Интернете.

5.6 Научно-исследовательская практика магистранта

Научно-исследовательская практика направлена на закрепление связи теоретического обучения с научной деятельностью, обеспечивающую прикладную направленность и адаптацию к исследовательской деятельности, формирование навыков ведения самостоятельной научно-исследовательской работы, планирования научных экспериментов.

Соискатель магистерской степени проходит научно-исследовательскую практику на выпускающей кафедре под руководством своего научного руководителя.

Во время практики магистрантом самостоятельно изучаются и выполняются следующие работы:

- поиск патентных и литературных источников по разрабатываемой теме с целью их использования при выполнении магистерской работы;
- выбор метода исследования и проведения экспериментальных работ;
- правила эксплуатации исследовательского оборудования;
- выбор метода анализа и обработки экспериментальных данных;
- физические и математические модели процессов и явлений, относящихся к исследуемому объекту;
- информационные технологии в научных исследованиях, программные продукты, относящиеся к профессиональной сфере;
- требования к оформлению научно-технической документации.

За время научно-исследовательской практики магистрант должен в окончательном виде сформулировать тему магистерской диссертации и обосновать актуальность ее разработки.

Структура отчета о прохождении научно-исследовательской практики

Титульный лист должен содержать: наименование учебного заведения и структурного подразделения (институт, кафедра), в котором обучается магистрант, шифр и наименование направления, название практики, место прохождения практики, Ф.И.О. магистранта и его подпись, подпись руководителя практики от кафедры, год прохождения практики. Форма титульного листа представлена в прил. 5.

Во *введении* определяются цели научно-исследовательской практики и задачи для их достижения, указываются используемые методы разработки и источники информации. Примерный объем введения – 1 – 2 страницы.

В *первом разделе* «Общая характеристика базы практики» дается название организации (предприятия), кратко характеризуется организация и управление деятельностью подразделения, нормативная документация и соответствующее научное оборудование.

Во *втором разделе* «Методика написания магистерской диссертации» описываются научные положения и стандарты по написанию магистерской диссертации.

В *третьем разделе* «Поиск патентных и литературных источников» описываются существующие патентные решения по разрабатываемой теме, отечественные и зарубежные аналоги разрабатываемого программного продукта.

В *четвертом разделе* «Экспериментальные исследования» дается характеристика специфики проведения экспериментальных исследований по теме магистерской диссертации, описываются методы проведения экспериментальных исследований, приводятся технические средства, используемые магистрантом при проведении научных исследований.

В *заключении* приводятся теоретические и практические выводы и результаты работы. Они должны излагаться в краткой форме и давать представление о степени выполнения поставленной перед магистрантом задачи. Примерный объем заключения – 1 страница.

В *списке литературы* приводятся все первоисточники: книги, журналы, использованные магистрантом в ходе прохождения научно-исследовательской практики, а также ссылки на сайты в Интернете.

Раздел 6

САМОСТОЯТЕЛЬНАЯ РАБОТА В ПРОЦЕССЕ ВЫПОЛНЕНИЯ НАУЧНО-ИССЛЕДОВАТЕЛЬСКОЙ РАБОТЫ, НАПИСАНИЯ КУРСОВЫХ И ВЫПУСКНЫХ РАБОТ

Научно-исследовательская работа студентов (НИРС) является частью общего образовательного процесса и включается в содержание всех изучаемых дисциплин и видов учебных занятий.

Система НИРС – одно из важнейших средств повышения уровня подготовки специалистов с высшим профессиональным образованием через освоение в процессе обучения по учебным планам и дополнительно основ профессионально-творческой деятельности, методов, приемов и навыков индивидуального и коллективного выполнения научно-исследовательских работ, развитие способностей к научному творчеству, самостоятельности.

Составными частями системы НИРС являются:

- включение элементов научных исследований в учебную программу (защита курсовых работ и дипломных проектов с элементами научных исследований, проблемные лекции, лабораторно-практические занятия с элементами научных исследований, выполнение рефератов по интересным научным тематикам);

- участие во всех видах научно-исследовательских работ, конференциях, конкурсах, представление работ для публикации, пользование услугами научных подразделений;

- научно-исследовательская работа в составе научных студенческих объединений, кружков, позволяющая студентам не только знакомиться с реальными задачами, разрабатывать проекты их решения, но и самим осуществлять свои предложения на практике.

В процессе выполнения научно-исследовательской работы, написания курсовых и выпускных работ студентами могут быть использованы следующие формы самостоятельной работы:

- теоретический анализ учебной и научной литературы в соответствии с утвержденной темой;

- планирование, подбор методов, построение модели, организация и проведение экспериментального или эмпирического исследования, интерпретация данных, выводы по работе;

– самостоятельная подготовка итогового документа: научного доклада, научного реферата (обзора), статьи (тезисов доклада), курсовой или выпускной работы.

6.1 Курсовая работа (проект)

Курсовое проектирование должно способствовать закреплению, углублению и обобщению знаний, полученных студентами во время обучения, применению этих знаний к комплексному решению конкретных задач.

Курсовые работы (проекты) являются одним из основных видов самостоятельной работы студентов в рамках содержания основной образовательной программы и направлены на развитие исследовательской сущности. Именно при таком виде выполняемых работ студент учится работать с информацией, анализировать ее и выделять главное из частного. Именно этот вид работы студента в большей степени воспитывает дисциплинированного специалиста и учит придерживаться норм и правил. Этому так же способствует и постепенное усложнение задач студента при написании курсовой работы. На втором курсе, курсовая работа, скорее напоминает усложненный вариант реферата, и ее написания не предоставляет никакой сложности. Уровень сложности растет постепенно к пятому курсу, и курсовая работа приобретает статус дипломной работы. Системой курсовых работ (проектов) студент готовится к выполнению выпускной квалификационной работы.

Курсовые работы (проекты) предусматриваются учебным планом со 2 по 10 семестры, как правило, после изучения теоретического курса.

Выдача заданий на курсовую работу (проект) фиксируется на кафедре и должна проводиться не позднее, чем за 2 месяца до защиты. Индивидуальная беседа руководителя со студентом по заданию является необходимым условием. В ходе беседы руководитель выясняет степень подготовленности студента к выполнению данного задания, дает консультацию о порядке выполнения задания и график работы. В результате этой беседы тема может быть уточнена. Дальнейшие консультации по курсовой работе (проекту) определяются кафедрой вне

расписания учебных занятий и могут быть как фронтальными, так и индивидуальными. Курсовая работа (проект) перед сдачей на проверку должна быть подписана студентом. На проверку преподавателю отводится от 3 до 7 дней. Если курсовая работа (проект) удовлетворяет требованиям, преподаватель делает запись о допуске к ее защите, если не удовлетворяет – работа выдается студенту на доработку с рецензией. Защита курсовой работы (проекта) может проводиться как в форме публичного выступления, так и в форме индивидуального собеседования. Студенту отводится 5 – 7 минут для доклада и до 10 минут для ответа на вопросы. Сроки защиты сообщаются студенту кафедрой не менее чем за 1 месяц до защиты. Студент, не представивший курсовую работу (проект) или не защитивший ее на положительную оценку, считается имеющим академическую задолженность. При повторном обучении или восстановлении студенту выдается новая тема.

Курсовое проектирование по дисциплинам кафедры ПО ВТ и ИС предназначено для формирования у студентов навыков разработки программного обеспечения (ПО) и оформления курсового проекта (работы).

Составление программной документации – очень важный процесс. На каждое ПО разрабатывается документация двух типов: для пользователей различных групп и для разработчиков.

Содержание расчетно-пояснительной записки

Согласно ГОСТ 7.32-91 (ИСО 5966-82) «Отчет по научно-исследовательской работе. Структура и правила оформления» пояснительная записка содержит:

- титульный лист (образец в прил. 6),
- аннотацию,
- оглавление,
- введение,
- основную часть,
- заключение,
- список литературы,
- приложения.

Основная часть расчетно-пояснительной записки должна содержать все разделы, отражающие этапы создания программного продукта:

1. *Изучение и описание предметной области.* На этом этапе необходимо проанализировать задание на курсовую работу, изучить предметную область задачи, выполнить обзор и провести анализ существующих решений аналогичных задач.

2. *Постановка задачи.* В процессе постановки задачи следует определить требования к программному продукту:

- установить перечень исходных данных, их характеристики и способ представления;
- установить набор выполняемых функций;
- установить перечень результатов, их характеристики и способ представления;
- уточнить среду функционирования программного продукта (параметры технических средств, версия операционной системы и другого программного обеспечения, с которым будет взаимодействовать программный продукт).

В результате составляется техническое задание на разработку программного обеспечения в соответствии с ГОСТ 19.201-78, которое служит основанием для дальнейшей работы.

3. *Выбор структур данных для решения поставленной задачи.* На этом этапе анализируются средства выбранного языка программирования, выбираются типы данных, необходимые для решения задачи и создается таблица идентификаторов следующего вида:

Наименование	Обозначение	Тип данных

4. *Логическое проектирование* предполагает изучение или разработку алгоритмов, решающих данную задачу. При изучении уже существующих алгоритмов следует обратить внимание на расчетные формулы, характеристики алгоритма по скорости, точности, требуемой памяти, области применения. Необходимо понять теоретические возможности существующего алгоритма и суметь реализовать его, сохраняя все его достоинства.

При разработке собственных алгоритмов следует использовать метод пошаговой детализации. Для оформления алгоритмов в виде блок-схем необходимо следовать требованиям ГОСТ 19.701-90 (ISO 5807-85).

5. *Физическое проектирование программного обеспечения.* В процессе физического проектирования разрабатывается модульная структура программного обеспечения. Здесь следует описать взаимосвязь модулей и привести спецификации каждого модуля в таблицах вида:

Имя модуля	Заголовок процедуры или функции	Формальные параметры	Выполняемое действие

6. *Проектирование интерфейса.* На данном этапе дается обоснование выбора типа пользовательского интерфейса, описание и представление форм ввода/вывода.

7. *Кодирование.* На этапе кодирования разработанные алгоритмы реализуют, составляя по ним текст программы. В расчетно-пояснительной записке при этом помещают описание программы либо в виде комментариев в тексте программы, либо отдельно (при этом все строки программы должны быть пронумерованы).

8. *Тестирование.* На этапе тестирования разрабатываются тестовые данные и оформляются в виде таблицы:

Исходные данные	Тестируемый модуль или подпрограмма	Ожидаемый результат

Результаты выполнения тестирования также оформляются в виде таблицы:

Дата и время тестирования	Тестируемый модуль или подпрограмма	Кто проводил тестирование	Описание теста	Результаты тестирования
				«Успех» или описание неполадки

Руководство пользователя

Составление документации для пользователей имеет свои особенности, так как пользователь, как правило, не является профессионалом в области разработки программного обеспечения. Поэтому руководство должно содержать подробные инструкции, необходимые пользователю на доступном языке, без узкоспециальной терминологии. Необходимые термины, используемые в руководстве, объясняются.

Руководство пользователя содержит следующие разделы:

- общие сведения о программе,
- описание установки,
- описание запуска,
- инструкции по работе (или описание пользовательского интерфейса),
- сообщения пользователю.

В разделе *Общие сведения о программе* содержится наименование программы, краткое описание ее функций, реализованных методов и областей применения.

В разделе *Описание установки* содержится подробное описание действий по установке программы и сообщений, которые при этом могут быть получены.

В разделе *Описание запуска* описываются действия по запуску программы и сообщения, которые при этом могут быть получены.

В раздел *Инструкции по работе* содержится описание режимов работы, возможных настроек, форматов ввода-вывода информации.

В разделе *Сообщения пользователю* содержится перечень возможных сообщений, описание их содержания и действий, которые необходимо предпринять по этим сообщениям.

В зависимости от вида ПО, может возникнуть необходимость создания отдельного руководства системного программиста и (или) оператора.

Согласно ГОСТ 19.503-79 «Руководство системного программиста. Требования к содержанию и оформлению» руководство системного программиста содержит всю информацию, необходимую для установки программного обеспечения, его настройки и проверки работоспособности.

Руководство системного программиста содержит следующие разделы:

- общие сведения о программном обеспечении,
- структура,
- настройка,
- проверка,
- дополнительные возможности,
- сообщения системному программисту.

Согласно ГОСТ 19.505-79 «Руководство оператора. Требования к содержанию и оформлению» *руководство оператора* содержит следующие разделы:

- назначение программы,
- условия выполнения программы,
- выполнение программы,
- сообщения оператору.

6.2 Содержание и оформление технического задания

Согласно ГОСТ 19.201-78 «Техническое задание. Требования к содержанию и оформлению», техническое задание содержит следующие части:

- титульный лист (образец приведен в прил. 7),
- введение,
- основания для разработки,
- назначение разработки,
- требования к программе,
- требования к программной документации,
- технико-экономические показатели,
- стадии и этапы разработки,
- порядок контроля и приемки.

Введение включает наименование и краткую характеристику области применения программы, а также объекта, в котором предполагается их использовать.

В разделе *Основания для разработки* содержится наименование документа, на основании которого ведется разработка, организации, утвердившей данный документ, дата его утверждения, а также наименование темы разработки.

В разделе *Назначение разработки* содержится описание функционального и эксплуатационного назначения программы.

В разделе *Требования к программе* содержатся следующие подразделы:

- требования к функциональным характеристикам (перечисляются выполняемые функции, описываются состав, характеристики и формы представления исходных данных и результатов, а также указываются критерии эффективности: максимально допустимое время ответа системы, максимальный объем используемой оперативной и внешней памяти и др.);

- требования к надежности (указываются требования к обеспечению надежного функционирования, время восстановления после сбоя, контроль входной и выходной информации, создание резервных копий промежуточных результатов);

- условия эксплуатации (температура окружающей среды, относительная влажность воздуха для выбранных типов носителей данных, вид обслуживания, необходимое количество и квалификация персонала);

- требования к составу и параметрам технических средств (тип микропроцессора, объем памяти, наличие внешних устройств и т.п.);

- требования к информационной и программной совместимости (определяется язык и среда программирования, используемая операционная система и другие системные и пользовательские программные средства, с которыми взаимодействует разрабатываемое программное обеспечение, а также степень защиты информации);

- требования к маркировке и упаковке (указываются требования к маркировке ПО, варианты и способы упаковки);

- требования к транспортированию и хранению (условия транспортирования ПО, место хранения, условия и сроки хранения, условия складирования);

- специальные требования.

В разделе *Требования к программной документации* указывается наличие руководства программиста, руководства пользователя, ру-

ководства системного программиста, расчетно-пояснительной записки и т.п.

Раздел *Технико-экономические показатели* содержит ориентировочную экономическую эффективность, предполагаемую годовую потребность и экономические преимущества разработки по сравнению с существующими отечественными и зарубежными аналогами.

Раздел *Стадии и этапы разработки* содержит этапы разработки, содержание работ, сроки и результат разработки. Оформляется в виде таблицы:

Наименование этапа разработки ПО	Сроки разработки	Результат выполнения	Отметка о выполнении

Раздел *Порядок контроля и приемки* содержит виды испытаний и общие требования к приемке работ. Оформляется в виде таблицы:

Наименование контрольного этапа выполнения курсовой работы	Сроки контроля	Результат выполнения	Отметка о приемке результата контрольного этапа

В случае если какие-либо требования, предусмотренные техническим заданием, к ПО не предъявляются, в соответствующем месте необходимо указать «Требования не предъявляются».

В зависимости от особенностей разрабатываемого ПО, разрешается уточнение содержания разделов (использование подразделов, ввод новых разделов, объединение их).

6.3 Выпускная квалификационная работа бакалавра

Выпускная квалификационная работа (ВКР) является завершающим этапом подготовки бакалавра. Успешная защита ВКР служит основанием для присвоения выпускнику квалификации бакалавр.

В процессе выполнения ВКР студент должен проявить самостоятельность, умение обосновывать принимаемые решения, спо-

способность решать поставленные перед ним научно-технические задачи. В этом ему помогут современные средства для разработки спецификаций: Erwin, BPwin и Rational Rose, которые описаны в разделе 9.

Перед студентами в процессе проектирования стоят следующие цели и задачи:

- систематизация и закрепление теоретических знаний по профессиональным дисциплинам, применение этих знаний при разработке программно-информационных систем, информационных технологий, программного обеспечения средств вычислительной техники и автоматизированных систем;
- развитие навыков самостоятельной работы с учебно-методической и научной литературой, законодательными и нормативными документами;
- овладение студентами методикой исследования, обобщения и логического изложения материала.

В ходе проектирования студент обязан:

- правильно применять теоретические положения математических и естественнонаучных, профессиональных дисциплин и дисциплин профилизации;
- уметь самостоятельно выявлять и анализировать аналоги и прототипы разрабатываемых программно-информационных систем, информационных технологий, программного обеспечения средств вычислительной техники и автоматизированных систем;
- проявлять организаторские способности и навыки делового общения для преодоления реальных трудностей, возникающих при разработке и внедрении новых технических решений.

Структура выпускной квалификационной работы бакалавра

ВКР содержит следующие разделы:

- титульный лист,
- задание на ВКР,
- аннотацию,
- оглавление,
- введение,

- основную часть,
- заключение,
- список литературы,
- приложения,
- графический материал.

Титульный лист должен содержать: наименование учебного заведения и структурного подразделения (институт, кафедра), в котором обучается студент, шифр и наименование направления, наименование профиля, название темы ВКР, Ф. И. О. студента и его подпись, согласование допуска к защите с директором института, зав. выпускающей кафедрой, руководителем и консультантами, нормоконтролером, название города и год окончания учебного заведения, где выполнялась работа. Форма титульного листа представлена в прил. 8 данного учебно-методического пособия.

После утверждения темы студент совместно с руководителем составляет *задание на выполнение ВКР* в двух экземплярах на бланке, форма которого приведена в прил. 9. Один экземпляр студент прикладывает к расчетно-пояснительной записке, второй – остается на кафедре и является основанием для включения студента, темы ВКР, руководителя, рецензента и консультантов в приказ на ВКР. Задание подписывается студентом, руководителем, консультантами и утверждается заведующим кафедрой.

В *аннотации* в краткой форме отражаются цель и объект проектирования, полученные результаты, перечень основных терминов и определений, количество иллюстраций, таблиц, приложений и использованных первоисточников. Объем аннотации – не более 1 страницы. Бланк аннотации представлен в прил. 14.

Оглавление включает в себя: введение, наименование всех разделов и подразделов, заключение, список литературы, перечень приложений с указанием номеров страниц. Пример оглавления приведен в прил. 15 (для структурного подхода) и в прил. 16 (для объектного подхода).

Во *введении* обосновывается актуальность разработки, определяются цель работы и задачи для ее достижения, указываются используемые методы и источники информации. Примерный объем введения 1 – 2 страницы.

Основная часть содержит, как правило, семь разделов, которые могут иметь пункты (например, 5.3) и подпункты (например, 5.3.1). В них содержатся необходимые обоснования, математические модели, модульная структура системы, функциональные схемы, диаграммы, описания и др. Каждый раздел заканчивается выводами, облегчающими понимание замысла, логики изложения и результатов исследования.

В разделе «Сравнительный анализ отечественных и зарубежных аналогов проектируемого объекта» проводится обзор и анализ известных проектных решений по тематике ВКР с указанием их достоинств и недостатков, делаются выводы о целесообразности и необходимости разработки.

В разделе «Выбор технологии, среды и языка программирования» дается обоснование выбора модели жизненного цикла программного продукта, подхода к разработке, а также выбора инструментальных средств.

В разделе «Анализ процесса обработки информации, выбор структур данных для ее хранения, выбор методов и алгоритмов решения задачи» уточняются требования, сформулированные в техническом задании.

В разделе «Разработка спецификаций проектируемой системы» дается полное и точное описание функций и ограничений создаваемого продукта, при этом используются схемы и/или диаграммы. Каждое графическое представление должно сопровождаться соответствующими подписями и пояснениями.

В разделе «Проектирование системы» приводятся детальные логическая и физическая модели разрабатываемой системы (структурная и функциональная схемы, модульная структура с описанием всех модулей системы с точки зрения их назначения, карты Константайна, диаграммы классов, алгоритмы функций (методов) и описание межмодульных интерфейсов). Пример заполнения спецификации на программное изделие приведен в прил. 20.

В разделе «Проектирование интерфейса пользователя» дается описание разработки интерактивного графического интерфейса.

В разделе «Выбор стратегии тестирования, разработка тестов, программа и методика испытаний» определяются тестируемые модули или подсистемы, окружение при тестировании (аппаратное и

программное обеспечение и используемые ресурсы) и результаты тестирования, которые могут оформляться в виде табл. 6.1.

Таблица 6.1

Результаты тестирования

Дата и время тестирования	Тестируемый модуль и подсистемы	Кто проводил тестирование	Способ тестирования	Результаты тестирования
		<i>Разработчик или тестировщик</i>	<i>Автоматически или вручную</i>	<i>«Успех» или описание неполадки</i>

Таблица должна содержать описание всех тестов, их название, ответственного за тестирование, способ реализации теста, требование, проверяемое данным тестом, а также результат, полученный после выполнения теста. При необходимости в нее могут быть внесены дополнительные поля, а после таблицы – пояснения.

Наборы тестовых данных могут быть оформлены в виде табл. 6.2 (столбцы можно добавлять свои).

Таблица 6.2

Наборы тестовых данных

Тестовое значение	Реакция подсистемы	Пояснения

В *заключении* приводятся теоретические и практические выводы и результаты работы. Они должны излагаться в краткой форме и давать представление о степени выполнения поставленной перед выпускником задачи. В заключении делается акцент на актуальность, практическую значимость разработанной системы. Примерный объем заключения – 1-2 страницы.

Рекомендуемый объем расчетно-пояснительной записки – 50-65 страниц.

Перечень *литературы* помещается непосредственно после основного текста работы. Библиографический список используемой литературы оформляется в соответствии с требованиями действующих ГОСТов.

В *приложения* включают материал, не вошедший в пояснительную записку, но представляющий интерес для более глубокого понимания: 1 – техническое задание (ГОСТ 19.201-78), 2 – схемы и/или диаграммы (ГОСТ 19.701-90), 3 – текст программы (ГОСТ 19.401-78), 4 – спецификация (ГОСТ 19.202-78), 5 – руководство пользователя (ГОСТ 19.503-79, ГОСТ 19.504-79, ГОСТ 19.505-79), 6 – наборы тестовых данных и результатов тестирования (ГОСТ 19.301-79).

Конкретный *графический материал* зависит от темы ВКР и согласовывается с руководителем работы.

Плакаты, выносимые на защиту, выполняются на листе ватмана формата А1 (594 × 841) тушью или распечатываются на плоттере. Количество плакатов – 5 – 8.

Примерный перечень плакатов (для структурного подхода).

1. Постановка задачи.
2. Логическая модель (функциональные диаграммы, диаграммы процессов, диаграммы потоков данных, диаграммы переходов состояний).
3. Диаграммы структур данных (диаграммы «сущность-связь»).
4. Физическая модель (структурная и/или функциональная схемы системы).
5. Схема взаимодействия программных модулей системы.
6. Алгоритмы функционирования основных модулей системы.
7. Результаты испытаний системы.

Примерный перечень плакатов (для объектного подхода).

1. Постановка задачи.
2. Модель использования (диаграммы вариантов использования).
3. Концептуальная модель предметной области (контекстная диаграмма классов).

4. Модель процессов (диаграммы последовательностей системы, диаграммы деятельности).
5. Диаграммы пакетов, диаграммы классов пакетов.
6. Модель реализации и развертывания (диаграммы компонентов, диаграммы размещения).
7. Результаты испытаний системы.

Самостоятельная работа студента при написании и защите ВКР

ВКР разрабатывается студентом самостоятельно. Студент составляет календарный план-график выполнения ВКР с указанием очередности отдельных этапов и согласует его с руководителем. Календарный план-график является основным документом, по которому контролируется процент выполнения ВКР.

ВКР является личной разработкой студента, руководитель лишь помогает советами и контролирует выполнение плана-графика. При необходимости руководитель делает замечания студенту по работе. Ответственность за достоверность информации, правильность расчетов и работоспособность системы несет студент.

Студент и руководитель встречаются для обсуждения решений задач проектирования не реже 1 раза в неделю.

Не позднее, чем за 10 дней до защиты ВКР выпускник подписывает расчетно-пояснительную записку у консультантов, отдает ее руководителю для ознакомления и написания отзыва, затем проходит нормоконтроль.

Для получения допуска, за 10 дней до защиты студент предоставляет на кафедру расчетно-пояснительную записку законченной работы, подписанную им самим, консультантами, руководителем и нормоконтролером. Расчетно-пояснительная записка должна содержать отзыв руководителя.

После получения допуска к защите студент передает расчетно-пояснительную записку на рецензирование. Рецензентами могут быть высококвалифицированные специалисты соответствующего профиля, работающие на предприятиях, в организациях, научно-

исследовательских и проектных институтах. В рецензии дается оценка актуальности темы работы, отмечаются достоинства и недостатки разработки, а также указывается практическое значение работы и качество оформления, степень использования выпускником современных источников, технологий, последних достижений науки и техники.

В заключительной части рецензии дается однозначная оценка (отлично, хорошо, удовлетворительно, неудовлетворительно) и указывается, заслуживает или не заслуживает выпускник присвоения квалификации бакалавр. Подпись рецензента должна быть заверена в кадровой службе по месту его работы. Бланк рецензии приведен в прил. 13.

За 2 дня до защиты ВКР расчетно-пояснительная записка вместе с отзывом руководителя и рецензией сдается на кафедру секретарю Государственной аттестационной комиссии (ГАК).

При подготовке к защите выпускнику рекомендуется составить тезисы доклада, оформить плакаты, а также продумать ответы на замечания рецензента.

Содержание тезисов доклада и информацию, выносимую на плакаты, студент согласовывает с руководителем.

Тезисы доклада обычно состоят из трех частей (вступление, основная часть, выводы). Во вступлении формулируется цель работы, подчеркивается ее актуальность, дается краткая характеристика существующих аналогов и показываются основные пути решения поставленной задачи. В основной части доклада дается описание основных этапов разработки системы и демонстрируются преимущества нового варианта по сравнению с уже существующими. В выводах дается рекомендация по его использованию.

На плакаты, как правило, выносятся информация, иллюстрирующая основные этапы разработки (структурная и/или функциональная схемы системы, диаграммы, алгоритмы, таблицы с результатами испытаний). В докладе обязательны ссылки на все представленные плакаты. Время доклада не должно превышать 10 минут.

6.4 Выпускная квалификационная работа специалиста

Выпускная квалификационная работа (ВКР) является завершающим этапом подготовки специалиста. Успешная защита ВКР служит основанием для присвоения выпускнику квалификации инженера по направлению подготовки дипломированного специалиста 654600 «Информатика и вычислительная техника» (специальность 230105 «Программное обеспечение вычислительной техники и автоматизированных систем»).

В процессе выполнения ВКР студент должен проявить самостоятельность, умение обосновывать принимаемые решения и способность решать поставленные перед ним научно-технические задачи. В этом ему помогут современные средства для разработки спецификаций: Erwin, BPwin и Rational Rose, которые описаны в разделе 9.

Для дипломированного специалиста предусмотрены два вида ВКР:

- дипломный проект,
- дипломная работа.

Перед студентами в процессе дипломного проектирования стоят следующие цели и задачи:

- систематизация и закрепление теоретических знаний по дисциплинам специальности, применение этих знаний при разработке программного обеспечения;
- развитие навыков самостоятельной работы с учебно-методической и научной литературой, законодательными и нормативными документами;
- овладение студентами методикой исследования, обобщения и логического изложения материала.

В ходе дипломного проектирования студент обязан:

- правильно применять теоретические положения естественно-научных, общепрофессиональных и специальных дисциплин;
- уметь самостоятельно выявлять и анализировать аналоги и прототипы разрабатываемого программного обеспечения;

– проявлять организаторские способности и навыки делового общения для преодоления реальных трудностей, возникающих при разработке и внедрении новых технических решений.

Структура выпускной квалификационной работы специалиста

ВКР содержит следующие разделы:

- титульный лист (образец в прил. 8),
- задание на ВКР,
- аннотацию,
- оглавление,
- введение,
- основную часть,
- технико-экономическое обоснование разработки,
- анализ мероприятий по безопасности жизнедеятельности и обеспечению экологической чистоты,
- заключение,
- список литературы,
- приложения,
- графический материал.

1. Титульный лист

Титульный лист должен содержать: наименование учебного заведения и структурного подразделения (институт, кафедра), в котором обучается студент, шифр и наименование специальности, название темы ВКР, Ф.И.О. студента и его подпись, согласование допуска к защите с директором института, зав. выпускающей кафедрой, руководителем и консультантами, нормоконтролером, название города и год окончания учебного заведения, где выполнялась работа.

2. Задание на ВКР

После утверждения темы студент совместно с руководителем составляет задание на выполнение ВКР в двух экземплярах на бланке, форма которого приведена в прил. 9. Один экземпляр студент прикладывает к расчетно-пояснительной записке, второй – ос-

тается на кафедре и является основанием для включения студента, темы ВКР, руководителя, рецензента и консультантов в приказ на дипломное проектирование. Задание подписывается студентом, руководителем, консультантами и утверждается заведующим кафедрой.

3. Аннотация

В аннотации в краткой форме отражаются цель и объект дипломного проектирования, полученные результаты и их новизна, особенности разрабатываемого программного продукта, перечень основных терминов и определений, количество иллюстраций, таблиц, приложений и использованных первоисточников. Объем аннотации – не более 1 страницы. Бланк аннотации представлен в прил. 14.

4. Оглавление

Оглавление включает в себя: введение, наименование всех разделов и подразделов, заключение, список литературы, перечень приложений с указанием номеров страниц. Пример оглавления приведен в прил. 17 (для структурного подхода) и в прил. 18 (для объектного подхода).

5. Введение

Во введении обосновывается актуальность разрабатываемого программного продукта, определяются цель работы и задачи для ее достижения, указываются используемые методы разработки и источники информации. Примерный объем введения 1-2 страницы.

6. Основная часть

Основная часть содержит, как правило, семь подразделов, которые могут иметь пункты (например, 1.5.3) и подпункты (например, 1.5.3.1). В них содержатся необходимые обоснования, математические модели, модульная структура системы, функциональные схемы, диаграммы, описания и др. Каждый подраздел заканчивается выводами, облегчающими понимание замысла, логики изложения и результатов исследования.

В подразделе *«Сравнительный анализ отечественных и зарубежных аналогов проектируемого объекта»* проводится обзор и анализ известных проектных решений по тематике ВКР с указани-

ем их достоинств и недостатков, делаются выводы о целесообразности и необходимости разработки.

В подразделе *«Выбор технологии, среды и языка программирования»* дается обоснование выбора модели жизненного цикла программного продукта, подхода к разработке, а также выбора инструментальных средств.

В подразделе *«Анализ и уточнение требований к программному продукту»* проводится анализ процесса обработки информации, выбор структур данных для ее хранения, разрабатываются методы и алгоритмы решения задачи. Данный подраздел уточняет требования к ПО, сформулированные в техническом задании.

В подразделе *«Разработка спецификаций программного обеспечения»* дается полное и точное описание функций и ограничений программного продукта, при этом используются схемы и/или диаграммы. Каждое графическое представление должно сопровождаться соответствующими подписями и пояснениями.

В подразделе *«Проектирование программного обеспечения»* приводятся детальные логическая и физическая модели разрабатываемого ПО (структурная и функциональная схемы, модульная структура ПО с описанием всех модулей системы с точки зрения их назначения, карты Константайна, диаграммы классов, алгоритмы функций (методов) и описание межмодульных интерфейсов). Пример заполнения спецификации на программное изделие приведен в прил. 20.

В подразделе *«Проектирование интерфейса пользователя»* дается описание разработки интерактивного графического интерфейса.

В подразделе *«Выбор стратегии тестирования, разработка тестов, программа и методика испытаний»* определяются тестируемые модули или подсистемы, окружение при тестировании (аппаратное и программное обеспечение и используемые ресурсы) и результаты тестирования, которые могут оформляться в виде табл. 6.3.

Таблица 6.3

Результаты тестирования

Дата и время тестирования	Тестируемый модуль и подсистемы	Кто проводил тестирование	Способ тестирования	Результаты тестирования
		<i>Разработчик или тестировщик</i>	<i>Автоматически или вручную</i>	<i>«Успех» или описание неполадки</i>

Таблица должна содержать описание всех тестов, их название, ответственного за тестирование, способ реализации теста, требование, проверяемое данным тестом, а также результат, полученный после выполнения теста. При необходимости в нее могут быть внесены дополнительные поля, а после таблицы – пояснения.

Наборы тестовых данных могут быть оформлены в виде табл. 6.4 (столбцы можно добавлять свои).

Таблица 6.4

Наборы тестовых данных

Тестовое значение	Реакция подсистемы	Пояснения

7. Техничко-экономическое обоснование разработки

Задание на технико-экономическое обоснование разработки (вторую часть) дипломного проекта (работы) выдается консультантом по экономике, назначаемым выпускающей кафедрой. Бланк задания по технико-экономическому обоснованию разработки приведен в прил. 10.

Примерный объем – 10 – 15 % от общего объема пояснительной записки.

В этом разделе дипломник рассчитывает следующие экономические показатели:

- себестоимость программного продукта;
- цену программного продукта;
- годовой экономический эффект;
- срок окупаемости.

Экономическая часть ВКР завершается выводами о целесообразности и экономической эффективности применения разработанного программного продукта. Подробная методика расчета технико-экономических показателей приведена в разделе 7.

8. Анализ мероприятий по безопасности жизнедеятельности и обеспечению экологической чистоты

Задание на эту (третью) часть дипломного проекта (работы) выдается консультантом по безопасности жизнедеятельности, назначаемым выпускающей кафедрой. Бланк задания приведен в прил. 11.

Примерный объем – 10 % от общего объема пояснительной записки. Во время преддипломной практики студент изучает вопросы промышленной санитарии и техники безопасности на рабочем месте, условия труда на данном производстве и др.

В этой части необходимо изложить следующее:

– анализ мероприятий по безопасности жизнедеятельности (общая характеристика безопасности производственного процесса; анализ вредных производственных факторов и мероприятия по их снижению; анализ опасных производственных факторов и мероприятия по их снижению; оценка электробезопасности помещения и мероприятия по повышению электробезопасности; оценка помещения по пожарной безопасности и мероприятия по повышению пожарной безопасности; анализ неполадок и аварийных ситуаций в рабочем помещении);

- анализ мероприятий по обеспечению экологической чистоты.

При написании данной части студент использует ГОСТы, СНиПы и НПБ.

Подробные рекомендации по анализу факторов безопасности жизнедеятельности приведены в разделе 8.

9. Заключение

В заключении приводятся теоретические и практические выводы и результаты работы. Они должны излагаться в краткой форме и давать представление о степени выполнения поставленной перед дипломником задачи. В заключении делается акцент на актуальность, степень новизны и практическую значимость разработанного программного продукта. Здесь же приводятся данные об экономической эффективности при внедрении данного программного продукта. Примерный объем заключения – 1-2 страницы.

Рекомендуемый объем дипломного проекта (работы) – 75-90 страниц.

10. Список литературы

Перечень литературы помещается непосредственно после основного текста работы. Библиографический список используемой литературы оформляется в соответствии с требованиями действующих ГОСТов.

11. Приложения

В приложения включают материал, не вошедший в пояснительную записку, но представляющий интерес для более глубокого понимания:

- 1 – техническое задание (ГОСТ 19.201-78),
- 2 – схемы и/или диаграммы (ГОСТ 19.701-90),
- 3 – текст программы (ГОСТ 19.401-78),
- 4 – спецификация (ГОСТ 19.202-78),
- 5 – руководство пользователя (ГОСТ 19.503-79, ГОСТ 19.504-79, ГОСТ 19.505-79),
- 6 – наборы тестовых данных и результатов тестирования (ГОСТ 19.301-79).

12. Графический материал

Конкретный графический материал зависит от темы ВКР и согласовывается с руководителем работы.

Плакаты, выносимые на защиту, выполняются на листе ватмана формата А1 (594 × 841) тушью или распечатываются на плоттере. Количество плакатов – 5 – 8.

Примерный перечень плакатов для структурного подхода.

1. Постановка задачи.

2. Логическая модель (функциональные диаграммы, диаграммы процессов, диаграммы потоков данных, диаграммы переходов состояний).

3. Диаграммы структур данных (диаграммы «сущность-связь»).

4. Физическая модель (структурная и/или функциональная схемы системы).

5. Схема взаимодействия программных модулей системы (структурные карты Константайна).

6. Алгоритмы функционирования основных модулей системы.

7. Результаты испытаний системы.

Примерный перечень плакатов для объектного подхода:

1. Постановка задачи.

2. Модель использования (диаграммы вариантов использования).

3. Концептуальная модель предметной области (контекстная диаграмма классов).

4. Модель процессов (диаграммы последовательностей системы, диаграммы деятельности).

5. Диаграммы пакетов, диаграммы классов пакетов.

6. Модель реализации и развертывания (диаграммы компонентов, диаграммы размещения).

7. Результаты испытаний системы.

Самостоятельная работа студента при написании и защите ВКР

Дипломный проект (работа) разрабатывается студентом очной пятилетней формы и очной сокращенной формы обучения самостоятельно. Студент составляет календарный план-график выполнения дипломного проекта (работы) с указанием очередности отдельных этапов и согласует его с руководителем. Календарный план-график является основным документом, по которому контролируется процент выполнения дипломного проекта (работы).

Дипломный проект (работа) является личной разработкой студента, руководитель лишь помогает советами и контролирует вы-

полнение плана-графика. При необходимости руководитель делает замечания студенту по дипломной работе. Ответственность за достоверность информации, правильность расчетов и работоспособность программного продукта несет студент.

Студент и руководитель встречаются для обсуждения решений задач дипломного проектирования не реже 1 раза в неделю.

Не позднее чем за 10 дней до защиты ВКР студент-дипломник подписывает расчетно-пояснительную записку у консультантов, отдает ее руководителю для ознакомления и написания отзыва, затем проходит нормоконтроль.

Для получения допуска за 10 дней до защиты студент-дипломник предоставляет на кафедру расчетно-пояснительную записку законченного дипломного проекта (работы), подписанную им самим, консультантами, руководителем и нормоконтролером. Расчетно-пояснительная записка должна содержать отзыв руководителя.

После получения допуска к защите студент-дипломник передает расчетно-пояснительную записку на рецензирование. Рецензентами могут быть высококвалифицированные специалисты соответствующего профиля, работающие на предприятиях, в организациях, научно-исследовательских и проектных институтах. В рецензии дается оценка актуальности темы дипломного проекта (работы), отмечаются достоинства и недостатки разработки, а также указывается практическое значение работы и качество оформления, степень использования выпускником современных источников, технологий, последних достижений науки и техники.

В заключительной части рецензии дается однозначная оценка (отлично, хорошо, удовлетворительно, неудовлетворительно) и указывается, заслуживает или не заслуживает выпускник присвоения квалификации инженер. Подпись рецензента должна быть заверена в кадровой службе по месту его работы. Бланк рецензии приведен в прил. 13.

За 2 дня до защиты ВКР расчетно-пояснительная записка вместе с отзывом руководителя и рецензией сдается на кафедру секретарю Государственной аттестационной комиссии (ГАК).

При подготовке к защите дипломнику рекомендуется составить тезисы доклада, оформить плакаты, а также продумать ответы на замечания рецензента.

Содержание тезисов доклада и информацию, выносимую на плакаты, дипломник согласовывает с руководителем.

Тезисы доклада обычно состоят из трех частей (вступление, основная часть, выводы). Во вступлении формулируется цель работы, подчеркивается ее актуальность, дается краткая характеристика существующих аналогов и показываются основные пути решения поставленной задачи. В основной части доклада дается описание основных этапов разработки программного обеспечения и демонстрируются преимущества нового варианта по сравнению с уже существующими. В выводах подчеркивается экономическая эффективность внедрения нового программного продукта, срок его окупаемости, а также дается рекомендация по его использованию.

На плакаты, как правило, выносятся информация, иллюстрирующая основные этапы разработки (структурная и/или функциональная схемы системы, диаграммы, алгоритмы, таблицы с результатами испытаний). В докладе обязательны ссылки на все представленные плакаты. Время доклада не должно превышать 12 минут.

6.5 Выпускная квалификационная работа магистра – магистерская диссертация

Написанием и защитой выпускной квалификационной работы (магистерской диссертации) магистрант завершает двухлетний период учебной, научно-исследовательской, педагогической и профессиональной деятельности.

Магистерская диссертация является самостоятельным научным исследованием, выполняемым под руководством научного руководителя. Ответственность за достоверность информации, правильность расчетов и работоспособность программного продукта несет магистрант.

Научный руководитель:

- оказывает магистранту помощь в правильной формулировке темы магистерской диссертации, подборе списка литературных источников и информации, необходимых для подготовки магистерской диссертации;

- проводит консультации с магистрантом (не реже 1 раза в две недели), оказывает ему необходимую методическую помощь при написании магистерской диссертации;

- содействует магистранту в разработке и контролирует ход выполнения индивидуального плана работы над диссертацией.

Перед магистрантами в процессе выполнения магистерской диссертации стоят следующие цели и задачи:

- развитие навыков ведения самостоятельной научной работы, исследования и экспериментирования, входящих в сферу будущей профессиональной деятельности магистра техники и технологии;

- систематизация и закрепление теоретических знаний по направлению, применение этих знаний при разработке программного обеспечения;

- развитие навыков самостоятельной работы с учебно-методической и научной литературой, законодательными и нормативными документами;

- овладение магистрантами методикой исследования, обобщения и логического изложения материала.

В ходе написания магистерской диссертации магистрант/выпускник обязан:

- правильно применять теоретические положения общепрофессиональных и специальных дисциплин;

- уметь самостоятельно выявлять аналоги и прототипы существующих систем;

- научно обосновывать выбор методов и алгоритмов решения данной задачи;

- проявлять организаторские способности и навыки делового общения для преодоления реальных трудностей, возникающих при разработке и внедрении новых технических решений.

Результаты исследования содержат совокупность научных положений и результатов, выдвигаемых автором для публичной защиты, имеют внутреннее единство, свидетельствуют о личном

вкладе и способности автора проводить самостоятельные научные исследования, используя теоретические знания и практические навыки.

Процесс выполнения магистерской диссертации состоит из следующих этапов:

1. Утверждение темы магистерской диссертации.
2. Подготовка магистерской диссертации.
3. Регистрация и допуск к защите магистерской диссертации.
4. Рецензирование магистерской диссертации.
5. Защита магистерской диссертации.

Темы магистерских диссертаций определяются выпускающей кафедрой с учетом современных тенденций развития технологии разработки программного обеспечения в России и за рубежом, предложений специалистов предприятий, а также на основе тематики научно-исследовательских работ, проводимых на профильных кафедрах. Название темы магистерской диссертации должно быть кратким, отражающим направление разработки, и в нем, как правило, указывается объект (предприятие, акционерное общество, банк и т.д.), где выполнялась работа.

Темы магистерских диссертаций доводятся до сведения магистрантов-выпускников, которые имеют право выбора или могут предложить свою тему с обоснованием целесообразности ее разработки.

За 10 дней до защиты магистерской диссертации магистрант подписывает два экземпляра ВКР у нормоконтролера, отдает научному руководителю для ознакомления и написания отзыва, а также проходит предзащиту на выпускающей кафедре с целью получения допуска к основной защите.

После получения допуска к защите магистрант передает ВКР на рецензирование. Рецензентами могут быть высококвалифицированные специалисты соответствующего профиля, работающие на предприятиях, в организациях, в научно-исследовательских и проектных институтах.

За 2 дня до защиты два экземпляра ВКР вместе с отзывом руководителя и рецензией сдается на кафедру секретарю ГАК.

При подготовке к защите магистранту рекомендуется составить тезисы доклада, оформить слайды и плакаты, а также продумать ответы на замечания рецензента.

Содержание тезисов доклада и информацию, выносимую на слайды и плакаты, магистрант согласовывает с руководителем.

Тезисы доклада обычно состоят из трех частей (вступление, основная часть, выводы). Во вступлении формулируется цель работы, подчеркивается ее актуальность, дается краткая характеристика существующих аналогов и показываются основные пути решения поставленной задачи. В основной части доклада дается описание предлагаемого метода и алгоритмов решения поставленной научной задачи, кратко формулируются основные этапы научного исследования и демонстрируются преимущества нового варианта решения по сравнению с уже существующими. В выводах подчеркивается эффективность предлагаемого метода, а также дается рекомендация по его использованию.

На слайды, как правило, выносятся информация, иллюстрирующая этапы решения научной задачи. В докладе обязательны ссылки на все представленные слайды и плакаты. Время доклада не должно превышать 15 минут.

Структура магистерской диссертации

Магистерская диссертация, помимо текстовой части, может содержать демонстрационные материалы в виде чертежей, плакатов, моделей, макетов, рисунков, компьютерных программ и т.д.

В тексте диссертации магистра активно применяется **метатекст**. Метатекст – это отрезки, фрагменты текста, в которых автор текста объясняет, мотивирует построение текста, его содержание, логику изложения. Метатекст – это несколько групп вводных слов, таких как «во-первых», «во-вторых», «таким образом», «иначе говоря», «с другой стороны» и т.п. Также к метатексту относятся некоторые предлоги, такие как «насчет», «что касается». Метатекст также может выражаться частями сложных предложений, например: «Далее отметим, что...», «Обратим внимание на тот факт, что...». Каждая часть научной работы начинается с метатекстовой конструкции ви-

да «В данной главе/данном разделе мы рассмотрим такие вопросы, как...».

Метатекстовый компонент обязателен в научных работах как во введении, где объясняется выбор темы и объясняется композиция будущей работы, так и в последующих разделах.

Магистерская диссертация должна включать в себя следующие структурные элементы:

- титульный лист;
- аннотацию;
- оглавление;
- введение;
- разделы, структурированные на параграфы;
- заключение;
- список литературы;
- приложения (если это необходимо);
- графический материал.

Примерное содержание магистерской диссертации приведено в прил. 19.

1. Титульный лист

Титульный лист должен содержать: наименование учебного заведения и структурного подразделения (институт, кафедра), в котором обучается магистрант, шифр и наименование направления, наименование магистерской программы, название темы магистерской диссертации, Ф.И.О. магистранта, согласование допуска к защите с директором института, зав. выпускающей кафедрой, руководителем магистерской программы, научным руководителем, нормоконтролером, название города и год окончания учебного заведения, где выполнялась работа.

Форма титульного листа представлена в прил. 8.

2. Аннотация

В аннотации в краткой форме отражаются цель и объект научного исследования, полученные результаты и их новизна, особенности разрабатываемого программного продукта, перечень основных терминов и определений, количество иллюстраций, таблиц, приложений и использованных первоисточников. Объем аннотации – не

более 1 страницы. Бланк оформления аннотации представлен в прил. 14.

3. Оглавление

Разделы магистерской диссертации оформляются в виде отдельных частей, содержание которых определяется целью и задачами магистерской диссертации. Оглавление включает в себя: введение, наименование всех разделов и подразделов, заключение, список литературы, перечень приложений с указанием номеров страниц.

4. Введение

Введение содержит обоснование актуальности темы исследования, характеристику методологического инструментария: объекта, предмета, цели, задач, гипотезы, источниковедческой базы и методов исследования, характеристику научной/научно-методической новизны и практической значимости исследования. Примерный объем введения 4 – 6 страниц.

При доказательстве **актуальности** исследования можно выделить следующие ее аспекты:

- значение важности проблемы (задачи), решаемой в диссертации для теории и практики соответствующей отрасли науки (знаний) и техники, производства, проектирования и т.п.;
- направленность на решение ключевых проблем и важнейших задач разработки новых или повышения качественных характеристик известных объектов, на выполнение целевых комплексных и отраслевых программ, перспективных и годовых планов;
- обобщение опыта отечественной и зарубежной практики, учет перспектив развития, влияние на проблемы (задачи) различных факторов (технических, экономических и т.п.). Кратко, в несколько абзацев, аннотируются результаты обзора литературы. Указывается, кто из ученых заложил основы данного направления науки (2-5 чел.). Перечисляются основные специалисты, публиковавшие свои результаты в последнее время (5-10 чел.);
- нерешенные вопросы (теоретического и практического плана), сдерживающие научно-технический и социальный прогресс. Указывается, почему описанные в литературе исследования и полученные предыдущими исследователями научные результаты не

позволяют решить рассматриваемую проблему. Как правило, это следствие новизны задачи, изменения экономической обстановки, появление новых методов исследования.

Объект исследования – та часть материального мира, которой занимается исследователь, например, автоматизированные системы, информационные системы, системы управления базами данных, информационные и компьютерные сети, интерактивные графические системы, пользовательский интерфейс, методы обработки информации и управления и т.д. По отношению к объекту исследования соискатель должен ответить на вопрос: объект исследования *традиционный* или *новый*. Например, автоматизированная система – традиционный (когда-то он был новым), а интерактивные графические системы – новый объект исследования. В свою очередь, интерактивные графические системы – может рассматриваться как традиционный объект, а трехмерное моделирование и анимация – как новый.

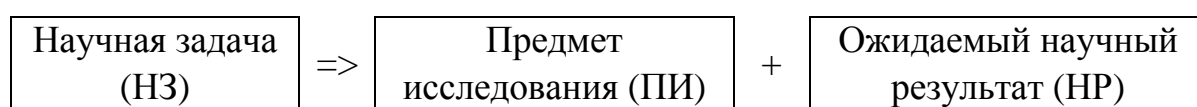
Предмет исследования – это та сторона объекта исследования, те его свойства, та его область приложения, которые исследуются, например, точность вычислений, оптимизация вычислительных процессов, управление ресурсами компьютера. По отношению к предмету исследования решается тот же вопрос: предмет исследования *новый* или *традиционный*. Например, структурирование данных – традиционный предмет исследования, а структурирование данных с учетом их иерархической организации – новый предмет исследования.

Цель исследования – это планируемый конечный результат, т.е. устранение уязвимого звена в общей цепи (повысить, улучшить, обеспечить качество и/или эффективность и т. п.). Цель должна быть конкретной, т.е. при ее достижении будет производиться программное обеспечение, создаваться интерфейс пользователя и т.п. с лучшими показателями, чем было раньше. По установленным правилам подготовки диссертаций *цель* всегда формулируется и трактуется как *практическая*. Именно та практическая потребность, во имя которой осуществляется решение актуальной научной задачи (для магистерской диссертации), должна отражать цель диссертационного исследования. Иногда при формулировании

цели указывают области и способы использования научного результата(ов).

Научная задача для магистерской диссертации – это то, что надо решить, причем возможно, что одно из решений или один из методов решения известны. Требуется найти новое решение научной задачи, которое бы отвечало новым условиям и новым целям, либо новым условиям для достижения “старой” цели.

Несколько условно научную задачу можно выразить в виде пары суждений (слов), которые включают предмет исследования и ожидаемый (требуемый) научный результат.



Формулировка научной задачи, конкретизирующая предмет исследования и ожидаемый (требуемый) научный результат, называется постановкой научной задачи.

Научная задача ставится с учетом всех допущений, ограничений, предпосылок и т.п.

Решение научной задачи как процесс самого исследования включает в себя ее постановку и определение методов исследования (способа достижения поставленной цели).

Новое решение научной задачи будет получено тогда, когда:

- расширен, уточнен, лишен каких-либо ограничений предмет исследования;
- поставлены новые цели исследования, выражаемые через требуемые научные результаты (что изменило глубину исследования, его масштаб и область практических приложений);
- применен новый метод исследования;
- усовершенствован, развит, приспособлен для данных целей один из известных методов исследования с гарантией или доказательством достоверности;
- показано и доказано преимущество предлагаемого метода исследований.

Методы исследования. Для решения поставленных задач магистерского диссертационного исследования может быть использован

аппарат теории вероятностей и математической статистики, теории алгоритмов и программ и т.д.

Научные результаты – это такие научные положения, которые являются итогом научной разработки и удовлетворяют требованиям новизны, достоверности и практической ценности. Они подразделяются на теоретико-методологические, характерные для теоретических работ и инструментальные, характерные для прикладных, экспериментальных и эмпирических исследований.

Виды результатов фундаментальных и теоретических работ:

- Закон, закономерность, явление.
- Гипотезы, принципы, правила.
- Понятия, категории.
- Теория, способы.
- Теоретическое, теоретико-экспериментальное, математическое обоснование, доказательство.
- Научно-методический (методический, математический) аппарат (подход, метод, методика).
- Формализованное (математическое) описание, математическая модель.
- Математические предложения (соотношения), аксиома, теорема, лемма, формула (формульное соотношение), математическая зависимость.
- Методика исследования или прогнозирования.

Виды прикладных результатов:

- Научно-техническое обоснование требований, облика, задач, характеристик, исходных данных и т.п.).
- Программы общего, системного и специального математического обеспечения.
- Модели (математические, физические, имитационные, экспериментальные, описательные).
- Методики (расчета, анализа, синтеза, конструирования и т.д.), методические указания.
- Алгоритмы.
- Технологические процессы или технологическое оборудование.

– Учебно-тренировочные, учебно-лабораторные, моделирующие комплексы, системы, средства (установки), стенды, тренажеры и документация на них.

– Макетные и экспериментальные образцы систем, установок, узлов, приборов.

Научный результат – продукт исследовательской деятельности, полученный с помощью научных методов (анализа, обоснования, прогнозирования и др.), т.е. на основе применения того или иного научно-методического аппарата.

Научно-методический аппарат – это совокупность разработанных в ходе развития соответствующей отрасли знаний и широко используемых методов, приемов, способов и средств получения научных результатов. Он апробирован на практике, прошел экспертизу специалистов и обладает высокой достоверностью. Обычно научно-методический аппарат включает в себя средства получения результатов и средства обоснования (проверки, доказательства или оценки) их достоверности или точности.

Новый научный результат может быть получен несколькими путями. Первый путь – *создание новой, более эффективной методики исследования*, например, разработка нового метода. Второй путь – *применение известного математического аппарата в новой области*. По этому пути проводятся исследования, когда новая задача или проблема не может быть решена с помощью существующих методик или эти методики не обеспечивают необходимой точности результата. Третий путь – *усовершенствование известной методики исследования дополнением используемого в ней научно-методического аппарата новыми элементами*, приводящими к получению положительного эффекта (например, повышение точности, уменьшение объема памяти, ускорение вычислений и т.п., отражение в математической модели дополнительных, ранее не учитываемых факторов для новых исследуемых условий, введение дополнительных ограничений или допущений, либо снятие ранее сделанных и т.д.).

В качестве новых научных результатов рассматривается и сама новая или усовершенствованная методика исследования (элементы

методики) и то, что с ее помощью получено (результаты, новые выводы, рекомендации, эффекты).

Практическая ценность (значимость) магистерской диссертации определяется наличием практических рекомендаций, широтой области практического приложения, внедрением полученных результатов или степенью и масштабом возможного их внедрения, оценкой технико-экономической эффективности.

Показателями оценки практической ценности могут служить:

- использование или возможность использования конкретных научных результатов в создаваемых образцах с отражением возможных эффектов;
- обоснованные выводы и рекомендации;
- конкретные инженерные методы, методики, технические предложения, справочники, наставления, руководства и т.п.;
- использование результатов в учебном процессе.

В диссертационных работах практическая ценность подтверждается обычно документами о реализации (внедрении) и/или возможном использовании научных результатов.

В конце введения приводится список конференций, на которых докладывались основные результаты исследования, а также количество публикаций по теме исследования.

5. Содержание разделов

Разделы отражают результаты теоретико-методологического анализа проблемы, содержат характеристику методологических позиций исследователя; результаты сопоставительного и/или ретроспективного анализа исследований по проблеме; интерпретацию ключевых понятий по теме исследования; теоретическое обоснование и описание построения системы экспериментальной части исследования, а также анализ, обобщение результатов основных этапов эксперимента.

В *первом разделе*, как правило, проводится обзор и анализ известных проектных решений по тематике магистерской диссертации с указанием их достоинств и недостатков, делаются выводы о целесообразности и необходимости разработки, дается обоснование выбора метода решения поставленной задачи, исходя из простоты реализации, надежности, доступности и т.д.

В отдельных случаях дается некое общее вступление к главе, используя метатекст. Например: *«Изучением различных способов ... занимались многие российские и зарубежные ученые. В данной главе будут рассмотрены основные вехи теоретических разработок в этой области...»*

Поиск литературы проводится:

- в библиотеке университета, в том числе и электронной;
- в библиотечной сети учреждений науки и образования Северо-Западного региона России RUSLANet <http://www.ruslan.ru:8001/rus/>;
- в библиотеках ведущих предприятий города и региона;
- по базам данных ГПНТБ РФ <http://www.gpntb.ru/win/search/>, в том числе по БД «Авторефераты диссертаций»;
- по базе диссертаций <http://db2.rulib.com/cgi-bin/search.pl> (в параметрах поиска указывайте шифр специальности).

В составе источников обязательно должны быть монографии, журнальные статьи и патенты, рефераты статей и патентов, в том числе на иностранных языках.

Затем приводится обзор методов решения научной задачи. Он должен присутствовать, если в работе используются нестандартные методы исследования (и разработанные не самим автором, а заимствованные). Необходимо описать эти методы, указать их источники, а также обосновать их выбор.

В выводах по первому разделу формулируется практическая проблема, подлежащая решению и перечень нерешенных задач, которые берет на себя автор.

Один из возможных вариантов плана первого раздела:

- общая характеристика данной области и предмета исследования, значение этой области в научном и практическом планах, актуальность задач, стоящих перед этой областью;
- классификация основных направлений исследований в данной области, практически используемые направления исследований и направления, находящиеся в процессе разработки, отражение разных точек зрения на решение проблемы;
- подробное изложение результатов существующих исследований по каждому разделу, разработанной автором классификации

(для теоретических работ описываются используемые методики, применяемый математический аппарат, а для экспериментальных – важнейшие схемы, модели и результаты эксперимента); необходим критический анализ этих материалов с конкретными предложениями;

- выводы из предыдущих исследований и перечисление основных нерешенных вопросов;

- переход к формулированию основных направлений своих исследований, их актуальность, основная и частные задачи и конечная цель.

Во *втором разделе*, как правило, подробно описывается предлагаемый метод решения научной задачи, приводятся математическая модель и алгоритмы обработки информации в разрабатываемой системе

Функция раздела – дать теорию вопроса в общем с модификацией, приближающей ее к задачам исследования. Указывается вклад диссертанта в подготовку теоретической базы исследования: установление обоснованных коэффициентов согласования, введение новых членов в уравнения математической модели или дополнительных уравнений, отражающих физику анализируемого процесса, новых обнаруженных факторов, особенностей протекания явления.

Также рекомендуется перед первым параграфом дать метатекстовые конструкции, предваряющие содержание главы и обеспечивающие связь с первым разделом: «Определив основные задачи исследования, перейдем к теоретическому анализу...»; «В предыдущем разделе мы отметили, что ... недостаточно разработаны. Во втором разделе будут рассмотрены теоретические основы ...».

Если в процессе работы проводились экспериментальные исследования, адекватность модели определяется соответствием теории практике и анализируется в четвертом разделе.

Раздел должен заканчиваться краткими выводами (0.5 – 1.0 страница), в которых содержатся:

- формулировка новизны результата;
- обоснование достоверности результата;
- доказательства практической ценности результата.

В *третьем разделе* описываются детальные логическая и физическая модели разрабатываемого информационного и программного обеспечения (структурная и функциональная схемы, модульная структура ПО с описанием всех модулей системы с точки зрения их назначения, карты Константайна, диаграммы классов, алгоритмы функций (методов) и описание межмодульных интерфейсов), разработка интерактивного графического интерфейса.

Раздел должен заканчиваться краткими выводами, в которых содержатся:

- краткая суть результатов моделирования с цифрами и фактами;
- оценка качества технологичности разработанного информационного и программного обеспечения;
- характеристика человеко-машинного взаимодействия.

В *четвертом разделе* излагается методика и результаты экспериментальных исследований, сопоставляются данные теории и эксперимента, приводятся оригинальные разработки программного обеспечения, описываются экспериментальные установки, их аппаратно-программное обеспечение, модели программно-информационных систем и т.п. Показывается новизна предложенных автором решений, практическая ценность, предлагаются формы реализации (внедрения).

Выводы из четвертого раздела должны содержать:

- суть эксперимента (цель, условия и т.п.);
- полученные результаты (факты и цифры);
- анализ соответствия теоретических и экспериментальных данных;
- характеристику новизны полученного результата;
- практическую ценность полученных результатов.

6. Заключение

Заключение начинается с общего вывода о решении поставленной задачи и включает 4 – 6 пунктов, в которых перечисляются основные научные и практические результаты с акцентом новизны и достоверности и доказывается обоснованность общего вывода.

При формулировании пунктов пользуются краткими формами причастий страдательного залога – доказана, доказано, получены, показано что, предложен и т.п..

В заключении должны быть не только теоретические выводы, но и цифровые данные, характеризующие принципиальные итоги работы и доказывающие реализуемость и результативность рекомендаций автора.

В заключении также указываются перспективы дальнейших исследований и нерешенные вопросы.

Выводы бывают трех типов.

1. Вывод типа *«показано, что...»*. Такой вывод говорит об установленной автором в результате исследования зависимости одних параметров, характеризующих объект исследования, от других его параметров. Таких выводов должно быть от 2 до 4.

2. Вывод типа *«предложено использовать установленные зависимости...»* говорит о реализации выявленных зависимостей в технических, экономических или технологических разработках автора. Таких выводов должно быть от 1 до 3.

3. Вывод типа *«внедрено»* говорит об использовании (внедрении) разработок автора различными организациями (юридическими лицами). Наименование организаций должно включать в себя указание их организационно-правовой формы. Факт использования (внедрения) подтверждается официальными актами, справками.

Выводы должны соответствовать задачам исследования, указанным в первом разделе.

Примерный объем заключения – 1 – 2 страницы.

7. Список литературы

Перечень литературы помещается непосредственно после основного текста работы. Библиографический список используемой литературы оформляется в соответствии с требованиями действующих ГОСТов.

8. Приложения

Приложения приводятся с целью освобождения основного текста работы от излишней затрудняющей восприятие детализации. В приложения выносятся тексты анкет, таблицы, расчетные формулы, справочные данные, планы-конспекты конкретных мероприятий, копии актов о внедрении результатов и т.д.

Приложения не входят в рекомендуемый объем диссертации.

9. Графический материал

Конкретный графический материал зависит от темы магистерской диссертации и согласовывается с руководителем работы.

Плакаты, выносимые на защиту, выполняются на листе ватмана формата А1 (594х841) тушью или распечатываются на плоттере. Количество плакатов – 5 – 8.

Примерный перечень плакатов:

1. Постановка задачи.
2. Математическая модель.
3. Логическая модель.
4. Физическая модель.
5. Отдельные алгоритмы решения задачи.
6. Структурная и (или) функциональная схемы системы.
7. Модульная структура программного обеспечения.
8. Результаты испытаний.

Требования к оформлению магистерской диссертации

Требования к оформлению магистерской диссертации следующие:

- магистерская диссертация представляется в напечатанном на компьютере виде в сшитом файле (либо в папке) в двух экземплярах, а также в электронном виде. Рекомендуемый объем магистерской диссертации 75 – 90 машинописных страниц формата А4;

- первый лист магистерской диссертации – титульный; остальные листы нумеруются внизу страницы арабскими цифрами; нумерация сплошная, включая титульный лист, при этом на титульном листе номер страницы не проставляется; сноски должны иметь постраничную нумерацию;

- размер шрифта основного текста – 14, размер шрифта сносок – 10. Межстрочный интервал – 1,5. Поля: верхнее и нижнее – 2 см, правое – 1,5 см, левое – 3 см;

- каждый раздел (введение, глава, параграф, список использованной литературы и т.д.) должен начинаться с новой страницы; заголовки разделов следует располагать в середине строки без точки в конце и печатать заглавными буквами жирным шрифтом;

- магистрант обязан давать ссылки на автора и источник, откуда он заимствует материалы; в случае использования заимствованного материала без ссылки на автора и источник заимствования магистерская диссертация к защите не допускается. Ссылки на использованные источники приводятся в виде номера источника в списке. Номер (номера) приводится в квадратных скобках, например [24], [25] – [28], [34], [37]. Особо важные ссылки приводятся с указанием автора, например, И.И. Иванов [38], D. Dzentshpshkevich [39]. Пробел между фамилией и инициалами не ставится. В тексте работы должны присутствовать ссылки на все источники, приведенные в списке;

- список использованной литературы печатается в алфавитном порядке с указанием источников опубликования.

Общие правила оформления текстовой и графических материалов рассмотрены в разделе 10.

МЕТОДИКА РАСЧЕТА ТЕХНИКО-ЭКОНОМИЧЕСКИХ ПОКАЗАТЕЛЕЙ

Любое техническое или программное решение предполагает вложение финансовых средств, отсюда возникает вопрос рационального их применения и выбора из многообразных способов решения задачи наиболее эффективного. Эффективность – широкое социально-экономическое понятие, предполагающее наличие положительных результатов, полезного эффекта, отдачи от всех средств, вложенных в разработку и внедрение проекта.

Существует несколько видов эффективности, среди которых обычно выделяют:

- коммерческую (финансовую) эффективность, учитывающую финансовые последствия реализации проекта для всех участников проекта;
- бюджетную эффективность, отражающую влияние последствий внедренного проекта на бюджет предприятия, отрасли или региона;
- экономическую эффективность, учитывающую затраты и результаты, связанные с реализацией проекта.

Экономическую эффективность (как и другие виды человеческой деятельности) в самом общем виде принято определять как отношение планируемых результатов производства к затратам. Экономический эффект выступает как показатель результата производства, как достижение ранее поставленной цели. Сущность экономического эффекта заключается в создании материальных благ, удовлетворяющих соответствующие потребности и в экономии всех производственных ресурсов (живого труда, материалов, капитальных вложений).

Эффективность – это мера соотношения затрат и результатов функционирования программного средства.

Экономический эффект – это результат внедрения некоторого мероприятия, выраженный в стоимостной форме в виде экономии от его осуществления.

Коэффициент эффективности капитальных вложений – это величина годового прироста прибыли, образующегося в результате производства или эксплуатации программного средства на 1 рубль единоразовых капитальных вложений.

Срок окупаемости – это величина, обратная коэффициенту эффективности. Представляет собой период времени, в течение которого затраты на программное средство окупятся полученным эффектом.

В процессе разработки программного средства в качестве критериев экономической эффективности могут быть выбраны следующие критерии:

1) максимальная экономическая эффективность функционирования программного средства за весь период жизненного цикла при ограниченных затратах на разработку программ;

2) минимальные затраты на разработку программ при заданной экономической эффективности применения и заданном качестве программного средства;

3) максимальное отношение экономической эффективности применения программного средства в течение времени эксплуатации к затратам на его создание;

4) максимальная разность эффекта от функционирования программного средства за весь жизненный цикл и затрат на его разработку, эксплуатацию и сопровождение.

Расчет экономической эффективности и срока окупаемости проектируемой программы реализованной на ЭВМ начинается с расчета трудовых затрат.

7.1 Трудозатраты на разработку и отладку программы

Процесс программирования необходимо планировать, контролировать и завершать в срок. Кроме того, труд программистов должен оплачиваться по результатам их труда (его качеству, количеству и интенсивности выполнения работ). Для этих целей определяют затраты труда (трудоемкость) на разработку программного продукта.

Для определения затрат труда на разработку используются аналитические и экспертные методы оценок. Выбор метода осуществляется в зависимости от степени учета факторов, влияющих на трудоемкость разработки. Для более точного определения трудоемкости разработки программного продукта по отдельным элементам и операциям процесса может быть использован метод, в котором составляющие затраты труда определяются с учетом особенностей организации, ведущей разработку, и основных параметров программного продукта: степени новизны задачи, сложности алгоритма, количества разновидностей входной и выходной информации, сложности организации контроля входной и выходной информации, языка программирования, использования стандартных модулей и типовых задач.

Трудоемкость разработки программного обеспечения решения задачи можно рассчитать по формуле:

$$t = t_{и} + t_{а} + t_{п} + t_{отл} + t_{д},$$

где $t_{и}$ – затраты труда на изучение описания задачи; $t_{а}$ – затраты труда на разработку алгоритма решения задачи; $t_{п}$ – затраты труда на программирование по готовой блок-схеме; $t_{отл}$ – затраты труда на отладку программы на ЭВМ; $t_{д}$ – затраты труда на подготовку документации.

Составляющие затрат можно определить через условное число операторов в разрабатываемом программном обеспечении. В их число входят те операторы, которые необходимо написать программисту в процессе работы над задачей с учетом возможных уточнений в постановке задачи и совершенствования алгоритма. Условное число операторов Q в программе может быть определено по формуле:

$$Q = q \cdot c \cdot (1 + p),$$

где q – предполагаемое число операторов в программе; c – коэффициент сложности программы по отношению к типовой задаче (величина c лежит в пределах от 1,25 до 2); p – коэффициент коррек-

ции программы в ходе ее разработки, увеличение объема работ за счет внесения изменений в алгоритм или программу по результатам уточнения постановок и описаний ее, изменения состава и структуры информации, а также уточнений, вносимых разработчиками для улучшения качества самой программы без изменения постановки задачи (величина p находится в пределах 0,05...0,1).

Кроме этого, используются коэффициенты: квалификации разработчиков алгоритмов и программ – k , отражающий степень подготовленности исполнителя к порученной ему работе (он определяется в зависимости от стажа работы и составляет: для работающих до двух лет – 0,8; от двух до трех лет – 1,0; от трех до пяти лет – 1,1 – 1,2; от пяти до семи – 1,3 – 1,4; свыше семи лет – 1,5 – 1,6), и увеличения затрат труда вследствие недостаточного или некачественного описания задачи – B , оценивающий качество постановки задачи, выданной для разработки, в связи с тем, что задачи, как правило, требуют уточнения и некоторой доработки (этот коэффициент в зависимости от сложности задачи принимается от 1,2 до 1,5).

Затраты труда на изучение описания задачи t_u с учетом уточнения описания и квалификации программиста могут быть определены по формуле, чел./ч:

$$t_u = Q \cdot B / (76 \cdot k),$$

где k – коэффициент квалификации разработчиков; B – коэффициент увеличения затрат труда вследствие недостаточного описания задачи.

Затраты труда на разработку алгоритма решения задачи t_a рассчитываются по формуле, чел./ч.:

$$t_a = Q / (22 \cdot k).$$

Затраты труда на программирование по готовой блок-схеме t_n определяются по формуле, чел./ч.:

$$t_n = Q / (23 \cdot k).$$

Затраты труда на отладку программы на ЭВМ $t_{отл}$ рассчитываются по следующим формулам, чел./ч.

$$t_{отл} = Q / (4 \cdot k) \text{ – при автономной отладке одной задачи,}$$
$$t_{котл} = 1,5 \cdot t_{отл} \text{ – при комплексной отладке.}$$

Затраты труда на подготовку документации:

$$t_d = t_{др} + t_{до},$$

где $t_{др} = Q / (17 \cdot k)$ – затраты труда на подготовку материалов в рукописи; $t_{до} = 0,75 \cdot t_{др}$ – затраты труда на редактирование, печать и оформление документации.

Пример расчета трудоемкости разработки

Предположим, что наш программный продукт относится к ряду оригинальных разработок. Для разработки выбран язык программирования C++, операционная система Windows 2000. Необходимые средства вычислительной техники – компьютер Pentium IV – 2200 / DDR 256 Mb / 160 Gb / CD-RW, принтер LaserShot LBP-1120.

Исходные данные для расчета определяются с учетом сложности разработки, наличия аналогов, квалификации разработчиков:

- предполагаемое число операторов $q = 15\,000$;
- коэффициент сложности разработки $c = 1,5$;
- коэффициент коррекции программы в ходе ее разработки $p = 0,07$;
- коэффициент квалификации разработчиков $k = 0,8$;
- коэффициент увеличения затрат труда вследствие недостатков описания задачи $B = 1,3$.

Условное число операторов Q в программе определяется по формуле

$$Q = q \cdot c \cdot (1 + p) = 15000 \cdot 1,5 \cdot (1 + 0,07) = 24\,075.$$

Затраты труда на изучение описания задачи $t_{и}$ (с учетом квалификации программиста):

$$t_{и} = Q \cdot B / (76 \cdot k) = 24075 \cdot 1,3 / (76 \cdot 0,8) = 515.$$

Затраты труда на разработку алгоритма решения задачи t_a :

$$t_a = Q / (22 \cdot k) = 24\,075 / (22 \cdot 0,8) = 1367,9.$$

Затраты труда на программирование по готовой блок-схеме:

$$t_{п} = Q / (23 \cdot k) = 24075 / (23 \cdot 0,8) = 1308,4.$$

Затраты труда на отладку программы на ЭВМ при автономной отладке одной задачи:

$$t_{отл} = Q / (4 \cdot k) = 24\,075 / (4 \cdot 0,8) = 7\,523,4;$$

при комплексной отладке задачи:

$$t_{котл} = 1,5 \cdot t_{отл} = 7\,523,4 \cdot 1,5 = 11\,285.$$

Затраты труда на подготовку документации по задаче

$$t_d = t_{др} + t_{до}$$

Затраты труда на подготовку материалов в рукописи:

$$t_{др} = Q / (17 \cdot k) = 24\,075 / (17 \cdot 0,8) = 1\,770,2.$$

Затраты труда на редактирование, печать и оформление документации:

$$t_{до} = 0,75 \cdot t_{др} = 0,75 \cdot 1\,770,2 = 1\,327,7;$$

$$t_d = 1\,770,2 + 1\,327,7 = 3\,097,9.$$

Суммарная трудоемкость разработки составит, чел./ч.:

$$t = 515 + 1367,9 + 1308,4 + 7\,523,4 + 3\,097,9 = 13812,6.$$

7.2 Трудозатраты на разработку и отладку программы

В себестоимость разработки входят: основная заработная плата разработчиков, дополнительная заработная плата, единый социальный налог, затраты на использование машинного времени, затраты на носители информации, затраты на текущий и профилактический ремонт вычислительной техники, прочие эксплуатационные расходы.

$$C_{пп} = Z_{тр} + Z_{м.вр} + Z_{н.и} + Z_{рем} + Z_{пр},$$

где $C_{пп}$ – себестоимость программного продукта, руб.; $Z_{тр}$ – затраты на оплату труда, руб.; $Z_{м.вр}$ – затраты на использование машинного времени, руб.; $Z_{н.и}$ – затраты на носители информации, руб.; $Z_{рем}$ – затраты на текущий и профилактический ремонт вычислительной техники, руб.; $Z_{пр}$ – прочие эксплуатационные расходы, руб.

Примерный состав разработчиков: программист-дипломник, руководитель ВКР.

Затраты на оплату труда при разработке программного продукта вычисляются по формуле:

$$Z_{тр} = (Z_{общ} + Отч) * T_n, \text{ – с окладом}$$

$$Z_{тр} = Z_{общ} * (1 + Отч) * T_n, \text{ – почасовая}$$

где $Z_{общ}$ – общая зарплата работника за час; Отч – отчисления с зарплаты, %; T_n – время написания программы.

Время написания программы T_n совпадает с временем работы компьютера.

Зарботная плата программиста за час определяется по следующей формуле:

$$З_{\text{пр}} = \frac{С_{\text{Т пр}}}{\Phi_{\text{вм}}},$$

где $С_{\text{Т пр}}$ – ставка программиста; $\Phi_{\text{вм}}$ – фонд рабочего времени в месяц, ч.

Заработная плата дополнительная определяется по следующей формуле:

$$З_{\text{доп}} = \frac{З_{\text{пр}} \cdot Н_{\text{доп}}}{100\%},$$

где $З_{\text{пр}}$ – заработная плата программиста; $Н_{\text{доп}}$ – норма отчислений на дополнительную зарплату (10 %).

Зарплата общая вычисляется по следующей формуле:

$$З_{\text{общ}} = З_{\text{пр}} + З_{\text{доп}},$$

Отчисления на соцстрах, фонд занятости и пенсионный фонд вычисляются по следующей формуле:

$$\text{Отч} = O_{\text{сс}} + O_{\text{фз}} + O_{\text{пф}},$$

где $O_{\text{сс}}$ – отчисления на соцстрах (0,5 % от $З_{\text{общ}}$); $O_{\text{фз}}$ – отчисления в фонд занятости (0,5% от $З_{\text{общ}}$); $O_{\text{пф}}$ – отчисления в пенсионный фонд (2 % от $З_{\text{общ}}$).

Все данные по заработной плате сводятся в табл. 7.1.

Таблица 7.1

Данные по заработной плате

Должность разработчика	Разряд	Время работы, мес.	$С_{\text{Т пр}}$, руб.	$З_{\text{пр}}$, руб.	$З_{\text{доп}}$, руб.	$З_{\text{общ}}$, руб.	Отч, руб.	$З_{\text{тр}}$, руб.
1								
...								
Итого:								

Затраты на использование машинного времени вычисляются по формуле:

$$З_{\text{м.вр}} = C_{\text{м.вр}} \cdot Вр_{\text{в.т}},$$

где $З_{\text{м.вр}}$ – затраты на использование машинного времени, руб.; $C_{\text{м.вр}}$ – стоимость одного часа машинного времени, руб./ч; $Вр_{\text{в.т}}$ – время использования вычислительной техники, ч.

Стоимость одного часа машинного времени рассчитывается по формуле:

$$C_{\text{м.вр}} = \frac{Ц_{\text{к}}}{C_{\text{сл.к}} \cdot K_{\text{р.д}} \cdot Вр_{\text{с}}} + C_{\text{т.э}} \cdot M_{\text{вс}},$$

где $C_{\text{м.вр}}$ – стоимость одного часа машинного времени, руб./ч; $Ц_{\text{к}}$ – покупная цена компьютера, руб.; $C_{\text{сл.к}}$ – срок службы компьютера, год; $K_{\text{р.д}}$ – количество рабочих дней в году; $Вр_{\text{с}}$ – время работы компьютера в течение суток, ч; $C_{\text{т.э}}$ – стоимость одного кВт · ч электроэнергии, руб.; $M_{\text{вс}}$ – мощность вычислительной системы, кВт.

Время использования вычислительной техники рассчитывается по следующей формуле:

$$Вр_{\text{в.т}} = K_{\text{д.р}} \cdot Вр_{\text{с}},$$

где $Вр_{\text{в.т}}$ – время использования вычислительной техники, ч; $K_{\text{д.р}}$ – количество дней разработки ПО.

Затраты на носители информации принимаются в размере 2 % от цены вычислительной техники.

Затраты на текущий и профилактический ремонт принимаются в размере 4 % от цены вычислительной техники.

Прочие эксплуатационные расходы включают в себя затраты на освещение, отопление, охрану, уборку и текущий ремонт помещений. Они принимаются в размере 10 % от стоимости помещения (или его аренды), где происходит разработка программного продукта.

7.3 Расчет цены программного продукта

Определение цены является одной из труднейших задач, стоящих перед любым предприятием. И именно цена предопределяет успехи предприятия – объемы продаж, доходы, получаемую прибыль.

Установление определенной цены на программный продукт служит для последующей его продажи и получения прибыли. Очень важно назначить цену таким образом, чтобы она не оказалась слишком высокой или слишком низкой.

Для определения минимальной цены, ниже которой разработчику будет невыгодно продавать программный продукт, используется следующая формула:

$$Ц_{п.п} = C_{п.п} \cdot (1 + H_{пр}),$$

где $Ц_{п.п}$ – цена программного продукта, руб.; $C_{п.п}$ – себестоимость программного продукта, руб.; $H_{пр}$ – норматив прибыли (20 %, в формуле $H_{пр} = 0,2$).

7.4 Расчет экономической эффективности

В расчет экономической эффективности входит определение эксплуатационных расходов и капитальных затрат потребителя, годовой экономии эксплуатационных расходов у одного потребителя, срока окупаемости программного продукта, годового экономического эффекта.

Расходы потребителя, связанные с эксплуатацией программы, определяются по следующей формуле:

$$P_{э.п} = Vp_{п.п} \cdot C_{м.вр} + Ц_{п.п}/C_{сл},$$

где $P_{э.п}$ – эксплуатационные расходы потребителя, руб.; $Vp_{п.п}$ – объем машинного времени в течение года, необходимый для решения

данной задачи с использованием программы, ч; $C_{м.вр}$ – стоимость одного часа машинного времени, руб./ч; $Ц_{п.п}$ – цена программного продукта, руб.; $C_{сл}$ – срок службы программного продукта, год. Обычно составляет 1 – 2 года, затем выпускается новая версия программного продукта.

Если на момент внедрения программного продукта у потребителя все работы выполнялись вручную, тогда капитальные затраты рассчитываются по формуле

$$P_{\text{кап}} = \frac{Вр_{п.п} \cdot K_{\text{ЭВМ}}}{\Phi_{\text{вр}}} + Ц_{п.п},$$

где $P_{\text{кап}}$ – капитальные расходы потребителя, руб.; $\Phi_{\text{вр}}$ – полезный годовой фонд времени работы вычислительной техники, принимается условно 2000 ч в год; $K_{\text{ЭВМ}}$ – капитальные затраты на вычислительную технику, для которой предназначена программа, руб.

Капитальные затраты на вычислительную технику рассчитываются по формуле

$$K_{\text{ЭВМ}} = Ц_{\text{ЭВМ}} + P_{п.п},$$

где $Ц_{\text{ЭВМ}}$ – цена вычислительной техники, руб.; $P_{п.п}$ – прочие расходы потребителя, связанные с помещением (отопление, освещение, уборка и т.д.), принимаются в размере 10 % от стоимости помещения потребителя (или его аренды), руб.

Для расчета годовой экономии эксплуатационных расходов потребителя вычисляются эксплуатационные затраты потребителя при решении задачи вручную:

$$P_{\text{э.руч}} = 1,21 \cdot \PhiЗП \cdot 12,$$

где $P_{\text{э.руч}}$ – эксплуатационные расходы потребителя при решении задачи вручную, руб.; $\PhiЗП$ – фонд заработной платы персонала, обслуживающего решение задачи вручную, руб.; 12 – количество месяцев в году; 1,21 – поправочный коэффициент.

Тогда годовая экономия эксплуатационных расходов у одного потребителя рассчитывается по формуле

$$\mathcal{E} = P_{\text{э.руч}} - P_{\text{э.п.}}$$

Срок окупаемости программного продукта рассчитывается по формуле

$$T_{\text{ок}} = \frac{P_{\text{кап}}}{\mathcal{E}}.$$

Годовой экономический эффект, получаемый одним потребителем, рассчитывается по формуле

$$\mathcal{E}\mathcal{E} = \mathcal{E} - E_{\text{н}} \cdot P_{\text{кап}},$$

где $E_{\text{н}}$ – нормативный коэффициент эффективности дополнительных капитальных вложений, равный 0,15.

Раздел 8

РЕКОМЕНДАЦИИ ПО АНАЛИЗУ ФАКТОРОВ БЕЗОПАСНОСТИ ЖИЗНЕДЕЯТЕЛЬНОСТИ

Полная безопасность труда человека в производственных условиях определяется тремя факторами: безопасность производственного оборудования, производственного и трудового процессов. Эти составляющие безопасности труда связаны между собой (рис. 8.1).



Рис. 8.1. Треугольник безопасности

Повышение или понижение пределов отклонений параметров технологического процесса приводит к возникновению опасных и вредных производственных факторов, что оказывает влияние на безопасность труда обслуживающего персонала. Аналогичное воздействие оказывает состояние производственного оборудования. Факторы безопасности производственного оборудования являются составной частью общей системы безопасности труда. Совокупность безопасности производственного процесса и оборудования образует материальный и технический фактор безопасности производства, который на действующем оборудовании и применяемой технологии может совершенствоваться как путем изменения технологии и оборудования, так и путем оснащения технологии и оборуду-

дования техническими средствами безопасности и производственной санитарии.

В связи с этим уровень безопасности труда можно рассматривать как произведение уровней безопасности оборудования b_1 , производственного b_2 и трудового b_3 процессов: $V = b_1 \cdot b_2 \cdot b_3$. Предел уровня безопасности для каждой подсистемы равен единице.

Необходимо выполнить анализ безопасности труда и определить долю каждого фактора в треугольнике безопасности. При этом дается характеристика роли пользователя в системе, уделяется внимание объективным (производственным) факторам: рабочее место, шумоизоляция, кондиционирование воздуха, освещенность и т.д., а также субъективным (человеческим) факторам: быстрота реакции, достаточно хорошее зрение, внимательность и т.д.

8.1 Анализ вредных производственных факторов и мероприятия по их снижению

Вредный производственный фактор – производственный фактор, воздействие которого на человека может привести к заболеванию.

Дипломник определяет основные вредные производственные факторы (запыленность, шум, вибрация, тепловое излучение, электрический ток, электромагнитное излучение и т.д.), их воздействие на жизнедеятельность человека и описывает мероприятия по снижению воздействия этих факторов на пользователя.

8.2 Анализ опасных производственных факторов и мероприятия по их снижению

Опасный производственный фактор – производственный фактор, воздействие которого на человека может привести к травме.

Дипломник определяет основные опасные производственные факторы (повышенная температура поверхности оборудования, концентрация горючих газов в воздухе рабочей зоны, взрывоопасность, движущиеся механизмы и машины и т.д.), их воздействие на

жизнедеятельность человека и описывает мероприятия по снижению воздействия этих факторов на пользователя.

8.3 Оценка электробезопасности помещения и мероприятия по повышению электробезопасности

В данном разделе дается оценка электробезопасности на рабочем месте, определяется класс помещения по опасности поражения электрическим током, характеристики сети и т.д.

По степени опасности поражения человека электрическим током помещения подразделяются на:

1) помещения без повышенной опасности, в которых отсутствуют условия, создающие повышенную или особую опасность;

2) помещения с повышенной опасностью, характеризующиеся наличием в них одного или нескольких условий: сырости, токопроводящей пыли, токопроводящих полов (металлических, земляных, железобетонных, кирпичных), высокой температуры, возможности одновременного прикосновения человека к имеющим соединение с землей металлоконструкциям здания, технологическим аппаратам, с одной стороны, и к металлическому корпусу электрооборудования – с другой;

3) особо опасные помещения, характеризующиеся наличием одного из следующих условий: особой сырости, химически активной или органической среды, одновременно двух или более условий повышенной опасности.

На основании анализа дипломник определяет конкретные средства защиты (заземление, ограждение, зануление, автоматические отключатели, наличие резиновых ковриков и т.д.).

8.4 Оценка помещения по пожарной безопасности и мероприятия по повышению пожарной безопасности

В данном разделе приводится классификация помещения по пожарной опасности, степень огнестойкости всего здания, причины и источники возгорания, классы и виды пожаров.

На основе полученных данных дипломник распределяет оборудование внутри помещения и выбирает необходимые средства пожаротушения (песок, кошма, огнетушители с указанием количества и марки, количество пожарных щитов, количество пожарных гидрантов и т.д.), указывает наличие или отсутствие автоматической противопожарной сигнализации.

8.5 Анализ неполадок и аварийных ситуаций

В данном разделе дается прогноз основных возможных неполадок и аварийных ситуаций и определяются мероприятия по защите от неполадок и аварий (источники бесперебойного питания для вычислительной техники; установка контролирующей аппаратуры: датчики, контрольные лампы; установка защитной аппаратуры: автоматы, предохранители; повышенная надежность и быстродействие системы и т.д.).

8.6 Анализ мероприятий по обеспечению экологической чистоты

Контроль за состоянием окружающей среды можно производить при помощи измерения:

- 1) температуры;
- 2) относительной влажности;
- 3) скорости движения воздуха;
- 4) теплоизлучения.

В данном разделе выполняется сравнение приведенных выше параметров с действующими санитарно-техническими требованиями к зданиям и сооружениям. Так, например, температура окружающей среды должна составлять 15 – 25 °С; относительная влажность – 40 – 60 %; скорость движения воздуха: летом – 0,5 – 1,0 м/с, зимой – 0,3 – 0,5 м/с. Приводится перечень мероприятий по снижению вредных воздействий на окружающую среду.

АВТОМАТИЗАЦИЯ ДИПЛОМНОГО ПРОЕКТИРОВАНИЯ С ИСПОЛЬЗОВАНИЕМ CASE-СРЕДСТВ

9.1 BPwin

BPwin имеет достаточно простой и интуитивно понятный интерфейс пользователя, дающий возможность создавать сложные модели. При запуске BPwin по умолчанию появляется основная панель инструментов, палитра инструментов (вид которой зависит от выбранной нотации) и навигатор модели – Model Explorer (рис. 9.1). Функциональность панели инструментов доступна из основного меню BPwin.

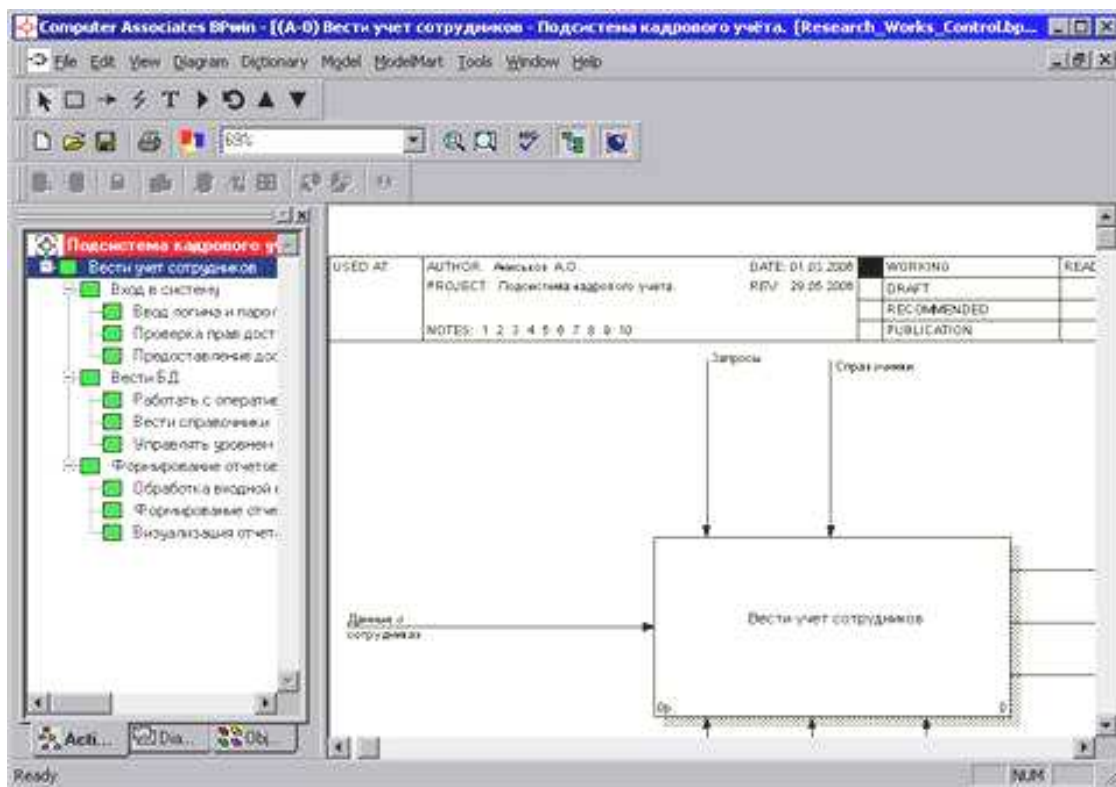


Рис. 9.1. Интегрированная среда разработки модели BPwin

При создании новой модели возникает диалог, в котором следует указать, будет ли создана модель заново или она будет открыта из файла, внести имя модели и выбрать методологию, в которой будет построена модель (рис. 9.2).

BPwin поддерживает три методологии – IDEF0, IDEF3 и DFD, каждая из которых решает свои специфические задачи. В BPwin возможно построение смешанных моделей, т.е. модель может содержать одновременно как диаграммы IDEF0, так и диаграммы IDEF3 и DFD. Состав палитры инструментов изменяется автоматически, когда происходит переключение с одной нотации на другую.

После щелчка по кнопке OK появляется диалог Properties for New Models (рис. 9.3), в котором следует внести свойства модели.

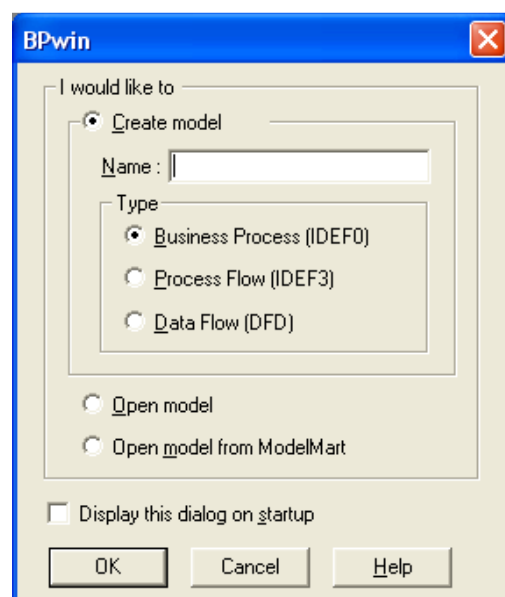


Рис. 9.2. Диалог создания модели

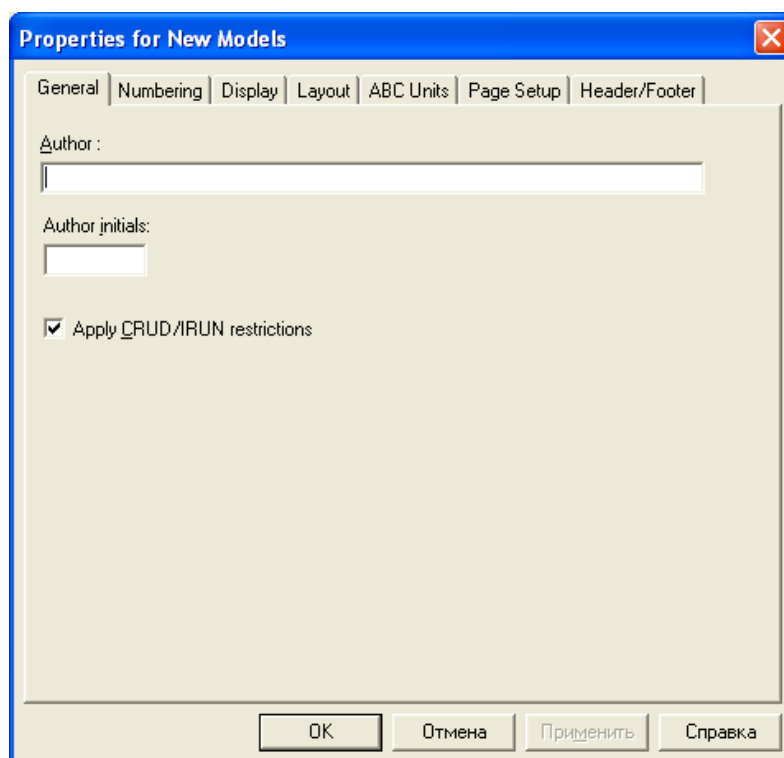


Рис. 9.3. Диалог Properties for New Models

Модель в BPwin рассматривается как совокупность работ, каждая из которых оперирует некоторым набором данных. Работа изображается в виде прямоугольника, данные – в виде стрелок.

Model Explorer – навигатор модели

Инструмент навигации Model Explorer имеет три вкладки – Activities, Diagrams и Objects. Вкладка Activities показывает в виде раскрывающегося иерархического списка все работы модели. Одновременно могут быть показаны все модели, открытые в BPwin. Работы с диаграмм IDEF0 отображаются зеленым цветом, IDEF3 – желтым, DFD – голубым.

Щелчок по работе во вкладке Activities переключает левое окно BPwin на диаграмму, на которой эта работа размещена. Для редактирования свойств работы следует щелкнуть по ней правой кнопкой мыши, вызвав контекстное меню. Значение пунктов контекстного меню приведено в табл. 9.1.

Таблица 9.1

Контекстное меню редактирования свойств работы

Пункт меню	Описание
Insert Before	Создать новую работу на той же диаграмме. В списке работ новая работа будет вставлена перед текущей
Insert After	Создать новую работу на той же диаграмме. В списке работ новая работа будет вставлена после текущей
Decompose	Декомпозировать работу. В результате будет создана новая диаграмма декомпозиции
Name	Вызов редактора имени работы
Definition/Note	Вызов редактора определения и примечания к работе
Font	Изменение шрифта работы
Color	Изменение цвета работы
Costs	Задание стоимости работе
UDP	Задание свойств, определяемых пользователем
UOW	Задание свойств для работ IDEF3

Вкладка Diagrams служит для перехода на любую диаграмму модели (рис. 9.4).

После перехода на вкладку Objects на ней отображаются все объекты, соответствующие выбранной на вкладке Diagrams диаграмме.

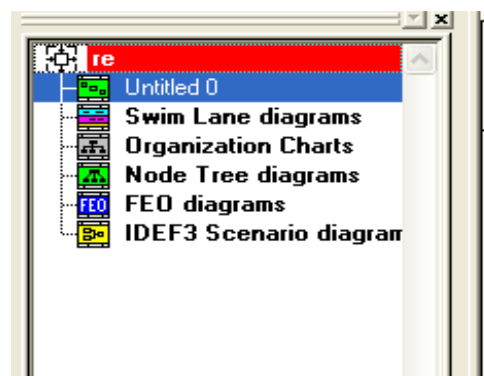


Рис. 9.4. Вкладка Diagrams навигатора Model Explorer


Создание функциональной модели с помощью BPwin

Создание модели в стандарте IDEF0

В IDEF0 система представляется как совокупность взаимодействующих работ или функций. Такая чисто функциональная ориентация является принципиальной – функции системы анализируются независимо от объектов, которыми они оперируют. Под моделью в IDEF0 понимают описание системы (текстовое и графическое), которое должно дать ответ на некоторые заранее определенные вопросы.

Процесс моделирования какой-либо системы начинается с определения контекста, т.е. наиболее абстрактного уровня описания системы в целом. В контекст входит определение субъекта моделирования, цели и точки зрения.

Итак, на первом этапе проектирования необходимо создать контекстную диаграмму. Для построения следующих диаграмм потребуются предыдущие, поэтому следует сохранять модель, полученную на каждом шаге.

1. После запуска пакета BPwin следует выбрать кнопку , ввести имя модели и выбрать тип диаграммы IDEF0 (функциональная диаграмма). При этом шаблон контекстной диаграммы создается автоматически.

2. Во вкладке Activities компонента Model Explorers щелчок правой кнопкой по объекту позволяет редактировать его свойства.

3. В пункте меню Model/Model Properties следует выполнить следующие действия (рис. 9.5):

- во вкладке General внести имя модели, имя проекта, имя автора и тип модели;
- во вкладке Purpose внести цель и точку зрения;
- во вкладке Definition внести определение и цель.

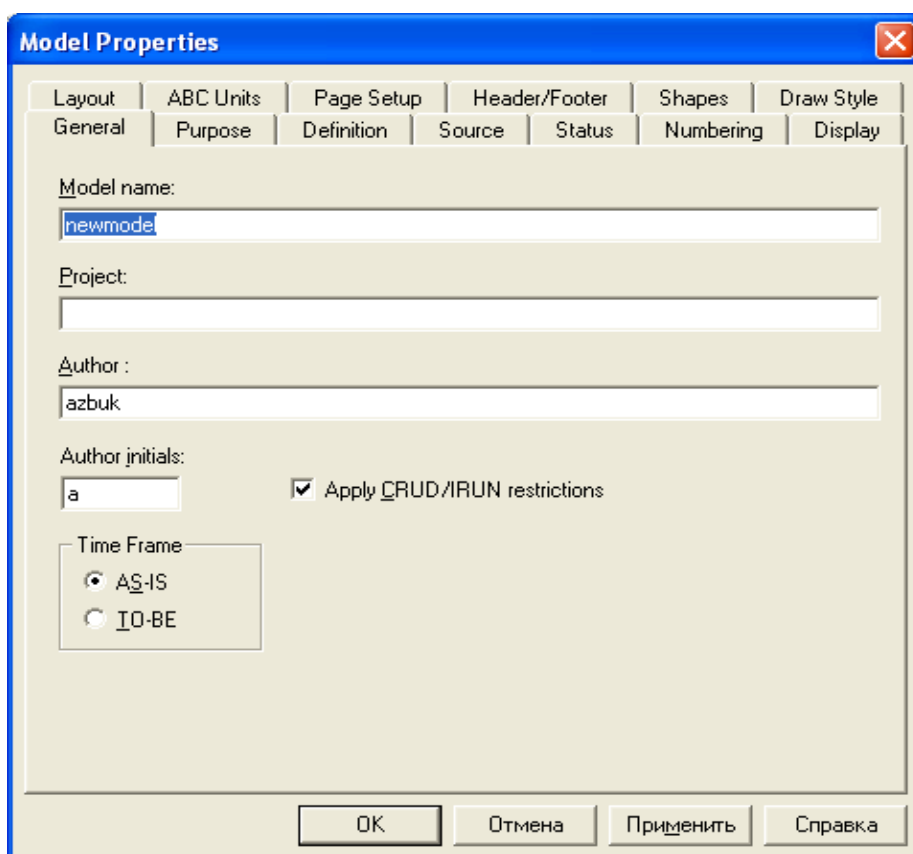


Рис. 9.5. Диалог задания свойств модели

4. Далее следует перейти на контекстную диаграмму и вызвать контекстное меню. Выбрав пункт Name, следует внести имя функции на контекстной диаграмме, во вкладке Definition – определение.

5. После этого необходимо на контекстной диаграмме создать стрелки, внести для каждой из них имя (Arrow Name), определение (Arrow Definition), тип (Arrow Type) (рис. 9.6).

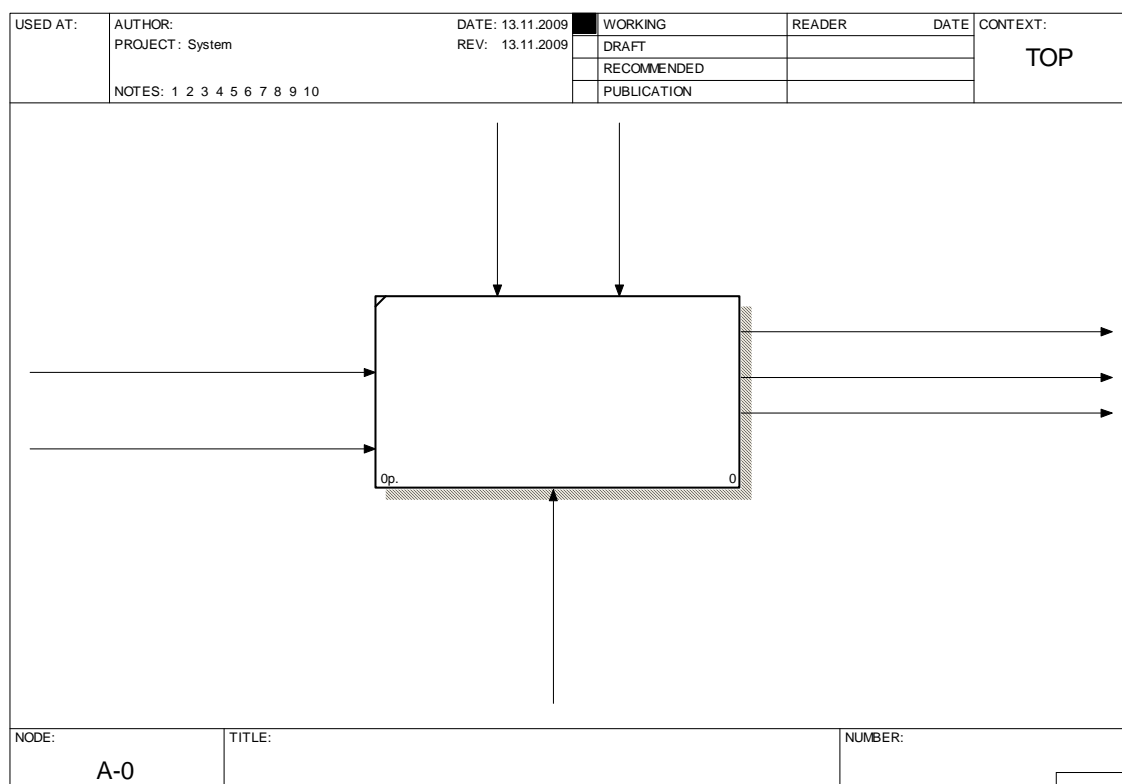



Рис. 9.6. Заготовка контекстной диаграммы


6. С помощью кнопки **Т** в поле диаграммы можно внести текст, например цель и точку зрения.

7. Результатом предыдущих действий является контекстная диаграмма.

На втором этапе проектирования необходимо создать диаграммы декомпозиции.

1. Для перехода к диаграммам декомпозиции следует выбрать в палитре инструментов кнопку перехода на нижний уровень  и в диалоге Activity Box Count установить количество работ на диаграмме нижнего уровня, а также нотацию IDEF0. После этого автоматически будет создана диаграмма декомпозиции (рис. 9.7).

2. Вызвав контекстное меню, следует внести имя (Name) и определение (Definition) для всех работ диаграммы. Для изменения свойств работ после их внесения в диаграмму можно воспользоваться словарем работ. Вызов словаря – меню Dictionary/Activity.

Если описать имя и свойства работы в словаре, то ее можно будет внести в диаграмму позже с помощью кнопки  в палитре инструментов.

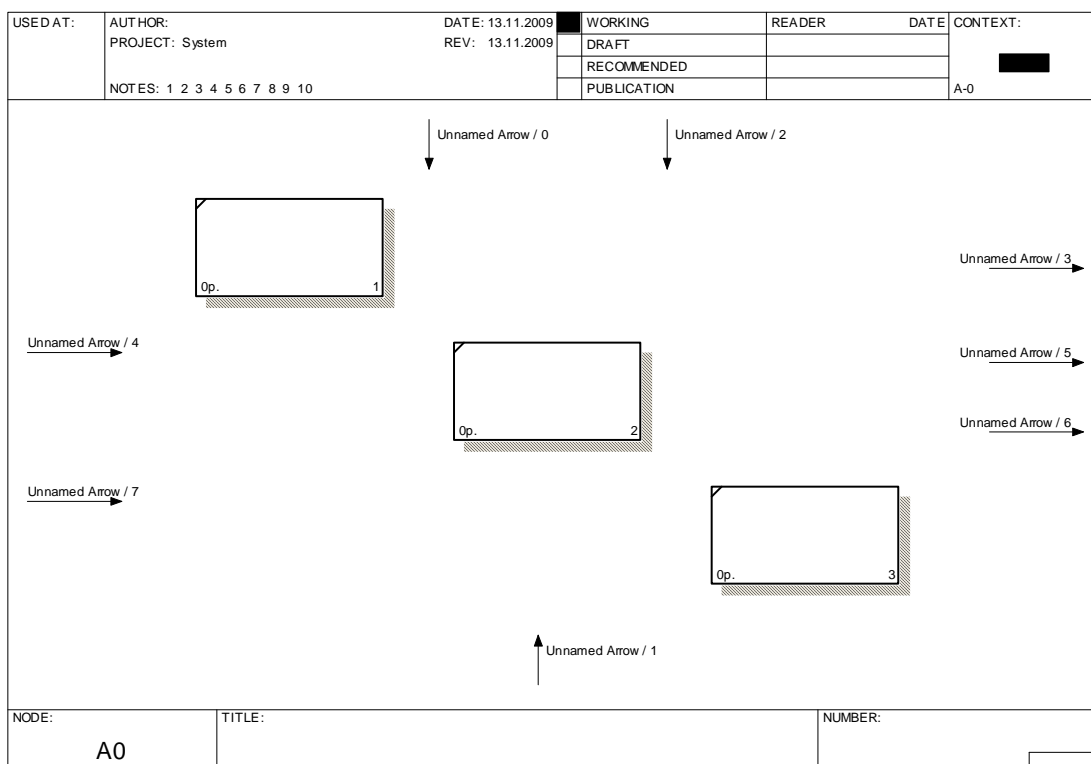



Рис. 9.7. Заготовка диаграммы декомпозиции

Невозможно удалить работу из словаря, если она используется на какой-либо диаграмме. Если работа удаляется с диаграммы, из словаря она не удаляется, поэтому ее имя и описание может быть использовано в дальнейшем.

Для добавления работы в словарь необходимо перейти в конец списка и щелкнуть правой кнопкой мыши по последней строке. При этом возникает новая строка, в которой нужно внести данные о работе.

3. Далее в режиме рисования стрелок следует связать граничные стрелки (кнопка  на палитре инструментов) с работами. Имена и свойства стрелок можно внести с использованием словаря стрелок Dictionary/Arrow или щелкнув правой кнопкой мыши по стрелке.

4. Работы следует также соединить внутренними стрелками, при этом могут быть созданы стрелки обратной связи.

5. При создании новой граничной стрелки выхода (если это необходимо) она автоматически не попадает на диаграмму верхнего уровня и имеет квадратные скобки на кончике []. Чтобы удалить квадратные скобки, достаточно щелкнуть по ним правой

кнопкой мыши и выбрать пункт меню Arrow Tunnel. В диалоге Border Arrow Editor следует выбрать опцию Resolve it to Border arrow. Для стрелки нужно выбрать опцию Trim из контекстного меню.

Следующий этап проектирования предполагает декомпозицию каждой из созданных на предыдущем этапе работ.

Создание диаграммы IDEF3

1. Контекстную диаграмму нотации IDEF0 можно декомпонировать, выбрав нотацию IDEF3 (диалог Activity Box Count).

2. Вызвав контекстное меню каждой работы, следует внести ее имя, определение и свойства.

3. Можно также внести перекрестки одного из типов (рис. 9.8), после чего связать их с работами. Если от перекрестка нужно сделать разветвление, следует щелкнуть по нему.

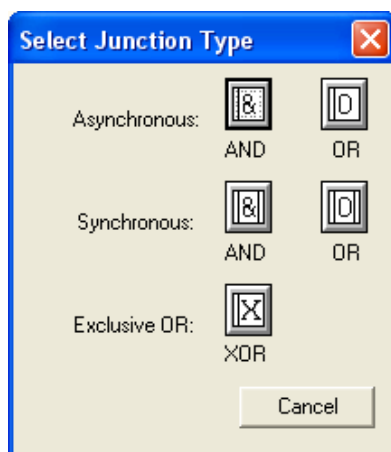


Рис. 9.8. Типы перекрестков

Связи в IDEF3 показывают взаимоотношения работ. Различают три типа стрелок, изображающих связи, стиль которых устанавливается во вкладке Style диалога Arrow Properties (рис. 9.9).

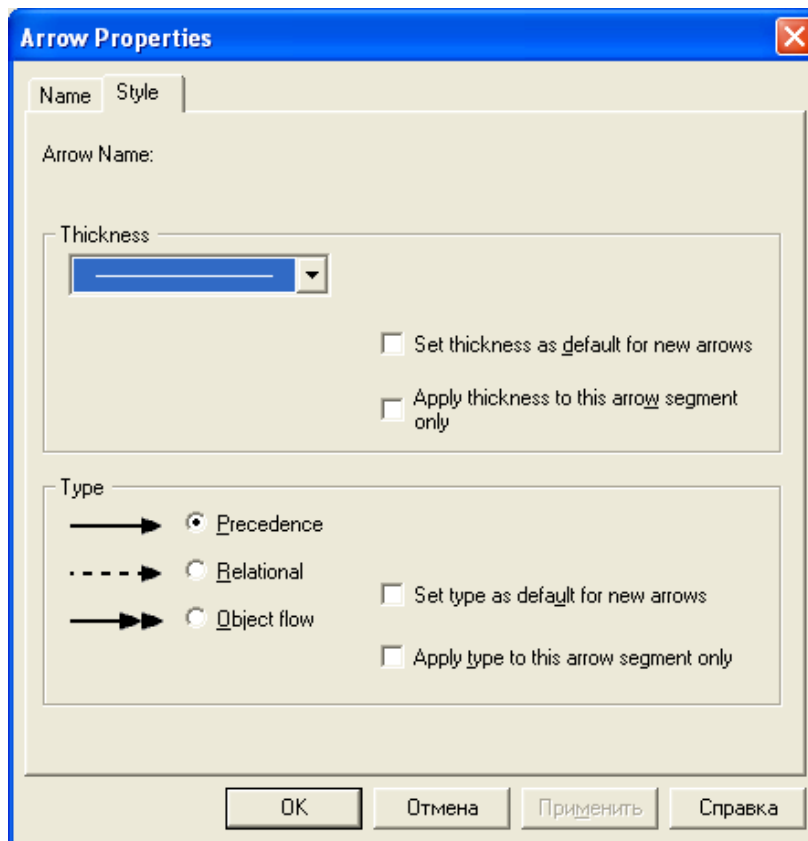


Рис. 9.9. Вкладка Style диалога Arrow Properties

Создание диаграммы DFD

1. Для декомпозиции диаграммы в нотации DFD следует выбрать соответствующий тип в диалоговом окне Activity Box Count.
2. Имена работ вносятся тем же способом, что и на предыдущих типах диаграмм.
3. Затем нужно удалить граничные стрелки с диаграммы.
4. Если внутренние стрелки должны быть двунаправленными, следует щелкнуть правой кнопкой по стрелке и выбрать в контекстном меню пункт Style, опцию Bidirectional.

9.2 ERwin

ERwin – средство концептуального моделирования БД, использующее методологию IDEF1X. Методология IDEF1X была разрабо-

тана с учетом таких требований, как простота изучения и возможность автоматизации. Она базируется на методе IDEF1, разработанном Т. Рэмей и основанном на подходе П. Чена. Метод IDEF1 позволяет построить модель данных, эквивалентную реляционной модели в третьей нормальной форме.

ERwin реализует проектирование схемы БД, генерацию ее описания на языке целевой СУБД (ORACLE, Informix, Ingres, Sybase, DB/2, Microsoft SQL Server, Progress и др.) и реинжиниринг существующей БД. Для ряда средств разработки приложений (PowerBuilder, SQLWindows, Delphi, Visual Basic) выполняется генерация форм и прототипов приложений.

Рабочее окно программы ERwin содержит (рис. 9.10):

- строку главного меню;
- панель инструментов, где находятся кнопки с командами, которыми можно пользоваться при построении и модифицировании диаграммы ERwin (Новая диаграмма, Открыть диаграмму, Сохранить диаграмму, Печать диаграммы, Уровень сущности, Уровень атрибута, Уровень определения сущности, Уровень физической схемы, Уменьшить, Увеличить, Прямое проектирование (генерация схемы БД), Синхронизация БД, Подсоединиться к базе данных, Выбрать сервер и др.);
- окно инструментов, которые используются при определении модели данных (Инструмент выбора, Создать сущность, Полная связь подтипов, Текстовый блок, Инструмент управления атрибутами, Идентифицирующая связь, Связь «многие ко многим», Неидентифицирующая связь);
- в нижней части окна – статусную строку с информацией о выбранной функции или команде меню;
- окно диаграммы.

Если работать в нескольких окнах диаграмм, то ERwin запоминает последний выбранный инструмент для каждого окна. При возвращении в окно можно сразу же использовать инструмент, который был выбран до того, как перешли к другому окну.

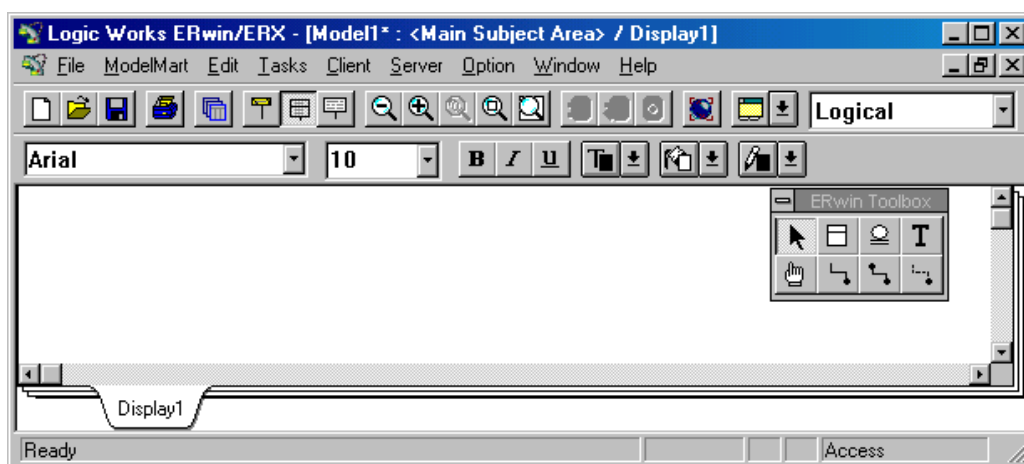


Рис. 9.10. Рабочее окно программы ERwin

Пример разработки логической модели в ERwin

Рассмотрим цикл разработки логической модели на примере следующей предметной области. Ведется учет служащих; для каждого служащего хранится информация о детях и о списке должностей, занимаемых этим служащим. Для должностей ведется учет информации об установленных должностных окладах.

Сначала создадим логический уровень модели. Для этого зададим режим отображения сущностей (Display/Entity Level). При помощи линейки инструментов получим сущности «Служащий», «Дети», «История работы», «История зарплаты». Будем именовать сущности на русском языке. Каждую сущность подробно опишем на русском языке в редакторе Entity Definition. Это описание появится в отчетах ERwin и может быть отображено на диаграмме. Укажем связи между сущностями. Например, «Служащий» связан идентифицирующей связью «является родителем» с сущностью «Дети». Описание связи вводится в редакторе Editor/Relationship.

Результат работы отображен на рис. 9.11.

Теперь перейдем в режим задания атрибутов (Display/Attribute Level). В редакторе Entity/Attribute на русском языке зададим имена ключевых и неключевых атрибутов. Заметим, что для дочерней сущности «Дети» ключевой атрибут «Номер служащего» не указы-

вается вручную. ERwin обеспечивает его миграцию из родительской сущности. То же происходит с другими дочерними сущностями. Для атрибута «Имя» сущности «Служащий» укажем, что этот атрибут является альтернативным ключом (будем считать, что у всех служащих уникальные имена/фамилии). Для этого после имени атрибута поместим указатель АК1 в скобках. Результат работы отображен на диаграмме ERwin в нотации IDEF1X (рис. 9.12).

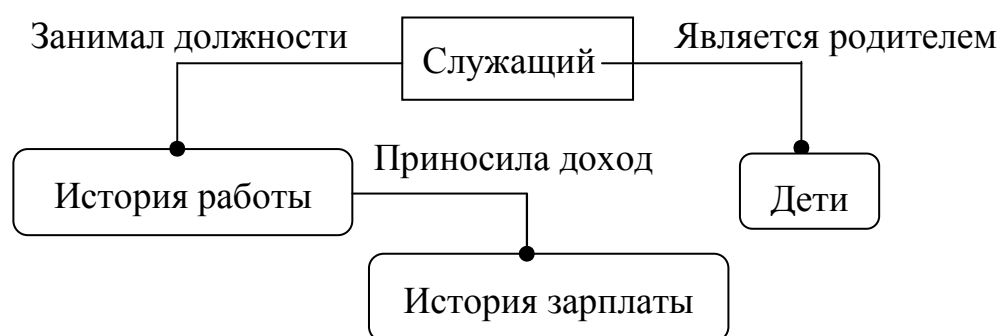


Рис. 9.11. Диаграмма уровня сущности

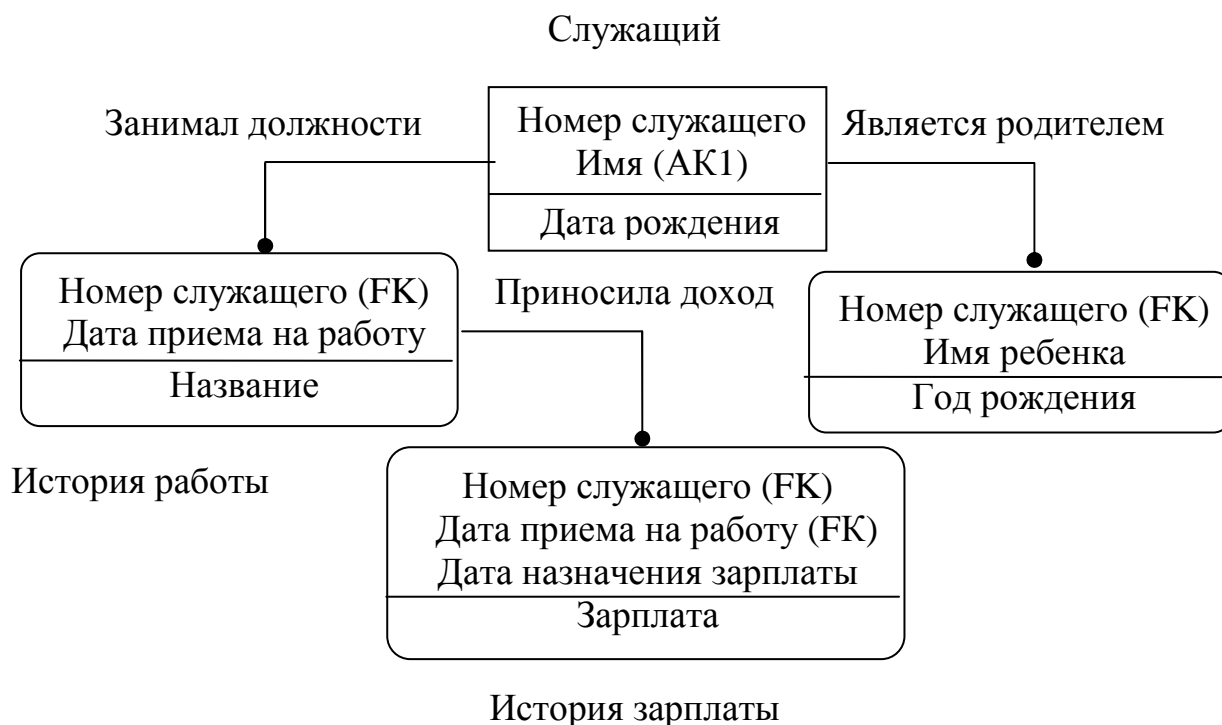


Рис. 9.12. Диаграмма уровня атрибутов в нотации IDEF1X

Вид той же диаграммы в нотации ІЕ (Information Engineering) показан на рис. 9.13.

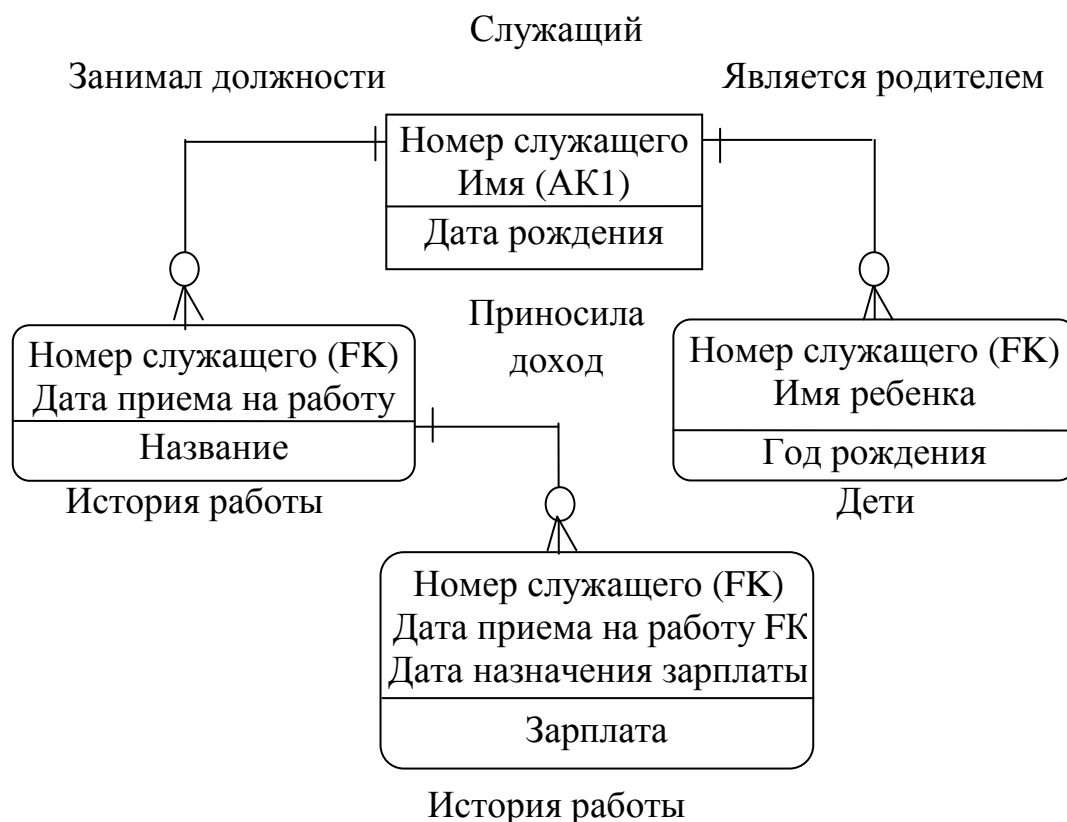


Рис. 9.13. Диаграмма уровня атрибутов в нотации ІЕ

Физическое моделирование в ERwin

В ERwin существует два уровня представления и моделирования – логический и физический. Логический уровень означает прямое отображение фактов из реальной жизни. Например, люди, столы, отделы, собаки и компьютеры являются реальными объектами. Они именуются на естественном языке, с любыми разделителями слов (пробелы, запятые и т.д.). На логическом уровне не рассматривается использование конкретной СУБД, не определяются типы данных (например, целое или вещественное число) и не определяются индексы для таблиц.

Целевая СУБД, имена объектов, типы данных и индексы составляют второй (физический) уровень модели ERwin. ERwin дает возможность создавать и управлять двумя различными уровнями

представления одной диаграммы (модели), а также позволяет иметь много вариантов отображения на каждом уровне.

Пример разработки физической модели в ERwin

Так как имена атрибутов и сущностей заданы на русском языке, для перехода к физическому уровню модели им следует поставить в соответствие идентификаторы таблиц, колонок и ограничений, удовлетворяющие правилам целевой СУБД (обычно это означает использование латинских букв, цифр и некоторых специальных символов). В редакторе Database Schema указываем для каждой сущности соответствующее имя таблицы. Затем в редакторе Attribute Definition задаем имена колонок таблиц, соответствующие атрибутам сущностей. ERwin и здесь обеспечивает миграцию имен колонок в подчиненные таблицы.

На этом этапе можно воспользоваться редактором Extended Attributes для определения расширенных атрибутов PowerBuilder (формата отображения, маски редактирования, правил контроля, выравнивания, заголовков и комментариев). В редакторе Relationship Definitions указывается физическое имя связи, которое соответствует имени ограничения (constraint), создаваемого ERwin в базе данных. Теперь все готово к созданию БД и нужно выбрать целевую СУБД (если этого не было сделано раньше). Выберем, например, СУБД Sybase System 10. В редакторе SYBASE Database Schema зададим типы данных для колонок таблиц. Диалог, в котором происходит выбор типа данных, приведен на рис. 9.14.

Теперь можно перейти к созданию базы данных. Для этого выполняется команда «Sybase schema generation». ERwin строит пакет SQL-предложений генерации базы данных. В диалоге выбора параметров для генерации БД может быть задан фильтр (генерация не всех таблиц), пакет SQL-предложений можно просмотреть (Preview), распечатать, сохранить в файле (Report), выполнить его генерацию (Generate).

Проектирование на физическом уровне выполняется с использованием терминов той базы данных, которую предполагается использовать в системе. Важно, что ERwin известны соответствия

между СУБД различных производителей, вследствие чего возможно преобразование физической схемы, спроектированной для одной СУБД, в другую.

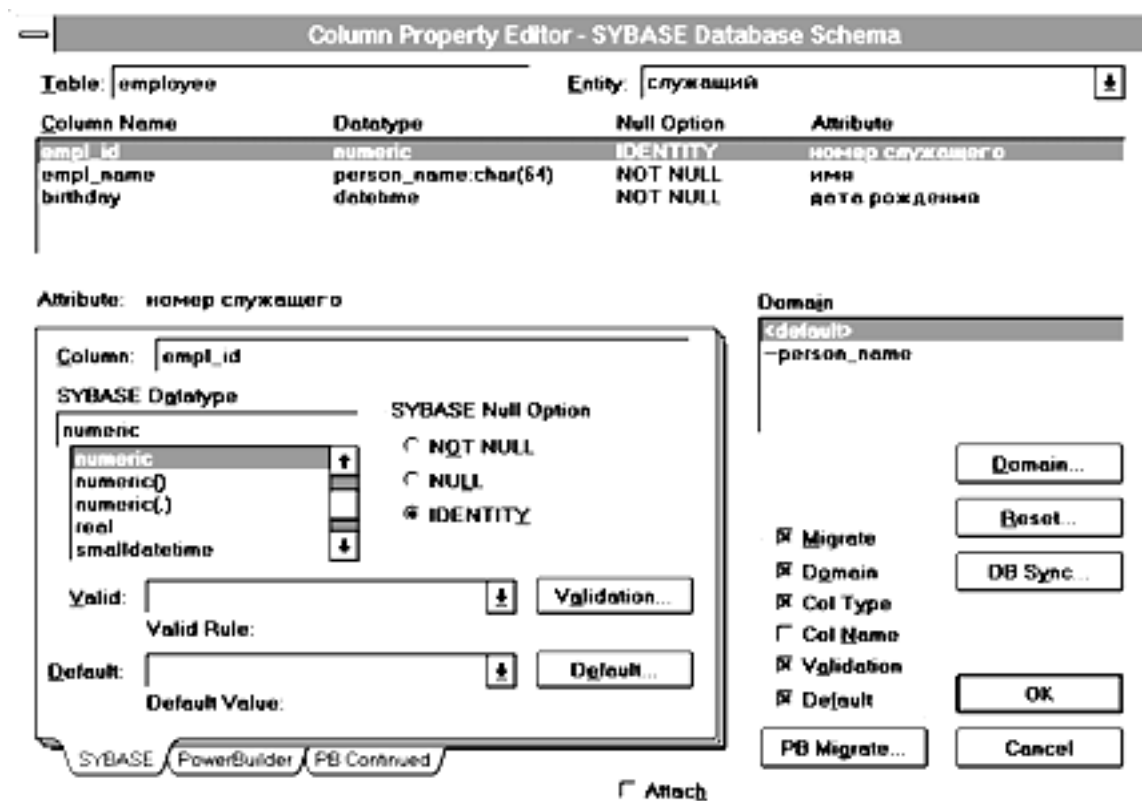


Рис. 9.14. Определение физической модели

Для создания физической структуры БД может быть запрошена генерация DDL-скрипта (Data Definition Language). При этом используется диалект SQL для выбранного типа и версии сервера. Хотя сгенерированный код не нуждается в модификации, его можно сохранить в файле или распечатать.

Обратное проектирование

Процесс генерации физической схемы базы данных из логической модели данных называется *прямым проектированием* (*Forward Engineering*). Когда генерируется физическая схема, ERwin позволяет включать триггеры ссылочной целостности, хранимые процедуры, индексы, ограничения. ERwin также имеет средства определения таблиц СУБД.

Процесс генерации логической модели из физической базы данных называется *обратным проектированием (Reverse Engineering)*. ERwin позволяет быстро создавать модель данных путем обратного проектирования имеющейся базы данных. После того как создана модель ERwin, можно произвести обратное проектирование структуры базы данных, а затем легко перенести его в другой формат базы данных.

Объемлющий механизм генерации схемы в ERwin, удобные возможности синхронизации с базой данных, способность подсоединяться ко всем ведущим источникам данных (SQL- и PC-ориентированным) облегчают проектирование, управление и перенос физических баз данных.

Обратное проектирование, т.е. восстановление информационной модели по существующей базе данных, используется при выборе оптимальной платформы для настольной (desktop) базы данных или базы данных на mainframe, а также при расширении (или модификации) структуры, которая была построена без необходимой сопроводительной документации. После завершения процесса восстановления модели ERwin автоматически раскладывает таблицы на диаграмме. Теперь можно выполнять модификации уже с использованием логической схемы – добавлять сущности, атрибуты, комментарии, связи и т.д. По завершении изменений одна команда (синхронизировать модель с базой данных) актуализирует все проведенные изменения.

Построение модели может быть выполнено как на основании данных каталога базы данных, так и на основании пакета операторов SQL, с помощью которого была создана база данных.

В процессе разработки информационной системы может возникнуть ситуация, при которой структура базы данных и информационная модель не соответствуют друг другу. ERwin дает возможность привести их в соответствие. Для этого предусмотрена функция синхронизации с базой данных. После подключения к СУБД предлагается список несоответствий между существующей структурой данных и моделью. Например, если в базе данных создана новая таблица, то ERwin предложит провести включение ее в модель. Если в модель добавлена новая таблица, ERwin предложит

создать ее в реальной базе данных. При добавлении колонок в базе данных или в модели ERwin предлагает провести соответствующие операции по синхронизации.

ERwin знает о таких особенностях хранения данных в отдельных СУБД, как сегменты (в Sybase) и табличное пространство (в Oracle). Информация о физическом размещении может быть включена в модель и использована при прямом и обратном проектировании.

Создание отчетов

ERwin позволяет создавать собственные отчеты, обобщающие графическую информацию, которая содержится в диаграмме ERwin. Для создания отчета можно выбрать один из распространенных форматов, например формат колонок (Columnar), когда информация появляется под соответствующим заголовком, или формат меток (Labeled), когда информация появляется под соответствующей меткой (например, Entity Name: Customer).

ERwin определяет, как будет форматироваться отчет, исходя из режимов форматирования, которые задаются в редакторе Report.

ERwin позволяет создавать отчеты четырех разных типов: Entity Report, Attribute Report, Relationship Report и Constraint Report. В табл. 9.2 включены все доступные режимы, а также расширения файлов для каждого типа отчета, создаваемого в ERwin.

Таблица 9.2

Типы отчетов и их характеристики

Тип отчета	Расширение файла	Доступные режимы
1	2	3
Entity	.ERE	<p>Детали сущности: имя, определение, notes и т.п.</p> <p>Определения связанных с сущностью атрибутов и связей.</p> <p>Параметры таблицы физической модели.</p> <p>Критерии валидации.</p> <p>Триггер Default или Entity OvERide</p>

1	2	3
Attribute	.ERA	<p>Детали атрибута: имя, базовое имя, имя роли и знак ключа.</p> <p>Оглавление и определения колонок таблицы физической модели.</p> <p>Значения связанных с атрибутом ограничений</p>
Relationship	.ER	<p>Детали связи: глагольная фраза, тип связи, имена родительской и дочерней сущностей.</p> <p>Значения режимов нулевых значений, ссылочной целостности и кардинальность.</p> <p>Характеристики физической модели: физические имена родительской, дочерней сущностей и связи</p>
Constraint	.ERC	<p>Тип отчета: выберите тип Domain, Default или Validation.</p> <p>Имя домена, тип данных и режим нулевых значений.</p> <p>Имя или значение по умолчанию.</p> <p>Имя атрибута логической модели.</p> <p>Имя колонки, тип данных колонки и имя таблицы физической модели.</p> <p>Критерии валидации</p>

ERwin поддерживает несколько распространенных форматов, в соответствии с которыми он выводит на экран данные отчета. В табл. 9.3 кратко описаны все режимы.

Таблица 9.3

Характеристика режимов форматирования отчета

Режим форматирования отчета	Что делает этот режим
1	2
Report Format	
Labeled	Обозначает каждое значение в отчете при помощи метки (например, Attribute Name: customer_name)

1	2
Fixed	Связывает каждое значение в отчете с фиксированным заголовком колонки
Tab Delimited	Использует метки табуляции для разделения значений в отчете
Comma Delimited	Использует запятую для разделения значений в отчете
DDE Table	Пересылает данные отчета в табличной форме в другое приложение
Multi-Valued Format	
Repeating Group	Помещает связанные между собой значения в одну ячейку таблицы, после каждого значения ставит знак «+»
Filled	Повторяет значение родителя для каждого экземпляра повторяющегося дочернего значения
Header	Выводит родительское значение на экран только один раз для всех связанных с ним дочерних значений
Merge	Определяет, будет ли первая строка значения сливаться с родительским значением. Чтобы слить значения, поставьте метку «X» в окне. Чтобы оставить родительское значение в своей строке, уберите метку «X» из окна
Remove Newline	Убирает символ перехода на новую строку из конца каждой строки отчета. Используйте этот режим для того, чтобы связанные между собой значения поместились на одну строку при экспорте отчета с использованием DDE

Когда ERwin впервые открывает редактор Report, он автоматически загружает определение отчета, устанавливаемое по умолчанию для выбранного типа отчета. ERwin присваивает имя по умолчанию (например, Attribute Report), задает режимы содержания (например, Attribute Name) и форматирования (например, labeled и merge – для заголовка). Можно изменить режимы и определения, задаваемые по умолчанию, включая свои собственные. При нажа-

тии одной из управляющих кнопок внизу редактора ERwin сразу же генерирует отчет, основываясь на заданных режимах содержания и форматирования. Отчет состоит из текстовых данных, взятых из диаграммы.

Порядок, в котором задаются режимы, определяет порядок появления заголовков в отчете. Например, если заданы режимы в следующей последовательности:

- 1) Attribute Name,
- 2) Base name,
- 3) Role Name, то заголовки или метки в отчете будут расположены в том же порядке.

Для получения отчета только один раз необходимо поставить метки в окнах нужных режимов и нажать кнопку Print... Для сохранения выбранных режимов в качестве спецификации, которую потом можно использовать для повторной генерации этого же отчета, следует ввести новое имя определения отчета в текстовое окно Report в верхней части окна-диалога и нажать кнопку New для сохранения определения отчета на диске как части текущей диаграммы.

Для изменения определения отчета нужно выбрать его из списка Report, расположенного в верхней части окна-диалога. Можно изменить имя отчета и (или) задать новые или другие режимы, затем следует нажать кнопку Update для сохранения изменений.

Чтобы удалить определение отчета, необходимо выбрать его из списка Report и нажать кнопку Delete. Файлы отчетов, созданные с использованием удаленного определения отчета, сохраняются.

Можно использовать кнопку Preview... в нижней части редактора Report, чтобы просмотреть содержимое отчета. Для распечатывания готового отчета следует нажать кнопку Print..., а для сохранения выходного файла отчета на диске – кнопку Report...

При нажатии кнопки Close..., расположенной внизу в редакторе Report, ERwin автоматически сохраняет определение отчета и его можно использовать повторно, не задавая снова те же самые режимы каждый раз.

По умолчанию ERwin генерирует отчет по всем сущностям текущей области. Для включения в отчет только каких-то отдельных

сущностей нужно нажать кнопку Filter... ERwin открывает редактор Report Filter, содержащий список всех объектов текущей области. Список находится в окне Report Filter. Исключение объектов, не входящих в отчет, возможно с помощью кнопок Remove и Remove All. Объекты, исключенные по неосторожности, можно включить с помощью кнопок Add и Add All.

Прежде чем распечатать отчет, можно просмотреть его в окне Preview редактора Report. Можно ввести текст непосредственно в окно Preview, чтобы снабдить отчет полезной информацией и комментариями.

Чтобы открыть окно Preview, следует нажать кнопку Preview... ERwin откроет окно Preview и покажет в нем текущий отчет. Можно использовать управляющие функции окна для просмотра всего текста отчетов и изменения размера окна.

Если для распечатки или сохранения отчета используются управляющие функции окна Preview, то ERwin включает всю вводимую информацию в выходной текст отчета. Однако при закрытии окна Preview эта информация пропадает.

Отчет в ERwin состоит из двух элементов: определение отчета, в котором заданы режимы содержания и форматирования отчета, и выходной файл отчета, содержащий реальные данные, сгенерированные определением отчета. Определение отчета можно сохранить как часть текущей диаграммы (подобно области), вводя имя отчета и нажимая кнопку New или Update в редакторе Report. Выходной файл можно сохранить в отдельном текстовом файле в формате ASCII, для этого нужно нажать кнопку Report... в редакторе Report, а затем задать имя файла и путь в окне-диалоге Save As.

Можно записать выходной файл отчета на диск, используя для этого кнопку Report... в редакторе Report или в окне Preview. При записи отчета на диск выходной файл отчета сохраняется в формате ASCII. ERwin использует расширение файла отчета, задаваемое по умолчанию (Entity = .ERE, Attribute = .ERA, Relationship = .ER, Constraint = .ERC), и предлагает ввести имя отчета.

Для записи отчета на диск следует нажать кнопку Report.... ERwin выводит на экран диалог Generate <Report> Type с расшире-

нием файла, задаваемым по умолчанию. Нужно ввести уникальное имя отчета и нажать кнопку ОК.

ERwin сохраняет данные отчета в отдельном файле. Можно открыть отчет ERwin из Microsoft Word, WordPerfect, Excel или любого другого приложения, обрабатывающего тексты или таблицы и читающего файлы ASCII.

Порядок создания отчета

1. Выбрать одну из команд меню Report (Entity Report..., Attribute Report..., Relationship Report... или Constraint Report...) для запуска соответствующего редактора.

2. Для выбора каких-то отдельных сущностей из текущей области и включения в отчет только их следует нажать кнопку Filter... в нижней части окна-диалога. ERwin открывает редактор Report Filter, в котором есть список всех сущностей области – Report Filter. Исключение из списка сущностей возможно с помощью кнопок Remove и Remove All. Для возвращения в редактор Report следует нажать кнопку Close.

3. Задать режимы, определяющие содержание и формат отчета.

Чтобы включить нужный режим, необходимо ввести «X» в окне рядом с названием режима.

Чтобы выключить включенный режим, нужно удалить «X» из окна режима.

4. Чтобы просмотреть содержание и формат отчета в ERwin, следует нажать кнопку Preview, расположенную внизу диалога. ERwin открывает окно Report Preview. Для возвращения в редактор Report нужно нажать кнопку Close.

5. Для изменения режимов содержания и форматирования можно повторить пункты 3 – 4, а затем снова просмотреть отчет.

6. Закончив формирование содержания и форматирование отчета, следует нажать кнопку Print для вывода отчета на печать, а для записи отчета на диск – кнопку Report.

7. Для сохранения определения текущего отчета с новым именем следует щелкнуть по текстовому окну Report, расположенному в верхней части диалогового окна, и, используя стандартные кла-

виши редактирования текста, ввести новое имя отчета и нажать кнопку New для сохранения нового определения отчета.

8. Закончив работу в редакторе Report, следует нажать кнопку Close для возвращения в окно диаграммы.

Определение отчета

Порядок действий при определении отчета следующий:

1. Выбрать одну из команд меню Report (Entity Report..., Attribute Report..., Relationship Report... или Constraint Report...) для входа в редактор Report.

2. Щелкнуть на стрелке «вниз» рядом со списком Report, расположенным наверху диалога, и выбрать отчет, который нужно изменить.

3. Задать режимы, определяющие содержание и формат отчета.

Чтобы изменить порядок расположения колонок, необходимо щелкнуть по тем колонкам, которые следует переставить местами. Затем снова выбрать их в нужном порядке.

Например, заголовки меток или колонок для отчета расположены в следующем порядке: 1) Attribute Name; 2) Base Name; 3) Definition. Для изменения мест Base Name и Definition следует щелкнуть по этим колонкам и убрать «X» из окон, расположенных рядом с данными заголовками. Затем необходимо щелкнуть, поставив метку сначала в окне Definition, а потом в окне Base Name. ERwin переставит местами колонки или метки: 1) Attribute Name; 2) Attribute Definition; 3) Attribute Base Name.

4. Чтобы просмотреть содержание и формат отчета в ERwin, следует нажать кнопку Preview, расположенную внизу диалога. ERwin открывает окно Report Preview. Для возвращения в редактор Report следует нажать кнопку Close.

5. Для изменения режимов содержания и форматирования можно повторить пункты 3 – 4, а затем снова просмотреть отчет.

6. Закончив формирование содержания и отчета, следует нажать кнопку Update для сохранения определения отчета. Для вывода отчета на печать необходимо нажать кнопку Print, для записи отчета на диск – кнопку Report, для возвращения в окно диаграммы – кнопку Close.

Порядок расположения информации в отчете задается последовательностью выбора режимов для отчета.

Если задается режим Fixed Column или DDE Table в групповом окне Report Format, то кнопки Print и Preview на экране становятся серыми. Чтобы просмотреть отчет в формате Fixed Column, необходимо экспортировать отчет в другое приложение с помощью режима DDE Table.

Запись выходного файла отчета на диск

1. Нажать кнопку Report..., расположенную в нижней части редактора Report или Report Preview. ERwin открывает окно-диалог Generate <Report> Type с расширением, присваиваемым по умолчанию (например, .ERE, .ERA, .ER, .ERC).

2. Ввести уникальное имя отчета и нажать ОК. ERwin записывает файл отчета на диск и выходит в редактор Report.

Печать отчета

Нажать кнопку Print..., расположенную в нижней части редактора Report или Report Preview. ERwin сразу же отправит отчет на печать.

Удаление определения отчета

1. Выбрать одну из команд меню Report (Entity Report..., Attribute Report..., Relationship Report... или Constraint Report...) для входа в соответствующий редактор.

2. Щелкнуть на стрелке «вниз» рядом со списком Report, расположенным вверху диалога, и выбрать отчет для удаления.

3. Нажать кнопку Delete, а затем – кнопку Close для возвращения в окно диаграммы.

При удалении определения отчета в редакторе Report сохраненные файлы отчетов, созданные с использованием этого определения, остаются неизменными.

Использование отчетов ERwin с другими приложениями

ERwin дает возможность работать с данными отчетов ERwin в других приложениях.

Задавая режим DDE Table в редакторе Report, можно экспортировать выходной отчет ERwin в любое приложение обработки текстов или таблиц (например, в Word, Excel и др.).

При сохранении отчета ERwin в любом формате (например, columnar, labeled, tab delimited, comma delimited) как текстового файла (например, .ERE, .ERA, .ER, .ERC) возможно открытие этого файла из любого приложения, которое читает файл ASCII. Чтобы изменить внешний вид отчета Erwin, при открытии этого отчета из приложения можно использовать режимы форматирования.

9.3 Rational Rose

Это визуальный редактор, позволяющий моделировать программные системы любой сложности на основе графических диаграмм языка UML (Unified Modeling Language).

После запуска открывается главное окно программы, показанное на рис. 9.15

В верхней части экрана находится меню и строка инструментов (Tool Bar), слева находится окно Browser для быстрого доступа к диаграммам. Под окном Browser находится окно Documentation (документация), в котором появляется описание, введенное разработчиком для выделенного в текущий момент элемента. Каждый объект в Rational Rose имеет свое контекстное меню, через которое можно изменить свойства объекта.

В распоряжение проектировщика система Rational Rose предоставляет следующие типы диаграмм, последовательное создание которых позволяет получить полное представление о всей проектируемой системе и об отдельных ее компонентах:

Use case diagram (диаграммы сценариев);

Deployment diagram (диаграммы топологии);

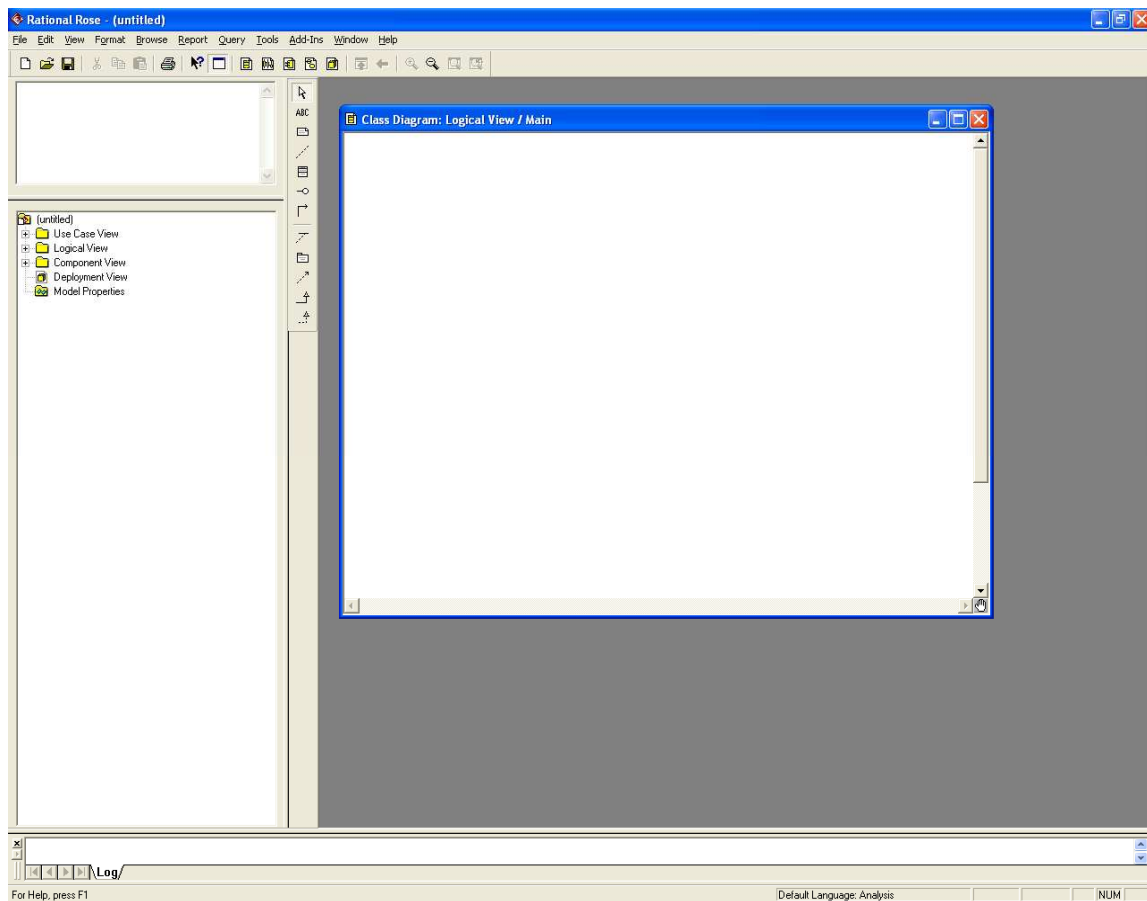


Рис. 9.15. Главное окно программы

Statechart diagram (диаграммы состояний);
 Activity diagram (диаграммы активности);
 Interaction diagram (диаграммы взаимодействия);
 Sequence diagram (диаграммы последовательностей действий);
 Collaboration diagram (диаграммы сотрудничества);
 Class diagram (диаграммы классов);
 Component diagram (диаграммы компонентов).

Состав и назначение пунктов меню

File (файл) предназначен для сохранения, загрузки, обновления проекта, печати диаграмм и дополнительных настроек.

Edit (редактирование) предназначен для копирования и восстановления данных в буфер обмена Windows, а также для редактирования свойств и стилей объектов.

View (вид) предназначен для настройки представления окон меню и строк инструментов.

Format (форматирование) предназначен для настройки формата текущего значка, цветовой гаммы, линий и т.д.

Browse (просмотр) предназначен для навигации между диаграммами и спецификациями диаграмм, представленных в модели.

Report (отчет) предназначен для получения различного вида справок и отчетов.

Query (запрос) предоставляет возможности контролировать, какие элементы модели будут показаны на текущей диаграмме.

Tools (инструменты) предоставляет доступ к различным дополнительным инструментам и подключаемым модулям.

Add-Ins (добавить) предоставляет доступ к менеджеру подключаемых модулей.

Window (окно) позволяет управлять окнами на рабочем столе.

Help (помощь) позволяет получать справочную информацию.

Для создания новой рабочей модели, в которой будут отражены все нюансы системы, необходимо перейти на диаграмму Use Case, как показано на рис. 9.16.

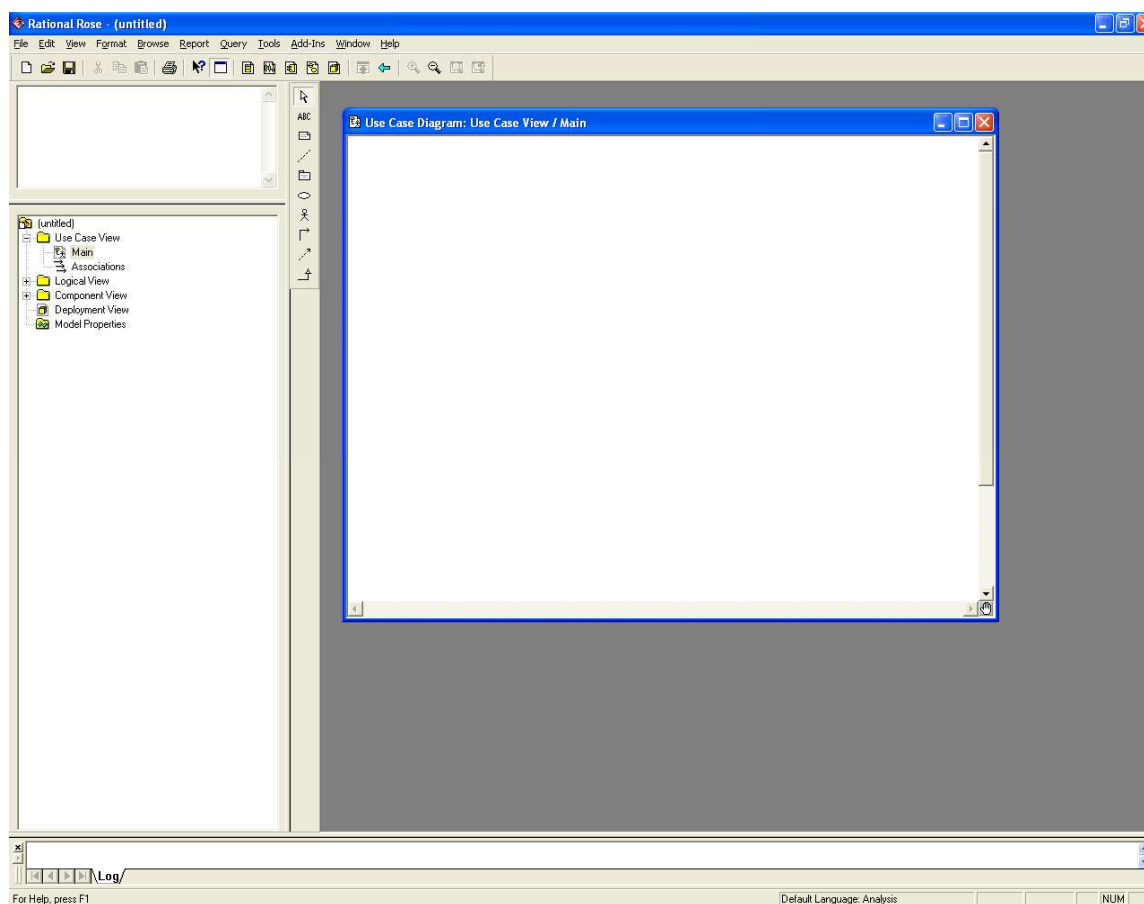


Рис. 9.16. Активизация Use Case диаграммы

Создание новых элементов

Rational Rose предоставляет несколько способов создания новых элементов в модели: пользуясь контекстным меню, при помощи Menu:Tools=>Create или при помощи строки инструментов.

После активизации диаграммы Use Case строка инструментов диаграммы приобретает вид, представленный на рис. 9.17.

Для того чтобы увеличить размер значков, можно выбрать Tool-box=>RClick=>Use Large Buttons.

По умолчанию строка инструментов состоит из десяти значков. Некоторые доступны только на данной диаграмме, но есть и такие, которые могут присутствовать на разных диаграммах, выполняя при этом одинаковые функции.

Selection Tool (инструмент выбора)



Основной инструмент, который позволяет выбирать элементы диаграммы, для того чтобы производить с ними дальнейшие действия. При создании нового элемента диаграммы необходимо выбрать нужный инструмент в строке инструментов, кнопка «залипает», а после создания необходимо опять перейти в режим Selection Tool.

Text Box (текст)



Данный инструмент позволяет создать произвольную надпись на диаграмме, не привязанную ни к какому элементу. Эта надпись не является полноценным элементом модели и не отображается в окне Browse, а используется как комментарий в конкретной диаграмме.

Для создания надписи необходимо нажать кнопку Text Box, при этом курсор примет вид вертикальной стрелки, и щелкнуть на том месте диаграммы, где необходимо создать надпись. В обозначенном квадратными точками в углах окне можно вводить надпись.




Рис. 9.17. Строка инструментов диаграммы Use Case

Для того чтобы изменить введенную надпись, нужно активизировать редактирование двойным щелчком мыши.

Note (замечание) 


Данный инструмент создает элемент замечания, позволяющий вписать в него принятые во время анализа решения. Заметки могут содержать простой текст, фрагменты кода или ссылки на другие документы. Обычно окно Note соединяют с другими элементами диаграммы при помощи инструмента Anchor Note, для того чтобы показать, к какому элементу диаграммы относится замечание. В этом отличие от элемента Text Box, который располагается на диаграмме без присоединения к другим элементам.

Данный элемент не имеет ограничения на количество вводимых символов, и окно Note может быть растянуто, для того чтобы вместить необходимый текст. При активизации этого инструмента курсор принимает форму креста. Контекстное меню для значка Note позволяет кроме установки шрифта устанавливать цвет линий и заливки. Курсор принимает форму креста при создании элементов диаграммы, являющихся объектами, и форму стрелки при создании связей.

Note Anchor (якорь для замечания) 

Данный инструмент позволяет соединить элемент Note с любым элементом на диаграмме, в том числе и с другим элементом Note. Нельзя соединить между собой два элемента Note Anchor. Для того чтобы присоединить замечание к элементу диаграммы, необходимо выбрать инструмент Note Anchor, при этом курсор приобретает форму вертикальной стрелки, щелкнуть по значку Note и, не отпуская кнопки мыши, «тянуть» линию до нужного значка, по достижении которого кнопку мыши отпустить.

Аналогично происходит соединение при помощи других инструментов для установки связей.

Package (пакет) 

Данный инструмент позволяет создавать пакеты, которые могут включать в себя группы элементов Use Case и в данной диаграмме может использоваться для определения более крупных сценариев

поведения объектов с дальнейшей детализацией. Причем пакеты могут включать в себя другие пакеты, что позволяет создавать значительный уровень вложенности детализации.

Use Case (сценарии поведения)



Данный инструмент позволяет создавать простые формы сценариев поведения объектов системы. Это представление работы системы с точки зрения исполнителей (actors), т.е. объектов, выполняющих в системе определенные функции.

Use Case могут отображать:

- образцы поведения для отдельных объектов системы;
- последовательность связанных транзакций, представляемых объектами или системой;
- получение некоторой информации объектами.

Создание Use Case необходимо для того, чтобы:

- формализовать требования к системе;
- организовать взаимодействие с будущими пользователями системы и экспертами предметной области;
- тестировать систему.

Actor (актер)



Данный инструмент используется для создания действующих лиц в системе. На диаграмме Use Case значком actor часто обозначают пользователей системы, для того чтобы определить задачи, выполняемые пользователями, и их взаимодействие.

Обычно значком Actor обозначают объект, который:

- взаимодействует с системой или использует систему;
- передает или принимает информацию в/из системы;
- является внешним по отношению к системе.

Actor позволяет узнать:

- кто пользуется системой;
- кто отвечает за сопровождение системы;
- внешнее аппаратное обеспечение, которое используется системой;
- другие системы, которые должны взаимодействовать с данной системой.

Unidirectional Association (однонаправленная связь) 

Данный инструмент позволяет обозначать связи между элементами. На диаграмме Use Case эти связи могут быть определены между use case и actor.

Кроме сценария поведения каждого объекта системы необходимо точно представлять взаимодействие этих объектов между собой, определение клиентов и серверов и порядка обмена сообщениями между ними. Обмен сообщениями происходит в определенной последовательности, и Sequence diagram позволяет получить отражение этого обмена во времени.

При переходе в эту диаграмму (рис. 9.18) на панели инструментов доступны значки Text Box, Note, Anchor to Item, Object, Message, Message to Self.

Object (объект)



Позволяет включить новый объект в диаграмму. Каждый объект является реализацией класса, поэтому в нем можно указать класс, на основе которого он создан.

Message (сообщение)



Позволяет создать сообщение, передаваемое от одного объекта к другому. Так как все взаимодействие в объектно-ориентированных системах осуществляется при помощи сообщений между объектами, то классы должны позволять отправку или прием сообщений.

Message to Self (сообщение самому себе)



Позволяет показать, что отправитель сообщения является одновременно и его получателем.

Return Message (возврат сообщения)



Позволяет показать, что происходит возврат управления из вызванной подпрограммы на сервере клиенту.

Destruction Marker (маркер уничтожения)



Показывает, что происходит уничтожение программного объекта.

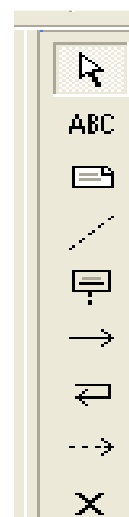



Рис. 9.18. Строка инструментов Sequence diagram

Диаграмма Statechart (диаграмма состояний) предназначена для описания состояний объекта и условий перехода между ними. Описание состояний позволяет точно описать модель поведения объекта при получении различных сообщений и взаимодействии с другими объектами.

После активизации диаграммы становятся доступны следующие инструменты (рис. 9.19).

Start State (начало) 

Инструмент Start State позволяет создать значок начала работы. Для диаграммы Statechart он обозначает событие, которое переводит объект в первое состояние на диаграмме. Все диаграммы состояний начинаются со значка Start State и должны заканчиваться значком End State. При этом значок начала работы может быть только один, а значков окончания может быть сколько угодно. За этим Rational Rose следит самостоятельно.

End State (завершение) 

Инструмент End State позволяет создать значок окончания работы. Направление перехода может быть установлено только в данный значок, однако никаких ограничений на количество переходов в End State, а также на количество таких элементов на диаграмме не налагается.

State Transition (состояние перехода) 

Инструмент State Transition позволяет создать значок состояния перехода, который означает, что объект переходит из одного состояния в другое в случае наступления определенного события или по изменению определенных условий. Пользователь может указать несколько переходов из одного состояния в другое в случае, если каждый такой переход осуществляется при наступлении разных событий или при соблюдении разных условий.

Transition To Self (переход на себя) 

Инструмент Transition To Self позволяет создать значок перехода в то же состояние, из которого осуществляется переход. Данный



Рис. 9.19. Строка инструментов диаграммы Statechart

переход похож на State Transition, однако он не осуществляет переход в другое состояние при наступлении некоторого события. Таким образом, при наступлении события оно обрабатывается, и после обработки объект возвращается в то состояние, в котором он находился до наступления события.

Class diagram (диаграмма классов) – основная диаграмма для создания кода приложения. При помощи диаграммы классов создается внутренняя структура системы, описывается наследование и взаимное положение классов друг относительно друга.

При активизации диаграммы строка инструментов приобретает следующий вид (рис. 9.20).

Class (класс) 


Данный инструмент позволяет создать новый класс в диаграмме и модели. Для детализации модели поведения классов создаются диаграммы состояний и действий.

Класс в UML нотации изображается как прямоугольник, разделенный на три части (рис. 9.21). В верхней части записывается название класса, в середине – атрибуты, в нижней части – операции.

Interface (интерфейс) 

Позволяет создать объект Interface, который указывает на видимые извне операции класса или компонента. Обычно интерфейс создается только для некоторых строго определенных классов или компонентов и предназначен скорее для логического отображения системы, но может присутствовать как на диаграмме классов, так и на диаграмме компонентов.

В диаграмме классов Interface обычно отображается как значок класса со стереотипом «interface».

Unidirectional Association (однонаправленная связь) 

Значок Unidirectional Association позволяет

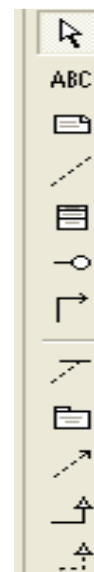


Рис. 9.20. Строка инструментов для диаграммы классов

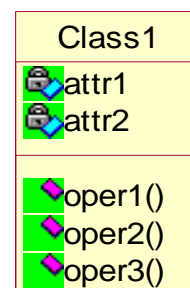


Рис. 9.21. Изображение класса

создать однонаправленную связь класса с классом или класса с интерфейсом. Это общий и самый слабый вид связи.

Association Class (ассоциация класса)



Значок Association Class позволяет связать классы ассоциативной связью. Это свойство сохраняется в классе, и для того чтобы его установить, необходимо создать класс и связать его реляцией с другим при помощи этого значка.

Package (пакет)



Позволяет создать элемент Package, который используется для физической или логической группировки элементов.

Dependency of instantiates (зависимость реализации)



Позволяет создать связь Dependency of instantiates, при этом генератор кода C++ Rational Rose создает код класса, включающий определения зависимого класса путем генерации директивы `#include`. Установка этого типа связей показывает, что класс использует другой класс как параметр в одном из методов.

Generalization (обобщение)



Позволяет создать связь Generalization, для которой Rational Rose создает код наследования, т.е. создается подкласс для соединенного этой связью класса, наследуемого из родительского класса.

Realize (выполнять)



Позволяет создать связь Realize между классом и интерфейсом или между компонентом и интерфейсом. Этот тип связи используется для того, чтобы показать, что класс выполняет операции, предоставляемые интерфейсом.

Для создания нового класса и помещения его на диаграмму классов можно воспользоваться соответствующим значком из строки инструментов или меню `Menu:Tools==>Create=>Class`. Для того чтобы поместить уже созданный класс на диаграмму классов, есть несколько путей:

- перетащить нужный класс мышкой из окна Browse;
- воспользоваться `Menu:Query=>Add Classes` и в диалоговом окне выбрать необходимые классы для включения в диаграмму.

Раздел 10

ОСНОВНЫЕ ПРАВИЛА ОФОРМЛЕНИЯ ПРОГРАММНОЙ ДОКУМЕНТАЦИИ

При оформлении текстовых и графических материалов, следует придерживаться общих требований ЕСПД.

Текстовые документы оформляются на белых листах формата А4, графический материал допускается представлять на листах формата А3. В соответствии с общими требованиями поля листа определяются следующим образом: левое – 30 мм, правое – 10 мм, верхнее – 20 мм, нижнее – 20 мм. Формат текста: Word for Windows, через полтора интервала (около тридцати строк на листе), шрифт – Times New Roman Cyr, размер шрифта – 14, отступ первой строки абзаца – 0,75 см. Количество знаков в строке, считая пробелы, – 60. Текст программы может быть расположен в две колонки, шрифт – Times New Roman Cyr, размер шрифта – 8.

Нумерация всех страниц (в том числе и приложений) сквозная. Номер проставляется в середине верхнего поля страницы арабской цифрой. Первая страница – титульный лист, вторая страница – аннотация, с третьей страницы начинается оглавление. Номера страниц на титульном листе, аннотации и оглавлении не проставляются.

Наименование разделов, подразделов, пунктов должно быть кратким и соответствовать содержанию. Каждая новая глава печатается с новой страницы. Это же правило относится и к другим основным структурным частям работы: аннотации, оглавлению, введению, заключению, списку литературы, приложениям.

Наименование разделов (основных частей) пишется прописными буквами по центру строки. Расстояние между заголовками и текстом, а также между заголовками разделов и подразделов должно быть равно двум интервалам.

Наименования подразделов и пунктов размещаются с абзацного отступа (0,75 см) и печатаются с прописной буквы, без подчеркивания и без точки в конце. Расстояние между последней строкой текста предыдущего раздела и последующим заголовком при рас-

положении их на одной странице должно быть равно трем интервалам.

Разделы и подразделы нумеруются арабскими цифрами с точкой. Разделы имеют порядковые номера 1, 2 и т.д. Номер подраздела состоит из номера раздела и порядкового номера подраздела, входящего в данный раздел, разделенных точкой (например: 1.1, 2.5). При использовании ссылок на пункты, разделы и подразделы указывается порядковый номер раздела или пункта (например: «в разд. 2», «в п. 2.3.1»).

Перечисления нумеруются арабскими цифрами со скобкой (например: 2), 3) и т. д.) с абзацного отступа. Для этого используются нумерованные списки. Допускается также применение маркированных списков одного вида.

Например:

1. ОСНОВНАЯ ЧАСТЬ

.....
.....

1.3. Анализ и уточнение требований к программному обеспечению

1.3.1. Анализ процесса обработки информации и выбор структур данных для ее хранения

.....
.....

- 1) текст текст текст текст текст ;
- 2) текст текст текст текст текст ;
- 3) текст текст текст текст текст .

.....
.....

Оформление графического материала

В соответствии с ГОСТом 2.105-79 «Общие требования к текстовым документам» иллюстрации (графики, схемы, диаграммы) могут приводиться как в основном тексте, так и в приложении. Все иллюстрации именуются рисунками.

Рисунки, таблицы и формулы нумеруются арабскими цифрами последовательно, так называемая сквозная нумерация, или в пределах раздела (относительная нумерация). В приложении нумеруются в пределах приложения.

Каждый рисунок имеет порядковый номер и название, помещаемые под рисунком по центру.

Например:

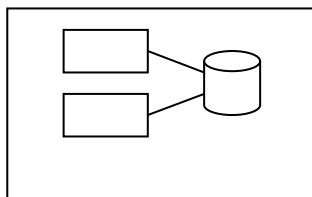
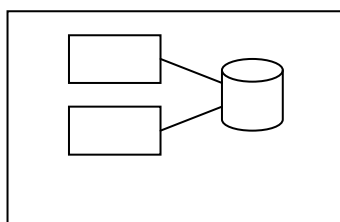


Рис. 5. Название (или)

Рис. 1.5. Название

Если рисунок занимает более одной страницы, то на всех страницах (кроме первой) проставляется порядковый номер рисунка и слово «Продолжение». Например:

1.5. Продолжение



Рисунки размещаются так, чтобы их можно было рассматривать без поворота страницы. В крайнем случае, рисунок располагается так, чтобы для просмотра нужно было повернуть страницу по часовой стрелке. Расположение и размеры полей сохраняются.

Таблицы в основном применяются для оформления цифрового материала. Шрифт – Times New Roman Cyr, размер шрифта – 14. Иногда возможен 10-й размер шрифта. Обозначения единиц физических величин необходимо применять в системе СИ. Номер таблицы размещается в правом верхнем углу над заголовком. Например:

Сравнительная характеристика звуковых плат

Название	Название	Название	Название	Название
значения	значения	значения	значения	значения

При переносе таблицы на следующую страницу необходимо пронумеровать графы и повторить их нумерацию на следующей странице.

Сравнительная характеристика звуковых плат

Название	Название	Название	Название	Название
1	2	3	4	5
значения	значения	значения	значения	значения

Эту страницу начинают с надписи «Продолжение табл.» с указанием номера таблицы. Например:

1	2	3	4	5
значения	значения	значения	значения	значения

Рисунки и таблицы размещаются сразу после абзаца, в котором они упоминаются в первый раз, или как можно ближе к этому абзацу на следующей странице.

При использовании ссылок на рисунки, таблицы и формулы они оформляются в следующем виде: «...(рис. 2)», или «...в табл. 5», или «...по формуле (9) ».

Схемы алгоритмов должны быть выполнены в соответствии с ГОСТом. Толщина сплошной линии составляет 0,6 – 1,5 мм. Надписи на схемах выполняются чертежным шрифтом, высота букв и цифр – не менее 3,5 мм.

Приводимые формулы даются без вывода, если их автором не является сам дипломник. Все формулы должны быть пронумерованы арабскими цифрами. Номер формулы ставится с правой стороны страницы в круглых скобках на уровне формулы. Значения символов и числовых коэффициентов, входящих в формулу, должны быть приведены непосредственно под формулой, начиная со слова «где» без двоеточия. Выше и ниже каждой из формул необходимо оставлять по одной свободной строчке. Формулы должны быть набраны с помощью редактора формул. Прописные и строчные буквы, надстрочные и подстрочные индексы в формулах должны быть обозначены четко. Рекомендуются следующие размеры знаков для формул: прописные буквы и цифры – 7 – 8 мм, строчные – 4 мм, показатели степеней и индексы не менее 2 мм. Например:

$$Vp_{в.т} = K_{д.р} \cdot Vp_c, \quad (2.10)$$

где $Vp_{в.т}$ – время использования вычислительной техники, ч; $K_{д.р}$ – количество дней разработки ПО; Vp_c – время работы компьютера в течение суток, ч.

Оформление приложений

Каждое приложение начинается с новой страницы с указанием в правом верхнем углу слова «ПРИЛОЖЕНИЕ» прописными буквами и имеет тематический заголовок, который располагается ниже по центру. При наличии более одного приложения все они нумеруются арабскими цифрами: ПРИЛОЖЕНИЕ 1, ПРИЛОЖЕНИЕ 2 и т.д.

Рисунки и таблицы в прил. нумеруются арабскими цифрами в пределах каждого приложения с добавлением буквы «П». Например: Рис. П1.3. Название рисунка – третий рисунок первого приложения, Таблица П1.2. Название таблицы – вторая таблица первого приложения.

Если в прил. приводится текст программы, то каждый файл оформляют как рисунок с наименованием файла и его назначением, например: Рис. П2.4. Файл `roisk.pas` – программа поиска.

Оформление списка литературы

Список литературы включает все использованные источники. В сведениях о книгах (монографиях, учебниках, пособиях, справочниках и т.д.) содержатся: фамилия и инициалы автора, название книги, место издания, издательство, год издания. При наличии трех и более авторов указывается фамилия и инициалы только первого из них со словами «и др.». Место издания приводится полностью в именительном падеже, кроме двух городов: Москва (М.) и Санкт-Петербург (СПб.).

В сведения о статье из периодического издания включаются: фамилия и инициалы автора, наименование статьи, издания (журнала), серии (если она есть), год выпуска, том (если есть), номер издания (журнала) и диапазон страниц, на которых помещена статья.

При ссылке в тексте расчетно-пояснительной записки на источник из списка литературы указывается порядковый номер по списку литературы, заключенный в квадратные скобки, например [5]. В необходимых случаях (обычно при использовании цифровых данных или цитаты) указываются и страницы, на которых помещен используемый материал, например: [5, с. 6 – 8].

Формирование списка литературы производится в порядке ссылок или в алфавитном порядке.

Например:

1. **Перроун, П. Д.** Создание корпоративных систем на базе Java 2 Enterprise Edition [Текст]: рук. разработчика: [пер. с англ.] / Поль Дж. Перроун, Венката С. Р. «Кришна», Р. Чаганти. – М. [и др.]: Вильямс, 2001. – 1179 с.

2. **ГОСТ Р 517721–2001.** Аппаратура радиоэлектронная бытовая. Входные и выходные параметры и типы соединений. Технические требования [Текст]. – Введ. 2002–01–01. – М.: Изд-во стандартов, 2001. – IV, 27 с.: ил.; 29 см.

3. **Хисамутдинов, С.Н.** Аппроксимация рядов экспериментальных данных фракталом Мандельброта / С.Н. Хисамутдинов // Известия высших учебных заведений. Черная металлургия. – 2002. – № 11. – С. 63 – 64.

Употребление сокращений

Правила сокращения слов и словосочетаний устанавливаются государственными стандартами (например, ГОСТ 7.12-77). В качестве приложения к этому ГОСТу приведен список особых случаев сокращения слов и словосочетаний, которые часто встречаются в библиографическом описании, и указаны условия их применения.

При сокращении слов используются три основных способа:

- 1) оставляется только начальная буква слова (год – г.);
- 2) оставляется часть слова (отлично – отл.);
- 3) пропускается несколько букв в середине слова, вместо которых ставится дефис (университет – ун-т).

Сокращение должно оканчиваться на согласную букву и не должно оканчиваться на гласную, на букву «й», на мягкий и твердый знак.

В дипломном проекте (работе), курсовой работе встречаются следующие виды сокращений:

- 1) буквенные аббревиатуры;
- 2) сложносокращенные слова;
- 3) условные графические сокращения по начальным буквам слова.

Первое упоминание вводимых аббревиатур указывается в круглых скобках после полного наименования, в дальнейшем они употребляются без расшифровки. Например, программное обеспечение автоматизированной системы управления (ПО АСУ).

СПИСОК ЛИТЕРАТУРЫ

Основная литература

1. Антипов С.Т., Валуйский В.Я., Панфилов В.А. Алгоритм дипломного проектирования: Учебник для вузов. – М: КолосС, 2006. – 136 с.
2. Вендров А.М. Проектирование программного обеспечения экономических информационных систем: Учебник. – М.: Финансы и статистика, 2006.
3. Единая система программной документации. – М., 1980.
4. Иванова Г.С. Технология программирования: Учебник для вузов. – М.: Изд-во МГТУ им. Н.Э. Баумана, 2006.
5. Кватрани Т. Rational Rose 2000 и UML. Визуальное моделирование. – М.: ДМК Пресс, 2001.
6. Константайн Л., Локвуд Л. Разработка программного обеспечения. – СПб., 2004.
7. Кудрявцев Е.М. Оформление дипломного проекта на ПК. – М.: ДМК Пресс, 2004. – 224 с.
8. Лачин В.И. Дипломное проектирование: Учеб. пособие. – М: Феникс, 2003. – 352 с.
9. Леоненков А.В. Самоучитель UML. – СПб.: БХВ-Петербург, 2006. – 432 с.
10. Маклаков С.В. Моделирование бизнес-процессов с BPWin 4.0. – М.: Диалог-МИФИ, 2002.
11. Маклаков С.В. Создание информационных систем с AllFusion Modeling Suite. – М.: Диалог-МИФИ, 2003.
12. Орлов С.А. Технология разработки программного обеспечения: Разработка сложных программных систем: Учебник для вузов. – СПб.: Питер, 2002. – 463 с.
13. Трофимов С. Технология разработки программного обеспечения. – М.: Финансы и статистика, 2002.
14. Трофимов С.А. CASE-технологии: практическая работа в Rational Rose. – 2-е изд. – М.: Бином-Пресс, 2002.
15. Черемных С.В. и др. Моделирование и анализ систем. IDEF-технологии: Практикум. – М.: Финансы и статистика, 2006.
16. Черемных С.В. и др. Структурный анализ систем. IDEF-технологии. – М.: Финансы и статистика, 2003.
17. Chen P.P. The Entity-Relationship Model: Toward a Unified View of Data // ACM Transactions on Database Systems. – 1976. – Vol. 1.

*Перечень основных стандартов в области обеспечения жизненного цикла
и качества программных средств*

1. ЕСПД.
2. ISO 12207:1995. (ГОСТ Р-1999). ИТ. Процессы жизненного цикла программных средств.
3. ISO 15271:1998. (ГОСТ Р-2002). ИТ. Руководство по применению ISO 12207.
4. ISO 16326:1999. (ГОСТ Р-2002). ИТ. Руководство по применению ISO 12207 при административном управлении проектами.
5. ISO 15504-1-9:1998. ТО. Оценка и аттестация зрелости процессов жизненного цикла программных средств. Ч. 1: Основные понятия и вводное руководство. Ч. 2: Эталонная модель процессов и их зрелости. Ч. 3: Проведение аттестации. Ч. 4: Руководство по проведению аттестации. Ч. 5: Модель аттестации и руководство по показателям. Ч. 6: Руководство по компетентности аттестаторов. Ч. 7: Руководство по применению при усовершенствовании процессов. Ч. 8: Руководство по применению при определении зрелости процессов поставщика. Ч. 9: Словарь.
6. ISO 9000-3:1997. Стандарты в области административного управления качеством и обеспечения качества. Ч. 3: Руководящие положения по применению стандарта ISO 9001 при разработке, поставке и обслуживании программного обеспечения.
7. ISO 9000:2000. (ГОСТ Р-2001). Система менеджмента (административного управления) качества. Основы и словарь.
8. ISO 9001:2000. (ГОСТ Р-2001). Система менеджмента (административного управления) качества. Требования.
9. ISO 9004:2000. (ГОСТ Р-2001). Система менеджмента (административного управления) качества. Руководство по улучшению деятельности.
10. ISO 10005:1995. Административное управление качеством. Руководящие указания по программам качества.
11. ISO 10006:1997. Руководство по качеству при управлении проектом.
12. ISO 10007:1995. Административное управление качеством. Руководящие указания при управлении конфигурацией.
13. ISO 10013:1995. Руководящие указания по разработке руководств по качеству.
14. ISO 10011-1-3:1990. Руководящие положения по проверке систем качества. Ч. 1: Проверка. Ч. 2: Квалификационные критерии для инспекторов-аудиторов систем качества. Ч. 3: Управление программами проверок.
15. ISO 9126:1991. (ГОСТ-1993). ИТ. Оценка программного продукта. Характеристики качества и руководство по их применению.

16. ISO 14598-1-6:1998–2000. Оценивание программного продукта. Ч. 1: Общий обзор. Ч. 2: Планирование и управление. Ч. 3: Процессы для разработчиков. Ч. 4: Процессы для покупателей. Ч. 5: Процессы для оценщиков. Ч. 6: Документирование и оценивание модулей.

17. ISO 9126-1-4. (проекты). ИТ. Качество программных средств. Ч. 1: Модель качества. Ч. 2: Внешние метрики. Ч. 3: Внутренние метрики. Ч. 4: Метрики качества в использовании.

18. ISO 14756: 1999. ИТ. Измерение и оценивание производительности программных средств компьютерных вычислительных систем.

19. ISO 12119:1994. (ГОСТ Р-2000). ИТ. Требования к качеству и тестирование.

20. ISO 13210:1994. ИТ. Методы тестирования для измерения соответствия стандартам POSIX.

21. ANSI/IEEE 1008-1986. Тестирование программных модулей и компонентов ПС.

22. ANSI/IEEE 1012-1986. Планирование верификации и подтверждения достоверности качества (валидации) программных средств.

23. ISO 9945-1:1990 (IEEE 1003.1). ИТ. Интерфейсы переносимых операционных систем. Ч. 1: Интерфейсы систем прикладных программ (язык Си).

24. ISO 9945-2:1992 (IEEE 1003.2). ИТ. Интерфейсы переносимых операционных систем. Ч. 2: Команды управления и сервисные программы.

25. ISO 15846:1998. ТО. Процессы жизненного цикла программных средств. Конфигурационное управление программными средствами.

26. ISO 14764:1999. (ГОСТ Р – 2002). ИТ. Сопровождение программных средств.

27. ISO 15408-1-3:1999. (ГОСТ Р-2002). Методы и средства обеспечения безопасности. Критерии оценки безопасности информационных технологий. Ч. 1: Введение и общая модель. Ч. 2: Защита функциональных требований. Ч. 3: Защита требований к качеству.

28. ISO 13335-1-5:1996-1998. ИТ. ТО. Руководство по управлению безопасностью. Ч. 1: Концепция и модели обеспечения безопасности информационных технологий. Ч. 2: Планирование и управление безопасностью информационных технологий. Ч. 3: Техника управления безопасностью ИТ. Ч. 4: Селекция (выбор) средств обеспечения безопасности. Ч. 5: Безопасность внешних связей.

29. ISO 10181:1-7. ВОО. 1996 – 1998. Структура работ по безопасности в открытых системах. Ч. 1: Обзор. Ч. 2: Структура работ по аутентификации. Ч. 3: Структура работ по управлению доступом. Ч. 4: Структура работ по безотказности. Ч. 5: Структура работ по конфиденциальности. Ч. 6: Структура

работ по обеспечению целостности. Ч. 7: Структура работ по проведению аудита на безопасность.

30. ISO 15910:1999 (ГОСТР-2002). ИТ. Пользовательская документация программных средств.

31. ISO 6592:1986. ОИ. Руководство по документации для вычислительных систем.

32. ISO 9294:1990. (ГОСТ 1993 г.). ТО. ИТ. Руководство по управлению документированием программного обеспечения.

33. ISO 14102:1995. ИТ. Оценка и выбор CASE-средств.

34. ISO 14471:1999. ИТ. Руководство по адаптации CASE-средств.

35. ГОСТ 34.602-89. ИТ. Техническое задание на создание автоматизированных систем.

36. ГОСТ 34.603-92. ИТ. Виды испытаний автоматизированных систем.

37. ГОСТ 34.201-89. ИТ. Виды, комплектность и обозначение документов при создании автоматизированных систем.

38. РД 50-34.698-90. Методические указания. Информационная технология. Автоматизированные системы. Требования к содержанию документов.

39. ГОСТ 28195-89. Оценка качества программных средств. Общие положения.

40. ГОСТ 28806-90. Качество программных средств. Термины и определения.

41. IEC 61508:1-6:1998-2000. Функциональная безопасность электрических/электронных и программируемых электронных систем. Ч. 3: Требования к программному обеспечению. Ч. 6: Руководство по применению стандартов IEC 61508-2 и IEC 61508-3.

42. www.IDEF.com

43. www.IDEFINE.com

44. www.INTERFACE.com

Дополнительная литература

1. Брукс Ф. Мифический человеко-месяц, или Как создаются программные системы / Пер. с англ. – СПб.: Символ, 2000.

2. Вендров А.М. CASE-технологии. Современные методы и средства проектирования информационных систем. – М.: Финансы и статистика, 1998.

3. Калянов Г.Н. CASE-структурный системный анализ (автоматизация и применение). – М.: ЛОРИ, 1996.

4. Канер С., Фолк Дж., Нгуен Е.К. Тестирование программного обеспечения. – К., 2000.

5. *Маклаков С.В.* ВРwin и ERwin: CASE-средства разработки информационных систем. – 2-е изд. – М.: ДИАЛОГ-МИФИ, 2001.

6. *Марка Д.А., МакГоуэн К.* SADT-методология структурного анализа и проектирования. – М.: Метатехнология, 1993.

7. *Орфали Р., Харки Д., Эдвардс Д.* Основы CORBA / Пер. с англ. – М.: МАЛИП, Горячая линия – Телеком, 1999.

8. *Фаулер М., Скотт К.* UML в кратком изложении. Применение стандартного языка объектного моделирования / Пер. с англ. – М.: МИР, 1999.

ПРИЛОЖЕНИЯ

Приложение 1

Титульный лист к домашней контрольной работе

МИНОБРНАУКИ РОССИИ
федеральное государственное бюджетное образовательное учреждение
высшего профессионального образования
«ЧЕРЕПОВЕЦКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

наименование института (факультета)

наименование кафедры

Контрольная работа № _____

по дисциплине

наименование дисциплины в соответствии с учебным планом

студента (студентки)

фамилия, имя, отчество

специальность

группа

Вариант № _____

Шифр зачетной книжки _____

Проверил

ФИО преподавателя

Оценка _____

Количество баллов

Подпись _____

Дата _____

Титульный лист к аудиторной контрольной работе

МИНОБРНАУКИ РОССИИ
федеральное государственное бюджетное образовательное учреждение
высшего профессионального образования
«ЧЕРЕПОВЕЦКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

наименование кафедры

Аудиторная контрольная работа № _____

по дисциплине

наименование дисциплины в соответствии с учебным планом

студента (студентки)

фамилия, имя, отчество

группа

Вариант № _____

Проверил

ФИО преподавателя

Оценка

Количество баллов

Подпись

Дата _____

Титульный лист к расчетно-графическому заданию (реферату)

МИНОБРНАУКИ РОССИИ
федеральное государственное бюджетное образовательное учреждение
высшего профессионального образования
«ЧЕРЕПОВЕЦКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

наименование института (факультета)

наименование кафедры

наименование дисциплины в соответствии с учебным планом

РАСЧЕТНО-ГРАФИЧЕСКОЕ ЗАДАНИЕ (РЕФЕРАТ)

Руководитель

ФИО преподавателя

Исполнитель

студент

группа

Фамилия, имя, отчество

Оценка

Подпись

20... год

Титульный лист к отчету по лабораторной работе

МИНОБРНАУКИ РОССИИ
федеральное государственное бюджетное образовательное учреждение
высшего профессионального образования
«ЧЕРЕПОВЕЦКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

наименование института (факультета)
наименование кафедры
наименование дисциплины в соответствии с учебным планом

ЛАБОРАТОРНАЯ РАБОТА №

Тема лабораторной работы

Руководитель

ФИО преподавателя

Исполнитель

студент

группа

Фамилия, имя, отчество

Оценка

Подпись

20... год

Титульный лист по практике

МИНОБРНАУКИ РОССИИ
федеральное государственное бюджетное образовательное учреждение
высшего профессионального образования
«ЧЕРЕПОВЕЦКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

наименование института (факультета)

наименование кафедры

ОТЧЕТ

по _____ практике

Листов ____

Студента _____ группы _____
Ф.И.О.

Место прохождения практики

Студент _____
(подпись)

Руководитель практики
от предприятия (_____) _____
(должность) (подпись, Ф.И.О.)

Руководитель практики
от кафедры ПО ЭВМ
(_____) _____
(должность) (подпись, Ф.И.О.)

20 г.

Титульный лист расчетно-пояснительной записки к курсовой работе

МИНОБРНАУКИ РОССИИ
федеральное государственное бюджетное образовательное учреждение
высшего профессионального образования
«ЧЕРЕПОВЕЦКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

наименование института (факультета)
наименование кафедры
наименование дисциплины в соответствии с учебным планом

ТЕМА КУРСОВОГО ПРОЕКТА (РАБОТЫ)

Расчетно-пояснительная записка
к курсовому проекту (работе)

Листов ____

Руководитель

ФИО преподавателя

Исполнитель

студент

группа

Фамилия, имя, отчество

Оценка

Подпись

20... год

Титульный лист технического задания к курсовой работе

МИНОБРНАУКИ РОССИИ
федеральное государственное бюджетное образовательное учреждение
высшего профессионального образования
«ЧЕРЕПОВЕЦКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

наименование института (факультета)

наименование кафедры

наименование дисциплины в соответствии с учебным планом

УТВЕРЖДАЮ
Зав. кафедрой ПО ВТ и ИС,
д. т.н., профессор _____ Ершов Е.В.
«___» _____ 20__ г.

ТЕМА КУРСОВОГО ПРОЕКТА (РАБОТЫ)

Техническое задание на курсовой проект (работу)
Листов ____

Руководитель _____
ФИО преподавателя

Исполнитель _____
студент _____
группа

Фамилия, имя, отчество

20... год

Форма титульного листа к выпускной квалификационной работе

МИНОБРНАУКИ РОССИИ
федеральное государственное бюджетное образовательное учреждение
высшего профессионального образования
«ЧЕРЕПОВЕЦКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Институт _____
Специальность (направление) _____
Выпускающая кафедра _____

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

Название работы _____

Студента _____
Ф.И.О.

ДОПУСТИТЬ К ЗАЩИТЕ

Директор института _____ (_____)
Зав. выпуск. кафедрой _____ (_____)
Руководитель выпуск. работы _____ (_____)
Консультант по _____ (_____)
Консультант по _____ (_____)
Нормоконтролер _____ (_____)
Выпускник _____ (_____)

Череповец
_____г.

Форма задания на выпускную квалификационную работу

МИНОБРНАУКИ РОССИИ
федеральное государственное бюджетное образовательное учреждение
высшего профессионального образования
«ЧЕРЕПОВЕЦКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

ЗАДАНИЕ

на выпускную квалификационную работу студента _____
шифр

Специальность (направление)

фамилия, имя, отчество

1. Тема работы: _____

(утверждена приказом по университету от «____» _____ № _____)

2. Срок сдачи студентом законченной работы: _____

3. Исходные данные: _____

4. Содержание расчетно-пояснительной записки (перечень вопросов, подлежащих разработке):

№	Наименование разделов и подразделов	Срок выполнения	
		по плану	фактически
1	2	3	4
	Введение		
1.	Основная часть		
1.2.			
1.3.			
...			

Продолжение табл.

1	2	3	4
...			
...			
	Заключение		

5. Содержание графической части:

№	Наименование плаката	Формат

6. Руководитель работы: _____

место работы

должность

фамилия, имя, отчество

подпись

7. Консультанты (по разделам содержания):

Технико-экономическое обоснование разработки

должность, фамилия, имя, отчество

подпись

Анализ мероприятий по безопасности жизнедеятельности и обеспечению экологической чистоты

должность, фамилия, имя, отчество

подпись

8. Дата выдачи задания: _____

Руководитель _____

подпись

9. Задание принял к исполнению: _____

дата подпись студента

**Форма задания по разделу
«Технико-экономическое обоснование разработки»**

ЗАДАНИЕ

по разделу «Технико-экономическое обоснование разработки»

студента _____ специальности 230105 «Программное обеспечение вычислительной техники и автоматизированных систем»

1. Содержание задания: рассчитать количество разработчиков, фонд заработной платы, стоимость 1 машино-часа, себестоимость и цену разработки, а также годовую экономию эксплуатационных расходов потребителя, срок окупаемости и годовой экономический эффект.

2. Перечень материалов и срок выполнения.

№	Наименование подраздела	Срок выполнения	
		по плану	фактически
2.1.			
2.2.			
2.3.			

3. Рекомендуемая литература:

а) Белов А.М., Добрин Г.Н., Карлик А.Е. Экономика организации (предприятия): практикум: Учеб. пособие для вузов. – М.: ИНФРА-М, 2003. – 272 с.

б) Киселева А.В., Кудряшова Ю.В. Экономика предприятия: Учеб.-метод. пособие. – Череповец: ЧГУ, 2001.

4. Дата выдачи задания: «__» _____ 20__ г.

Консультант _____
должность,
Ф.И.О. подпись

Задание принял к исполнению «__» _____ 20__ г. _____
подпись студента

Форма задания по разделу «Анализ мероприятий по безопасности жизнедеятельности и обеспечению экологической чистоты»

ЗАДАНИЕ

по разделу «Анализ мероприятий по безопасности жизнедеятельности и обеспечению экологической чистоты»

студента _____ специальности 230105 «Программное обеспечение вычислительной техники и автоматизированных систем»

1. Содержание задания: провести анализ опасных и вредных производственных факторов, разработать мероприятия по их снижению, оценить электробезопасность и пожаробезопасность помещения, а также проанализировать все возможные неполадки и аварийные ситуации при работе с данным программным обеспечением. Обосновать наличие или отсутствие мероприятий по обеспечению экологической чистоты.

2. Перечень материалов и срок выполнения.

№	Наименование подразделов и пунктов	Срок выполнения	
		по плану	фактически
3.1.	Анализ мероприятий по безопасности жизнедеятельности		
3.1.1.	Общая характеристика безопасности		
3.1.2.	Анализ вредных производственных факторов и мероприятия по их снижению		
3.1.3.	Анализ опасных производственных факторов и мероприятия по их снижению		
3.1.4.	Оценка электробезопасности помещения и мероприятия по повышению электробезопасности		
3.1.5.	Оценка помещения по пожарной безопасности и мероприятия по повышению пожаробезопасности		
3.1.6.	Анализ неполадок и аварийных ситуаций		
3.2.	Анализ мероприятий по обеспечению экологической чистоты		

3. Рекомендуемая литература:

а) Микрюков В.Ю. Безопасность жизнедеятельности: Учебник. – М.: ФОРУМ, 2008. – 464 с.

б) Гуляев Б.В. Дипломное проектирование: раздел «Безопасность жизнедеятельности»: Учеб.-метод. пособие. – Череповец: ЧГУ, 2002.

4. Дата выдачи задания: «__» _____ 20__ г.

Консультант _____

должность,

Ф.И.О. подпись

Задание принял к исполнению «__» _____ 20__ г. _____

подпись студента

Приложение 12

Форма отзыва руководителя выпускной квалификационной работы

МИНОБРНАУКИ РОССИИ

федеральное государственное бюджетное образовательное учреждение

высшего профессионального образования

«ЧЕРЕПОВЕЦКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Институт _____

Специальность (направление) _____

Выпускающая кафедра _____

**ОТЗЫВ РУКОВОДИТЕЛЯ
ВЫПУСКНОЙ КВАЛИФИКАЦИОННОЙ РАБОТЫ**

Студент _____

фамилия, имя, отчество

_____ группа

Тема _____

В отзыве следует указать задачи, поставленные перед выпускником (как справился с их решением, в какой мере проявлялись самостоятельность и

This image shows a blank sheet of white paper with horizontal ruling lines. The lines are evenly spaced and run across the width of the page. There are no margins, text, or other markings on the paper.

Дата «_____» _____ 20__ г. Подпись _____

Форма рецензии на выпускную квалификационную работу

МИНОБРНАУКИ РОССИИ
федеральное государственное бюджетное образовательное учреждение
высшего профессионального образования
«ЧЕРЕПОВЕЦКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Институт _____
Специальность (направление) _____
Выпускающая кафедра _____

**РЕЦЕНЗИЯ НА ВЫПУСКНУЮ
КВАЛИФИКАЦИОННУЮ РАБОТУ**

Студент _____
_____ фамилия, имя, отчество
_____ группа

Тема _____

В рецензии следует дать общую характеристику выпускной квалификационной работы (ее достоинства, недостатки, ошибки в расчетах, определениях, формулировках и др.); качество изложения и оформления; степень использования выпускником современных источников, технологий, последних достижений науки и техники.

This image shows a blank sheet of white paper with horizontal ruling lines. The lines are evenly spaced and run across the width of the page. There are no margins, text, or other markings on the paper.

Выводы: _____

Общая оценка работы: _____

Рецензент:

Ф.И.О. _____

Место работы _____

Должность _____

Дата «_____» _____ 20__ г. Подпись _____

Форма аннотации выпускной квалификационной работы

МИНОБРАЗОВАНИЯ РОССИИ
федеральное государственное бюджетное образовательное учреждение
высшего профессионального образования
«ЧЕРЕПОВЕЦКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

АННОТАЦИЯ

выпускной квалификационной работы

по теме:

Студент _____

фамилия, имя, отчество

ПОДПИСЬ

Руководитель работы _____

место работы

ДОЛЖНОСТЬ

ДОЛЖНОСТЬ

фамилия, имя, отчество

фамилия, имя, отчество

В аннотации следует перечислить основные вопросы, которые рассматривались; результаты работы.

[illegible]

**Пример оглавления ВКР бакалавра
(при использовании структурного подхода)**

Введение

Основная часть

1. Сравнительный анализ отечественных и зарубежных аналогов проектируемой системы
2. Выбор технологии, среды и языка программирования
3. Анализ процесса обработки информации, выбор структур данных для ее хранения, выбор методов и алгоритмов решения задачи
4. Разработка спецификаций проектируемой системы
 - 4.1. Построение функциональных диаграмм
 - 4.2. Построение диаграмм описания процессов
 - 4.3. Построение диаграмм потоков данных
 - 4.4. Проектирование структур данных и построение диаграмм отношений компонентов данных
 - 4.5. Построение диаграмм переходов состояний
5. Проектирование системы
 - 5.1. Проектирование структуры системы
 - 5.2. Разработка функциональной схемы
 - 5.3. Разработка модульной структуры
 - 5.3.1. Модульное описание программного обеспечения
 - 5.3.2. Описание взаимосвязей и взаимодействий модулей
 - 5.3.3. Описание интерфейсов модулей
 - 5.3.4. Спецификации модулей системы
6. Проектирование интерфейса пользователя
 - 6.1. Построение графа диалога
 - 6.2. Разработка форм ввода-вывода информации
7. Выбор стратегии тестирования, разработка тестов, программа и методика испытаний
 - 7.1. Объект и цель испытаний
 - 7.2. Требования к информационному, аппаратно-программному обеспечению и документации
 - 7.3. Состав, порядок и методы испытаний
 - 7.4. Результаты проведения испытаний

Заключение

Список литературы

Приложение 1. Техническое задание

Приложение 2. Схемы и/или диаграммы

Приложение 3. Текст программы

Приложение 4. Спецификация

Приложение 5. Руководство пользователя

Приложение 6. Наборы тестовых данных и результатов тестирования

**Оглавление ВКР бакалавра
(при использовании объектного подхода)**

Введение

Основная часть

1. Сравнительный анализ отечественных и зарубежных аналогов проектируемой системы
2. Выбор технологии, среды и языка программирования
3. Анализ процесса обработки информации, выбор структур данных для ее хранения, выбор методов и алгоритмов решения задачи
4. Разработка спецификаций проектируемой системы
 - 4.1. Построение диаграмм вариантов использования
 - 4.2. Построение контекстных диаграмм классов
 - 4.3. Построение диаграмм последовательностей системы
 - 4.4. Построение диаграмм деятельности сценариев вариантов использования
 - 4.5. Проектирование структур данных и построение диаграмм отношений компонентов данных
 - 4.6. Построение диаграмм переходов состояний
5. Проектирование системы
 - 5.1. Проектирование структуры системы и построение диаграмм пакетов
 - 5.2. Проектирование классов
 - 5.2.1. Построение исходных диаграмм классов в пакетах
 - 5.2.2. Построение диаграмм последовательностей действий
 - 5.2.3. Построение диаграмм кооперации
 - 5.2.4. Построение уточненных диаграмм классов в пакетах
 - 5.2.5. Детальное проектирование классов
 - 5.2.5.1. Разработка детальной структуры класса
 - 5.2.5.2. Построение диаграмм состояний объектов
 - 5.2.5.3. Построение диаграмм деятельности методов класса
 - 5.3. Построение диаграмм компонентов
 - 5.4. Построение диаграмм размещения
6. Проектирование интерфейса пользователя
 - 6.1. Построение графа диалога

- 6.2. Разработка форм ввода-вывода информации
- 7. Выбор стратегии тестирования, разработка тестов, программа и методика испытаний
 - 7.1. Объект и цель испытаний
 - 7.2. Требования к информационному, аппаратно-программному обеспечению и документации
 - 7.3. Состав, порядок и методы испытаний
 - 7.4. Результаты проведения испытаний
- Заключение
- Список литературы
- Приложение 1. Техническое задание
- Приложение 2. Схемы и/или диаграммы
- Приложение 3. Текст программы
- Приложение 4. Спецификация
- Приложение 5. Руководство пользователя
- Приложение 6. Наборы тестовых данных и результатов тестирования

**Пример оглавления для ВКР специалиста
(при использовании структурного подхода)**

ОГЛАВЛЕНИЕ

Введение

1. Основная часть
 - 1.1. Сравнительный анализ отечественных и зарубежных аналогов проектируемого объекта
 - 1.2. Выбор технологии, среды и языка программирования
 - 1.3. Анализ и уточнение требований к программному обеспечению
 - 1.3.1. Анализ процесса обработки информации и выбор структур данных для ее хранения
 - 1.3.2. Выбор методов и разработка основных алгоритмов решения задачи
 - 1.4. Разработка спецификаций программного обеспечения
 - 1.4.1. Построение функциональных диаграмм
 - 1.4.2. Построение диаграмм описания процессов
 - 1.4.3. Построение диаграмм потоков данных
 - 1.4.4. Проектирование структур данных и построение диаграмм отношений компонентов данных
 - 1.4.5. Построение диаграмм переходов состояний
 - 1.5. Проектирование программного обеспечения
 - 1.5.1. Разработка структуры программного обеспечения
 - 1.5.2. Разработка функциональной схемы
 - 1.5.3. Разработка модульной структуры программного обеспечения
 - 1.5.3.1. Модульное описание программного обеспечения
 - 1.5.3.2. Описание взаимосвязей и взаимодействий модулей
 - 1.5.3.3. Описание интерфейсов модулей
 - 1.5.3.4. Спецификации программных модулей
 - 1.5.4. Разработка программных компонентов распределенной обработки данных
 - 1.6. Проектирование интерфейса пользователя
 - 1.6.1. Построение графа диалога
 - 1.6.2. Разработка форм ввода-вывода информации
 - 1.7. Выбор стратегии тестирования, разработка тестов, программа и методика испытаний
 - 1.7.1. Объект испытаний
 - 1.7.2. Цель испытаний

- 1.7.3. Требования к программному обеспечению
- 1.7.4. Требования к программной документации
- 1.7.5. Состав и порядок испытаний
- 1.7.6. Методы испытаний
- 1.7.7. Результаты проведения испытаний
- 2. Технико-экономическое обоснование разработки
 - 2.1. Расчет себестоимости разработки
 - 2.2. Расчет цены разработки
 - 2.3. Расчет экономической эффективности
- 3. Анализ мероприятий по безопасности жизнедеятельности и обеспечению экологической чистоты
 - 3.1. Анализ мероприятий по безопасности жизнедеятельности
 - 3.1.1. Общая характеристика безопасности
 - 3.1.2. Анализ вредных производственных факторов и мероприятия по их снижению
 - 3.1.3. Анализ опасных производственных факторов и мероприятия по их снижению
 - 3.1.4. Оценка электробезопасности помещения и мероприятия по повышению электробезопасности
 - 3.1.5. Оценка помещения по пожарной безопасности и мероприятия по повышению пожаробезопасности
 - 3.1.6. Анализ неполадок и аварийных ситуаций
 - 3.2. Анализ мероприятий по обеспечению экологической чистоты
- Заключение
- Список литературы
- Приложения
 - Приложение 1. Техническое задание*
 - Приложение 2. Схемы и/или диаграммы*
 - Приложение 3. Текст программы*
 - Приложение 4. Спецификация*
 - Приложение 5. Руководство пользователя*
 - Приложение 6. Наборы тестовых данных*

**Пример оглавления для ВКР специалиста
(при использовании объектного подхода)**

ОГЛАВЛЕНИЕ

Введение

1. Основная часть
 - 1.1. Сравнительный анализ отечественных и зарубежных аналогов проектируемого объекта
 - 1.2. Выбор технологии, среды и языка программирования
 - 1.3. Анализ и уточнение требований к программному обеспечению
 - 1.3.1. Анализ процесса обработки информации и выбор структур данных для ее хранения
 - 1.3.2. Выбор методов и разработка основных алгоритмов решения задачи
 - 1.4. Разработка спецификаций программного обеспечения
 - 1.4.1. Построение диаграмм вариантов использования
 - 1.4.2. Построение контекстной диаграммы классов
 - 1.4.3. Построение диаграмм последовательностей системы для каждого варианта использования
 - 1.4.4. Построение диаграмм деятельности сценариев вариантов использования
 - 1.4.5. Проектирование структур данных и построение диаграмм отношений компонентов данных
 - 1.4.6. Построение диаграмм переходов состояний
 - 1.5. Проектирование программного обеспечения
 - 1.5.1. Проектирование структуры программного обеспечения и построение диаграмм пакетов
 - 1.5.2. Проектирование классов
 - 1.5.2.1. Построение исходных диаграмм классов для каждого пакета
 - 1.5.2.2. Построение диаграмм последовательностей действий
 - 1.5.2.3. Построение диаграмм кооперации
 - 1.5.2.4. Построение уточненных диаграмм классов для каждого пакета
 - 1.5.2.5. Детальное проектирование классов
 - 1.5.2.5.1. Разработка детальной структуры класса
 - 1.5.2.5.2. Построение диаграмм состояний объектов
 - 1.5.2.5.3. Построение диаграмм деятельности методов класса
 - 1.5.3. Построение диаграмм компонентов
 - 1.5.4. Построение диаграмм размещения

- 1.5.5. Разработка программных компонентов распределенной обработки данных
- 1.6. Проектирование интерфейса пользователя
 - 1.6.1. Построение графа диалога
 - 1.6.2. Разработка форм ввода-вывода информации
- 1.7. Выбор стратегии тестирования, разработка тестов, программа и методика испытаний
 - 1.7.1. Объект испытаний
 - 1.7.2. Цель испытаний
 - 1.7.3. Требования к программному обеспечению
 - 1.7.4. Требования к программной документации
 - 1.7.5. Состав и порядок испытаний
 - 1.7.6. Методы испытаний
 - 1.7.7. Результаты проведения испытаний
- 2. Техничко-экономическое обоснование разработки
 - 2.1. Расчет себестоимости разработки
 - 2.2. Расчет цены разработки
 - 2.3. Расчет экономической эффективности
- 3. Анализ мероприятий по безопасности жизнедеятельности и обеспечению экологической чистоты
 - 3.1. Анализ мероприятий по безопасности жизнедеятельности
 - 3.1.1. Общая характеристика безопасности
 - 3.1.2. Анализ вредных производственных факторов и мероприятия по их снижению
 - 3.1.3. Анализ опасных производственных факторов и мероприятия по их снижению
 - 3.1.4. Оценка электробезопасности помещения и мероприятия по повышению электробезопасности
 - 3.1.5. Оценка помещения по пожарной безопасности и мероприятия по повышению пожаробезопасности
 - 3.1.6. Анализ неполадок и аварийных ситуаций
 - 3.2. Анализ мероприятий по обеспечению экологической чистоты
- Заключение
- Список литературы
- Приложения
 - Приложение 1. Техническое задание*
 - Приложение 2. Схемы и/или диаграммы*
 - Приложение 3. Текст программы*
 - Приложение 4. Спецификация*
 - Приложение 5. Руководство пользователя*
 - Приложение 6. Наборы тестовых данных*

Примерное содержание магистерской диссертации

Введение

1. Анализ предметной области
 - 1.1. Анализ существующих технических решений (модели, методы, алгоритмы, средства)
 - 1.2. Описание предметной области как объекта (контроля, управления, прогнозирования)
 - 1.3. Постановка задачи, разработка требований к информационному и аппаратно-программному обеспечению
 - 1.4. Выводы по разделу 1
2. Теоретическое обоснование решаемой научной задачи
 - 2.1. Разработка математической модели
 - 2.2. Адаптация и тестирование математической модели
 - 2.3. Результаты моделирования
 - 2.4. Разработка (выбор) методы
 - 2.5. Разработка алгоритмов
 - 2.6. Выводы по разделу 2
3. Проектирование системы
 - 3.1. Концептуальное моделирование проектируемой системы
 - 3.1.1. Модель «черного ящика»
 - 3.1.2. Модель состава
 - 3.1.3. Модель структуры
 - 3.1.4. Модель «белого ящика»
 - 3.2. Проектирование информационного обеспечения
 - 3.2.1. Разработка информационной модели
 - 3.2.2. Разработка обобщенных структур данных
 - 3.2.3. Проектирование потоков информации в системе
 - 3.3. Проектирование программного обеспечения
 - 3.3.1. Выбор технологии проектирования, среды и языка программирования
 - 3.3.2. Разработка логической модели
 - 3.3.3. Разработка физической модели
 - 3.3.4. Разработка интерфейсов
 - 3.4. Выводы по разделу 3
4. Результаты экспериментальной проверки, предложенных технических решений
 - 4.1. Описание эксперимента и экспериментальных данных

4.2. Разработка методики настройки информационного и программного обеспечения

4.3. Результаты экспериментальной проверки модели

4.4. Перспективы применения предложенных технических решений

4.5. Выводы по разделу 4

Заключение

Список литературы

Приложение 1 Устав проекта

Приложение 2 Текст программы

Приложение 3 Руководство пользователя

Приложение 4 Наборы экспериментальных данных

Приложение 5 Акты о внедрении

Пример заполнения спецификации на программное изделие

СПЕЦИФИКАЦИЯ*

Обозначение	Наименование	Примечание
Документация		
Расчетно-пояснительная записка	«Разработка программного обеспечения»	
Приложение 2	Схемы и/или диаграммы	
Приложение 3	Текст программы	
Приложение 5	Руководство пользователя	
Приложение 6	Наборы тестовых данных	
...	...	
Комплексы**		
Компоненты		
Main.exe	Центральный модуль программы	
MathRoutines.dll	Модуль математических функций	
EngineF.doc	Модуль настройки параметров двигателя	
...	...	

* Правила оформления см. ГОСТ 19.202-78 (Переиздание (июль 1982 г.) с изменением № 1, утвержденным в сентябре 1981 г. (ИУС 11-81))

** Раздел «Комплексы» присутствует в Спецификации, когда несколько законченных программных продуктов объединяются в один глобальный программный продукт.

Оглавление

Введение	3
<i>Раздел 1. Подготовка к лекциям и семинарским занятиям</i>	<i>5</i>
1.1 Работа с учебной литературой	7
1.2 Подготовка докладов и рефератов	12
1.3 Подготовка презентации доклада.....	16
1.4 Рекомендации по использованию поисковых систем	22
<i>Раздел 2. Подготовка к практическим и лабораторным занятиям</i>	<i>31</i>
2.1 Выполнение расчетно-графических заданий.....	32
2.2 Отладка программного обеспечения	34
<i>Раздел 3. Подготовка к коллоквиумам, зачетам и экзаменам</i>	<i>49</i>
<i>Раздел 4. Применение информационных коммуникационных техно-</i> <i>логий</i>	<i>53</i>
<i>Раздел 5. Самостоятельная работа в процессе прохождения практики</i>	<i>57</i>
5.1 Учебная практика бакалавра	60
5.2 Научно-исследовательская практика бакалавра	61
5.3 Производственная практика бакалавра	63
5.4 Производственно-технологическая и преддипломная прак-	
тики специалистов	66
5.5 Педагогическая практика магистрантов	72
5.6 Научно-исследовательская практика магистранта	74
<i>Раздел 6. Самостоятельная работа в процессе выполнения научно-</i> <i>исследовательской работы, написания курсовых и выпускных работ</i>	<i>76</i>
6.1 Курсовая работа (проект)	77
6.2 Содержание и оформление технического задания	82
6.3 Выпускная квалификационная работа бакалавра	84
6.4 Выпускная квалификационная работа специалиста	92
6.5 Выпускная квалификационная работа магистра – маги-	
стерская диссертация	101
<i>Раздел 7. Методика расчета технико-экономических показателей</i>	<i>118</i>
7.1 Трудозатраты на разработку и отладку программы	119
7.2 Трудозатраты на разработку и отладку программы	124
7.3 Расчет цены программного продукта	127
7.4 Расчет экономической эффективности	127
<i>Раздел 8. Рекомендации по анализу факторов безопасности жизне-</i> <i>деятельности</i>	<i>130</i>
8.1 Анализ вредных производственных факторов и мероприя-	
тия по их снижению	131

8.2 Анализ опасных производственных факторов и мероприятия по их снижению	131
8.3 Оценка электробезопасности помещения и мероприятия по повышению электробезопасности	132
8.4 Оценка помещения по пожарной безопасности и мероприятия по повышению пожарной безопасности	132
8.5 Анализ неполадок и аварийных ситуаций	133
8.6 Анализ мероприятий по обеспечению экологической чистоты	133
<i>Раздел 9. Автоматизация дипломного проектирования с использованием case-средств</i>	<i>134</i>
9.1 BPwin	134
9.2 Erwin	142
9.3 Rational Rose	158
<i>Раздел 10. Основные правила оформления программной документации</i>	<i>168</i>
Список литературы	175
Приложения	180

Печатается в авторской редакции
Технический редактор М.Н. Авдюхова
Лицензия А № 165724 от 11.04.06 г.

Подписано в печать 11.03.12 г. Формат 60 × 84 ¹/₁₆.
Гарнитура таймс. Уч.-изд. л. 9,5. Усл. п.л. 12,2.
Тир. 4. Зак.

ФГБОУ ВПО «Череповецкий государственный университет»
162600 г. Череповец, пр. Луначарского, 5.