

Объектно–ориентированные системы программирования

Методические рекомендации
по выполнению курсовой работы для обучающихся

Оглавление

ОБЩИЕ ПОЛОЖЕНИЯ.....	5
ПОРЯДОК ВЫПОЛНЕНИЯ КУРСОВОЙ РАБОТЫ.....	6
ЗАЩИТА КУРСОВОЙ РАБОТЫ.....	7
СТРУКТУРА КУРСОВОЙ РАБОТЫ.....	8
ТРЕБОВАНИЯ К ОФОРМЛЕНИЮ КУРСОВОЙ РАБОТЫ.....	11
ОБЩИЕ ТРЕБОВАНИЯ К ПРОЕКТИРОВАНИЮ И РАЗРАБОТКЕ ПРОГРАММНОГО РЕШЕНИЯ.....	16
ПРИМЕРНЫЕ ТЕМЫ КУРСОВЫХ РАБОТ ДИСЦИПЛИНЫ "ОБЪЕКТНО - ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ".....	22
ПРИМЕРНЫЙ КАЛЕНДАРНЫЙ ПЛАН.....	24
ПРИЛОЖЕНИЕ А.....	25
ПРИЛОЖЕНИЕ Б.....	26
ПРИЛОЖЕНИЕ В.....	27
ПРИЛОЖЕНИЕ Г.....	28

Общие положения

Курсовая работа (КР) является одним из основных видов самостоятельной работы обучающихся в вузе, направленной на изучение, закрепление, углубление и обобщение знаний по дисциплинам профессиональной подготовки, освоение элементов научно-исследовательской работы и может служить основой выпускной квалификационной работы (ВКР).

Объем курсовой работы на втором курсе составляет не менее 15 и не более 35 страниц машинописного текста.

При выполнении и защите курсовой работы обучающийся должен продемонстрировать:

- владение соответствующим понятийным и терминологическим аппаратом;
- знакомство с основной литературой;
- умение выделить проблему и определить методы её решения;
- умение последовательно изложить существо рассматриваемых вопросов.

Темы курсовых работ и **задания** на их выполнение утверждаются на заседании кафедры.

Аттестация по курсовой работе по специальности производится в виде ее защиты в присутствии руководителя курсовой работы.

Порядок выполнения курсовой работы

1. Обучающийся выполняет курсовую работу по утвержденной теме под руководством преподавателя, являющегося его руководителем.

2. Научный руководитель составляет задание на курсовую работу, осуществляет ее текущее руководство, которое включает:

-систематические консультации с целью оказания научно-методической помощи обучающемуся;

- контроль за осуществлением выполнения работы;

- проверка содержания и оформления завершенной работы.

3. **Задание** на выполнение курсовой работы подписывается обучающимся, научным руководителем и утверждается на заседании кафедры. Один экземпляр выдается обучающемуся, другой остается на кафедре. В задании указываются:

- тема курсовой работы;

- краткая аннотация задания;

- срок сдачи курсовой работы на кафедру.

Защита курсовой работы

1. Выполненная курсовая работа сдается обучающимся руководителю в установленный срок. Научный руководитель дает рецензию в письменной форме с указанием сильных и слабых сторон курсовой работы и ставит предварительную оценку. Работа, не соответствующая предъявляемым требованиям, возвращается обучающемуся на доработку.

2. Курсовые работы, получившие положительный отзыв, допускаются к защите. Во время защиты докладчику дается возможность отстаивать и обосновывать свою точку зрения.

3. Порядок обсуждения курсовой работы предусматривает: ответы обучающегося на вопросы преподавателей кафедры и других лиц, присутствующих на защите, выступление научного руководителя. Право выступать с замечаниями и пожеланиями имеют все присутствующие.

4. Решение об оценке курсовой работы принимается преподавателями кафедры по результатам анализа представленной курсовой работы, доклада студента и его ответов на вопросы. Оценка по итогам защиты курсовой работы проставляется научным руководителем в ведомость и зачетную книжку обучающемуся (с указанием темы).

Критерием оценки курсовой работы являются самостоятельность и степень разработанности темы, умение пользоваться литературой, обоснованность выводов, правильность оформления курсовой работы и грамотная защита.

Курсовая работа после проверки и защиты хранится на кафедре в течение срока обучения.

Обучающийся, не сдавший и не защитивший в установленные сроки курсовую работу, не допускается к очередной экзаменационной сессии!

Структура курсовой работы

Структура курсовой работы включает три части: введение, основную часть и заключение. На этом этапе обучающимся необходимы консультации научного руководителя.

Обязательными элементами курсовой работы являются:

1. Титульный лист
2. Задание на курсовую работу
3. Содержание
4. Введение
5. Основная часть
6. Заключение
7. Список литературы
8. Приложения

Обязательные элементы курсовой работы

1. Титульный лист является первой страницей курсовой работы, включается в общую нумерацию, но номер страницы на нем не проставляется. *Образец титульного листа в Приложении А.*

2. Задание на курсовую работу является второй страницей КР, включается в общую нумерацию, но номер страницы на нем не проставляется. *Образец титульного листа в Приложении Б.*

3. Оглавление является третьей страницей, включает перечень основных элементов курсовой работы с указанием номеров страниц, с которых начинается их месторасположение, номер страницы 3 на нем проставляется. Желательно разместить на одной странице. *Образец содержания приведен в Приложении В.*

4. Введение нумеруется как 4-ая страница. Характеризует актуальность и социальную значимость рассматриваемой темы,

состояние ее разработанности в теории и практике, цель и задачи курсовой работы, обоснование выбора используемых методов, особенности курсовой работы и основное смысловое содержание ее разделов.

Чтобы осветить состояние разработки выбранной темы, составляется краткий обзор использованной литературы и источников.

После формулировки *цели* предпринимаемого исследования, следует указать конкретные *задачи*, которые предстоит решать в соответствии с этой целью. Это обычно делается в форме перечисления (изучить, описать, установить, выявить, вывести формулу, разработать и т.п.). Формулируя задачи, следует учитывать, что описание их решения должно составить содержание разделов курсовой работы.

5. Основная часть должна содержать текстовые материалы и числовые данные, таблицы, рисунки, графики, отражающие существо, методику и отдельные результаты, достигнутые в ходе выполнения курсовой работы.

Материал основной части рекомендуется делить на разделы (не менее двух разделов) и параграфы. Такое деление должно способствовать более стройному и упорядоченному изложению материала. При этом каждый пункт должен содержать законченную информацию, логически вписывающуюся в общую структуру работы и способствующую достижению ее целей.

В основной части должны освещаться следующие вопросы:

- методические аспекты решения поставленной задачи;
- общая характеристика объекта исследования;
- анализ предметной области, понятие, содержание и значение исследуемого предмета и его показателей;

-описание программного продукта, созданного в курсовой работе; - практическая реализация созданного программного проекта.

6. В заключении раскрывается значимость рассмотренных вопросов для теории и практики; приводятся выводы, характеризующие итоги проделанной работы, предложения и рекомендации по внедрению и эксплуатации и модернизации разработанного программного проекта.

7. Список литературы – это упорядоченный в алфавитнохронологической последовательности перечень библиографических описаний документальных источников информации по теме курсовой работы.

Образец оформления списка литературы приведен в Приложении Г.

8. Приложения к работе (их может быть несколько или не быть вовсе) включают материалы, которые по логике изложения неудобно размещать в составе основных разделов работы. Каждое приложение следует начинать с нового листа, в центре без абзацного отступа пишется слово Приложение А, которые обозначены русскими буквами. Приложения должны иметь общую с остальной частью курсовой работы нумерацию страниц. На все приложения в основной части курсовой работы должны быть ссылки. Приложения должны быть перечислены в оглавлении с указанием заголовков.

Требования к оформлению курсовой работы

При оформлении курсовой работы следует выдержать общие правила оформления, требования к текстовым документам, использование формул, таблиц, рисунков, сносок и других элементов, изложенные в ГОСТ

2.105-95.

Объем курсовой работы должен составлять не менее 25 листов и не более 30 листов машинописного текста.

Страницы нумеруются арабскими цифрами на середине верхнего поля страницы. Все страницы КР, включая приложения нумеруются по порядку без пропусков и повторений. Первой страницей считается титульный лист, второй страницей – задание на курсовую работу, на которых нумерация страниц не ставится, а на следующей ставится цифра «3» и т.д.

Курсовая работа должна быть напечатана (написана) только на одной стороне бумаги формата А4.

При оформлении курсовой работы на компьютере в любой версии редактора WORD устанавливаются следующие поля: левое – 25 мм, правое - 10 мм, верхнее - 20 мм, нижнее - 20 мм, абзацный отступ составляет 1,25 и должен быть одинаковым по всему тексту, размер шрифта 14 Times New Roman, межстрочный интервал – 1,5 и выравнивание по ширине страницы с автопереносом.

Текст курсовой работы содержит **заголовки**, которые не нумеруются: «Введение», «Заключение», «Список литературы», печатаются строчными (кроме заглавной буквы) буквами жирным шрифтом, по ширине строки с абзацным отступом. Расстояние между названием заголовка и последующим текстом должно быть равно трем межстрочным интервалам (то есть пропускается одна строка).

Основной текст КР разделен на **разделы**, которые нумеруются арабскими цифрами. Слово «Раздел» не пишется, но пишется номер раздела без точки, например:

1 Теоретические аспекты визуального программирования

Каждый раздел следует начинать с новой страницы. Переносы слов в разделах не допускаются, они не подчеркиваются, точка в конце не ставится. Если раздел состоит из двух предложений, их разделяют точкой.

Разделы содержат **подразделы** (параграфы), которые нумеруются арабскими цифрами в пределах каждого раздела (например: 1.1 - первый параграф первого раздела) и начинать с абзаца, по ширине, первая буква заглавная, остальные строчные. В конце не ставится точка, если не состоит из двух и более предложений, например:

1.1 Основные принципы объектно – ориентированного программирования

Расстояние между разделом и подразделом равно трем межстрочным интервалам (то есть пропускается одна строка).

Расстояние между подразделом и началом текста равно трем межстрочным интервалам (т.е. пропускается одна строка).

Расстояние между новым подразделом и последней строчкой предыдущего параграфа должно быть равно четырем межстрочным интервалам (т.е. пропускаются две строки).

Формулы должны органически вписываться в текст и не нарушать грамматической структуры текста курсовой работы. Формулы следует набирать в редакторе формул Word и располагать посередине строки, непосредственно следующей за строкой, содержащей ссылку на это выражение. Формулы, на которые имеются ссылки в тексте в пределах раздела, последовательно нумеруются с

указанием раздела через точку. Каждый номер должен быть заключен в скобки и помещен на правом поле выражения, к которому он относится. Ссылки в тексте на номер формулы дают в круглых скобках, например: «...в формуле (1.2)».

Внимание! Если в тексте содержатся латинские буквы из математических формул (это могут быть обозначения переменных, параметров, индексов и т.д.) обязательно их набирать в редакторе формул, например:

$$SMA_t = \frac{1}{n} \times \sum_{i=0}^{n-1} P_{t-i} \quad 2.1$$

Таблицы. В таблице можно представить используемые компоненты формы проекта с описанием их свойств. Таблицу оформляют в соответствии с требованиями ГОСТ 2.105.

Таблицу следует располагать в работе непосредственно после текста, в котором она упоминается впервые, или на следующей странице. На все таблицы должны быть ссылки в тексте, при ссылке следует писать слово таблица с указанием ее номера. Допускается в таблице применять размер шрифта меньший (12 пт), чем в тексте. Если таблица текстовая, то предложения в ячейках таблицы всегда должны начинаться с прописной буквы без абзацного отступа и в конце текста точка не ставится.

Каждая таблица должна иметь номер в пределах раздела и заголовок. Заголовок и слово «Таблица» начинаются с прописной буквы и в конце точка не ставится. Заголовок не подчеркивают. Название таблицы следует помещать над таблицей слева, без абзацного отступа в одну строку с ее номером через тире, например: Таблица 2.2 – Объекты формы Form2

Таблица 2.2 – Объекты формы Form2

Имя	Тип	Имя	Тип	Имя	Тип
Button1	TButton	Edit2	TEdit	Label2	TLabel
Button2	TButton	Edit3	TEdit	Label3	TLabel
Button3	TButton	Edit4	TEdit	Label4	TLabel
Button4	TButton	Edit5	TEdit	Label5	TLabel
Button5	TButton	Edit6	TEdit	Label6	TLabel
Button6	TButton	Edit7	TEdit	Label7	TLabel
Button7	TButton	TabSheet1	TTabSheet	Label8	TLabel
Button8	TButton	TabSheet2	TTabSheet	Label9	TLabel
Button9	TButton	TabSheet3	TTabSheet	StringGrid5	TStringGrid
Chart1	TChart	TabSheet4	TTabSheet	StringGrid1	TStringGrid
Chart2	TChart	TabSheet5	TTabSheet	StringGrid2	TStringGrid
PageControl1	TPageControl	TabSheet6	TTabSheet	StringGrid3	TStringGrid
Edit1	TEdit	Label1	TLabel	StringGrid4	TStringGrid

Таблицу с большим количеством строк допускается переносить на другой лист (страницу). При переносе части таблицы на другой лист слово «таблица» и ее номер указывают один раз над первой частью таблицы, над другими частями пишут слово «*Продолжение таблицы 1.1*». Заголовок помещают только над ее первой частью.

Графики, рисунки, скриншоты, диаграммы, схемы и другой графический материал оформляются **только как рисунки** и должны иметь порядковый номер и подрисуночные подписи согласно требованиям ГОСТ 2.105.

Рисунки помещаются по ходу текста сразу за ссылкой на него, или на следующей странице с соблюдением нумерации. В тексте обязательно должны быть ссылки и пояснения к приводимому рисунку. При ссылке следует писать слово "рисунок" с указанием его номера. Подрисуночные подписи и слово "Рисунок" начинаются с прописной буквы и в конце точка не ставится. Подрисуночные подписи не подчеркивают. Рисунки должны иметь нумерацию в пределах раздела арабскими цифрами. Номер и название рисунка

следует помещать внизу рисунка с выравниванием по центру, без абзацного отступа в одну строку с ее номером через тире, например:

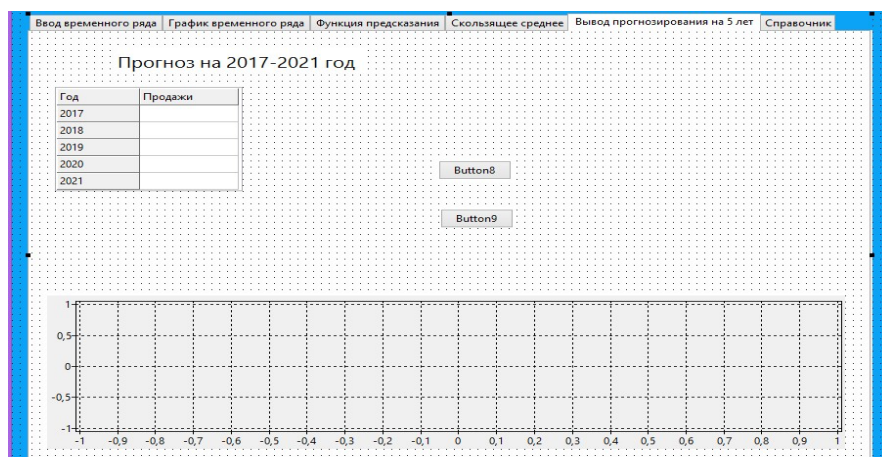


Рисунок 2.4 - Вывод прогнозирования на 5 лет

Курсовая работа в чистовом варианте должна быть оформлена в папке со скоросшивателем, либо переплетена с левой стороны.

Сокращения и условные обозначения. В тексте курсовой работы все слова должны быть написаны полностью, за исключением общепринятых обозначений. Например: т.е., и т.д., и т.п., и др., и пр.

Список литературы оформляют в соответствии с требованиями ГОСТ 7.1 и ГОСТ 7.80 и печатается как текст через 1,5 интервала по ширине, и каждое название начинается с абзаца, например :

1. Глушаков, С.В. Базы данных: учебный курс / С.В. Глушаков, Д.В. Ломотько.-Рн/Д.: Феникс, 2000. – 504 с.
2. Дейт К.Дж. Введение в системы баз данных – 6-е изд.: Пер с англ. / К.Дж. Дейт. – СПб.: Издательский дом «Вильямс», 2000 – 212 с.
3. Диго С.М. Базы данных: проектирование и использование: учебник / С.М. Диго.– М.: Финансы и статистика, 2005. – 157 с.
4. Карпова, Т.С. Базы данных: модели, разработка, реализация: учеб.

пособие / Т.С Карпова. – СПб: Питер, 2011. – 304 с.

5. Малыхина, М.П. Базы данных основы, проектирование, использование / М.П. Малыхина. –2-е изд., перераб. и доп. – СПб: Петербург, 2005.– 528 с.

Общие требования к проектированию и разработке программного решения

При выполнении работы обучающийся должен научиться составлять программы высокого качества, являющиеся легко модифицируемыми и простыми в обращении с использованием современных парадигм программирования. К этим парадигмам относятся: процедурное программирование, модульное программирование, событийно-ориентированное программирование и визуальное программирование. Также обучающийся должен продемонстрировать, закрепить и улучшить свои практические навыки проектирования и разработки программного обеспечения с использованием принципов объектно-ориентированного анализа и программирования, а также современных информационных технологий и инструментов:

- 1) формируются навыки составления технического задания на проект (ТЗ);
- 2) формируется умение проектировать дизайн графических пользовательских интерфейсов (подготовка дизайн-макетов);
- 3) закрепляются навыки использования интегрированной среды разработки;
- 4) закрепляются и углубляются знания основ программирования на современном высокоуровневом языке программирования с использованием современных парадигм;
- 5) закрепляются и углубляются знания главных принципов объектно-ориентированного программирования;
- 6) формируется умение проектировать архитектуру приложения с использованием классов (подготовка UML-диаграмм классов);

- 7) формируется умение проектировать классы для хранения и представления данных и моделей (подготовка UML-диаграмм вариантов-использования);
- 8) закрепляются и углубляются навыки разработки пользовательского интерфейса согласно дизайн-макетам;
- 9) закрепляются и углубляются навыки программирования алгоритмов обработки данных, взаимодействия с пользователем и др.;
- 10) формируются навыки составления отчета о проделанной работе и защиты результатов проекта.

Основой курсовой работы по дисциплине «Объектно-ориентированное программирование» является некоторая предметная область со своими терминами, понятиями, объектами, отношениями между этими объектами. Очевидно, что специалист в области информационных технологий далеко не всегда является специалистом в той сфере, для которой он выполняет разработку программного приложения. Поэтому залогом успешного решения поставленной перед ним задачи является подробный и качественный анализ всех аспектов той пользовательской среды, в которой будет функционировать создаваемое программное приложение. В ходе анализа предметной области необходимо на основе знакомства с литературными источниками и общения с заказчиком выявить:

1. Чему посвящена предметная область, какие в ней есть термины и понятия, субъекты и объекты, способы взаимодействия субъектов, способы использования объектов, закономерности. Например, если речь идёт о графических примитивах в трёхмерном пространстве, то следует выявить список возможных примитивов (точка, линия, прямоугольник, параллелепипед, шар и т.п.), способы их описания (так, для точки достаточно указать её координаты, а для шара необходимо знать координаты центра и радиус), возможные способы преобразования (перемещение, масштабирование, поворот и т.п.).

2. Что входит в словарь предметной области, отдельно выделив список существительных и список глаголов, которые могут быть связаны с существительными. Для графических примитивов существительными могут быть: «точка», «координата», «шар», «угол», «цвет», «длина», «ширина» и др. А в качестве глаголов можно указать: «нарисовать», «повернуть», «масштабировать», «переместить».

3. Каковы функциональные требования к разрабатываемого программного приложения. Основой их служат потребности заказчика, однако разработчик должен оценить возможность реализации требований, исходя из технических возможностей и имеющихся ресурсов. Результат анализа должен быть формализован. В реальной ситуации обычно оформляется протокол обсуждения, заключается договор, формулируется техническое задание. Все документы заверяют полномочные представители заказчика и разработчика. Во избежание конфликтных ситуаций следует задокументировать все решения, принятые по спорным моментам.

В рамках курсовой работы в роли заказчика выступает преподаватель, выдавший задание (либо представитель работодателя, если задание было сформулировано им). Обучающимися проводится анализ предметной области, основываясь на своих собственных знаниях, литературных источниках и в ходе общения с преподавателем. Результат должен быть оформлен в виде небольшого реферативного описания предметной области. Из этого описания должен логически следовать словарь предметной области, состоящий из списка существительных и глаголов. Именно он послужит основой следующего этапа работы.

Следующий этап посвящается описанию проектирования и разработки программного решения, в соответствии с темой курсовой работы (проекта). Обязательным элементом является

“Проектирование и разработка классов”. Входной информацией являются определенные на этапе анализа задач объекты, общая характеристика программы. Проектирование системы классов начинается с обработки словаря предметной области. Эта обработка состоит в выявлении того, какие слова соответствуют объектам, классам, свойствам и операциям. Список существительных служит основой для выделения классов и их свойств, а список глаголов – для определения операций.

В качестве примера для графических примитивов примера можно указать следующее соответствие:

- классы: точка, шар;
- свойства: координата, угол, цвет, длина, ширина;
- методы: нарисовать, повернуть, масштабировать, переместить.

Следующий шаг состоит в том, чтобы определить, какой из классов какие свойства и функции содержит. Следует обратить внимание на то, что эти наборы у разных классов могут «пересекаться». Например, и для класса «точка», и для класса «шар» справедливо наличие операций «нарисовать», «масштабировать», «переместить». В то же время, метод «повернуть» не имеет смысла по отношению к объектам данных классов, зато может присутствовать у класса «параллелепипед».

Ещё одним вопросом, требующим решения на данном шаге, является выявление отношений между классами. Речь идёт об отношениях наследования и включения. Следует обратить внимание, что понятие «наследование» чаще всего возникает тогда, когда разные классы обладают частично схожими наборами свойств и методов. При составлении словаря предметной области далеко не всегда в список могут попасть понятия, которым можно сопоставить базовые классы в иерархии наследования. Поэтому следует внимательно проанализировать список классов, свойств, методов, их соответствие,

и, возможно, выделить ряд новых классов, связанных с имеющимися отношениями наследования и включения. На данном этапе можно уже учитывать не только законы предметной области, но и такие принципы объектно-ориентированного подхода как абстракция, инкапсуляция, полиморфизм.

В рассматриваемом выше примере можно выделить абстрактный класс «фигура» со свойствами «абсцисса», «ордината», «аппликата», «цвет» и операцией «нарисовать». Классы «точка» и «шар» будут являться наследниками класса «фигура», а операция «нарисовать» может быть являться виртуальной, что даёт нам полиморфический кластер, включающий три класса.

Завершается этап тем, что разрабатываются полные спецификации базовых классов системы. При этом характеристики объектов вписываются в поля классов, а на основании операций разрабатываются методы и определяются способы доступа (public, private, protected). Следует иметь в виду, что все имена классов, полей и методов должны быть осмысленными, начинаться с заглавной буквы. На этапе проектирования для именования классов, полей и методов рекомендуется использовать русские слова, однако, на этапе разработки предпочтительным является английский язык.

Завершается этот этап подготовкой и представлением в этом разделе UML-диаграмм классов. После проектирования и разработки классов в тексте необходимо представить проектирование пользовательского интерфейса информационной системы.

На этом этапе разрабатываются:

- 1) структурная схема пользовательского интерфейса программы;
- 2) детали управления системой (пользовательские истории); 3) дизайн-макеты основных окон графического интерфейса.

Структурная схема позволяет выверить все детали проекта, определить взаимоотношения между отдельными частями программы, а также определяет содержание программных сообщений. На основании этой схемы в дальнейшем можно построить схему движения информационных потоков, диаграмму взаимодействия классов и т.д.

Начало данного раздела может быть, например, следующим: «В соответствии с проведенным выше анализом задачи разрабатываемая программа должна содержать: последовательное задание исходных данных для эксперимента, которые определяют условия для решения задачи; выбор вида решения и вида результатов (в соответствии с ранее определенными функциями).

После проектирования пользовательского интерфейса описывается разработка модульных элементов программы, приводится иерархическая структура приложения и схема связности модулей. Также в этом разделе нужно привести описание наиболее важных (сложных) алгоритмов, разработанных для данной информационной системы. Алгоритмы можно описать с использованием классического языка блок-схем или (более предпочтительный вариант) – с использованием нотации UML-диаграмм (activity, sequence). Завершается раздел кратким руководством пользователя по эксплуатации разработанной системы.

Исходный код разработанных программ рекомендуется включать в приложения к курсовой работе, а также в электронном виде приложить к работе на цифровом носителе информации (компакт-диск).

Примерные темы курсовых работ дисциплины «Объектно – ориентированное программирование»

1. Проект «Банковский счет»
2. Проект «Решение алгебраических уравнений n -й степени»
3. Проект «Операции с алгебраическими матрицами. Сложение матриц»
4. Проект «Операции с алгебраическими матрицами. Нахождение обратной матрицы»
5. Проект «Автоматизация рабочего места кассира по продаже билетов на поезда»
6. Проект «Оболочка для тестирования»
7. Проект «Оболочка для анкетирования»
8. Проект «Телефонный справочник»
9. Проект «Расчет стоимости электроэнергии»
10. Проект «Задача Фибоначчи»
11. Проект «Автоматическое рабочее место для работника склада»
12. Проект «Таксопарк»
13. Проект «Вывод на экран текущего каталога в графическом режиме (со скроллингом)»
14. Проект «Вычисление площади произвольного многоугольника»
15. Проект «Вычисление интеграла методом Ньютона - Котеса»
16. Проект «Вычисление затрат аварийно – диспетчерской службы предприятия»
17. Проект «Обработка экономической информации. Финансовый анализ»
18. Проект «Обработка экономической информации. Модель Альтмана»
19. Проект «Обработка экономической информации. Расчет основных экономических показателей»
20. Проект «Паспортный стол»
21. Проект «Перенос слов»
22. Проект «Морской бой»
23. Проект «Крестики - нолики»
24. Разработка программы, определяющей в тексте символов, отличные от буквы и пробела
25. Разработка программы, определяющей в тексте символов, отличные от буквы и пробела

26. Разработка программы, определяющей в тексте символов, отличных от русских гласных букв
27. Проект «Контрольная по математике. Выходной тестовый контроль
»
28. Проект «Контрольная по информатике. Выходной тестовый контроль »
29. Проект «Автоматизированная обучающая система по информатике»
30. Проект «Автоматизированная обучающая система по геометрии»
31. Проект «Вращение куба »
32. Проект «Автосалон»
33. Проект «Расписание занятий»
34. Проект «Успеваемость студентов»
35. Разработка программы, определяющей в тексте символов, отличных от знаков препинания
36. Проект «Даны файл, содержащий текст на русском языке, и некоторые буквы. Найти слово, содержащее наибольшее количество указанных букв»

ПРИМЕРНЫЙ КАЛЕНДАРНЫЙ ПЛАН

выполнения курсовой работы по дисциплине «Объектноориентированное программирование»

№ п.п.	Наименование этапов работы	Срок выполнения этапов работы
1.	Постановка задачи.	
2.	Разработка структуры программируемого проекта. Описание используемых форм с указанием структурных элементов	
3.	Разработка модулей проекта, программирование основных процедур	
4.	Работа над теоретической главой, оформление первой главы курсовой работы	
5.	Описание интерфейса проекта и этапов работы созданного проекта на Lazarus.	
6.	Оформление курсовой работы	
7.	Представление курсовой работы руководителю	
8.	Защита курсовой работы	

Руководитель _____
(подпись)

Студент _____
(подпись)

Приложение А

Образец оформления титульного листа курсовой работы

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ ФГБОУ ВО ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ
УНИВЕРСИТЕТ ИНЖЕНЕРНЫХ ТЕХНОЛОГИЙ**

**КАФЕДРА КОРПОРАТИВНЫХ ИНФОРМАЦИОННЫХ СИСТЕМ И
ПРОГРАММИРОВАНИЯ**

КУРСОВАЯ РАБОТА

по дисциплине «Объектно- ориентированные системы программирования»

Проект «Вычисление затрат аварийно – диспетчерской
службы
предприятия»

Обучающегося 2 курса

Иванова Ивана

Иванович

а по

направлен

ию

09.03.02 Информационные системы и технологии

Научный руководитель:

к.ф-м.н., доцент

И.И.

Иванов

Работа защищена с оценкой

«__» (_____)

«_____» _____20____г.

Воронеж 20__г.

Приложение Б

Образец оформления задания на курсовую работу

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ ФГБОУ ВО ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИНЖЕНЕРНЫХ ТЕХНОЛОГИЙ

Кафедра Корпоративных информационных систем и программирования

ЗАДАНИЕ НА КУРСОВУЮ РАБОТУ

Обучающийся _____
(Ф.И.О., группа)

Тема курсовой работы _____

Утверждена на заседании кафедры от _____ протокол № _____

Срок защиты работы _____

Краткая аннотация задания _____

Научный руководитель

к.ф-м.н., доцент

_____ **И.И. Иванов**

Задание принял к исполнению _____ И.О.Фамилия
(обучающегося)

Дата _____

Приложение В

Образец оформления оглавления

Содержание

	Введение.....	
1	Объектно-ориентированный анализ.....	
1.1	Принципы ООП	
1.2	Что такое Объектно-ориентированный анализ	
1.3	Среда разработки программного обеспечения Lazarus	
2	Разработка	проекта
	«Мастерская».....	2.1 Описание
	работы мастерской по ремонту компьютерной	
	техники.....	
2.2	Структура проекта «Мастерская».....	
3	Основные этапы работы приложения «Мастерская».....	
3.1	Каталог услуг	3.2
	Журнал.....	
		3.3 Система
	скидок.....	
	Заключение.....	
		.
	Список используемых источников и литературы.....	

Образец оформления списка литературы

Список литературы

1. Глушаков, С.В. Базы данных: учебный курс / С.В. Глушаков, Д.В. Ломотько.-Рн/Д.: Феникс, 2000. – 504 с.
2. Дейт К.Дж. Введение в системы баз данных – 6-е изд.: Пер с англ. / К.Дж. Дейт. – СПб.: Издательский дом «Вильямс», 2000 – 212 с.
3. Диго С.М. Базы данных: проектирование и использование: учебник / С.М. Диго.– М.: Финансы и статистика, 2005. – 157 с.
4. Карпова, Т.С. Базы данных: модели, разработка, реализация: учеб. пособие / Т.С Карпова. – СПб: Питер, 2011. – 304 с.
5. Малыхина, М.П. Базы данных основы, проектирование, использование / М.П. Малыхина. –2-е изд., перераб. и доп. – СПб: Петербург, 2005.– 528 с.