

*Министерство образования и науки Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования*

*АМУРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ*

*Факультет математики и информатики*

**Т. А. Галаган**

**Методические указания к курсовому  
проектированию по дисциплине  
«Объектно-ориентированный анализ,  
программирование»**

Благовещенск

2020

ББК  
Г

*Печатается по решению  
редакционно-издательского совета  
факультета математики и информатики  
Амурского государственного  
университета*

*Т.А. Галаган. Методические указания к курсовому проектированию по дисциплине «Объектно-ориентированный анализ, программирование» для направлений подготовки 09.03.01 Информатика и вычислительная техника, 09.03.02 Информационные системы и технологии – Благовещенск: Амурский гос. ун-т, 2020.*

Пособие предназначено для студентов второго курса бакалавриата. Оно знакомит с целями и задачами, решаемыми при выполнении курсового проектирования по дисциплине «Объектно-ориентированный анализ, программирование»; с требованиями, предъявляемыми к содержанию и оформлению пояснительной записки. Пособие также содержит краткие теоретические сведения, необходимые для выполнения курсовой работы.

*Рецензент:*

*Чалкина Н.А., канд. пед. наук, доцент кафедры общей математики и информатики ФГБОУ ВО «Амурский государственный университет».*

© Амурский государственный университет, 2020

© Кафедра информационных и управляющих систем, 2020

© Галаган Т.А., составление

## ВВЕДЕНИЕ

Курсовое проектирование является неотъемлемой частью учебного процесса. Оно направлено, прежде всего, на закрепление знаний, умений и навыков, полученных при изучении дисциплины «Объектно-ориентированный анализ, программирование», а также при изучении дисциплин «Информатика», «Программирование».

Целью дисциплины «Технология разработки программного обеспечения» является изучение этапов проектирования, разработки и тестирования и сопровождения программного обеспечения.

Задачи дисциплины: ознакомление с основными моделями жизненного цикла программного обеспечения, освоение методов и технологий разработки и модернизации программного обеспечения информационных и автоматизированных систем, формирование практических навыков самостоятельного разработки программного обеспечения информационных и автоматизированных систем, документирования, изменения и планирования требований к программному обеспечению с применением современных инструментальных средств.

Данное пособие содержит: описание целей и задач, решаемых в рамках выполнения курсовой работы; изложение требований к содержанию и оформлению материалов пояснительной записки; краткое изложение теоретических сведений, необходимых для выполнения задания. Пособие предназначения для самостоятельной работы студентов при выполнении курсового проектирования.

## ЦЕЛИ И ЗАДАЧИ КУРСОВОГО ПРОЕКТИРОВАНИЯ. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ВЫПОЛНЕНИЯ КУРСОВОЙ РАБОТЫ

Целью курсового проектирования является применение современного подхода к программированию на основе объектно-ориентированного анализа, приобретение практических навыков создания программ с использованием объектно-ориентированной парадигмы программирования.

Задачами выполнения курсового проектирования являются: изучение основных принципов парадигмы объектно-ориентированного программирования; изучение возможностей языка программирования C#, его стандартных библиотек классов и шаблонов; привитие устойчивых практических навыков применения современных инструментальных средств при разработке, отладке программного продукта и проектирование программных интерфейсов по готовому образцу или концепции интерфейса в среде программирования Visual Studio; создание, отладка и тестирование программы на языке программирования C#; составления пояснительной записки, содержащей обоснование принятых проектных решений и описание методов и способов реализации и сопровождения программного обеспечения; применения нормативных документов, регламентирующих состав, содержание и форму технической документации на разработанный программный продукт.

Выполнение курсовой работы по дисциплине «Объектно-ориентированный анализ, программирование» создает условия для практического применения знаний, полученных в области профессиональных дисциплин; приобретение навыков создания технической документации; обеспечения успеха в дальнейшей профессиональной карьере.

## СОДЕРЖАНИЕ ЭТАПОВ КУРСОВОГО ПРОЕКТИРОВАНИЯ

Курсовое проектирование включает в себя следующие этапы:

1. Формирование общего описания программного обеспечения, формулировка основных функций разрабатываемого программного обеспечения.
2. Выполнение объектно-ориентированной декомпозиции предметной области.
3. Разработка необходимых UML-диаграмм (диаграмм использования, классов, взаимодействия и др.).
4. Выполнение проекта экранных форм, необходимых для обеспечения удобного и эффективного интерфейса пользователя.
5. Выбор и обоснование паттерна проектирования (при необходимости).
6. Создание программного проекта на языке программирования C# в среде Visual Studio.
7. Отладка и тестирование созданного программного обеспечения.
8. Создание и оформление пояснительной записки к курсовой работе
9. Подготовка мультимедиа презентации по материалам работы;
10. Защита курсовой работы.

Темы курсовых работ ежегодно меняются в зависимости от предметной области, рассматриваемой студентом в рамках выполнения диссертационного исследования. Примерами тем курсовых работ являются:

Разработка автоматизированной справочной геоинформационной системы с применением объектно-ориентированного подхода.

Разработка Windows-приложения «Мониторинг погоды» с использованием паттерна проектирования «Наблюдатель»

Разработка клиент-серверного Windows-приложения для сбора информации о характеристиках сетевых устройств.

Разработка игры «Головоломка» на языке программирования C#.

Разработка программного продукта моделирования городских транс-

портных потоков с использованием объектно-ориентированного подхода.

Все формулировки тем курсовых работ можно разбить на следующие группы заданий:

1. на базе объектно-ориентированной парадигмы программирования спроектировать и разработать модель игры по индивидуальному варианту;
2. разработать систему сбора статистической информации по индивидуальному варианту с использованием объектно-ориентированного анализа;
3. использовать паттерн объектно-ориентированного проектирования для реализации программной реализации сложной системы по индивидуальному варианту;
4. на базе объектно-ориентированного анализа спроектировать и разработать Windows-приложение, реализующее модель сложной системы по индивидуальному варианту.

## ТРЕБОВАНИЯ К СОДЕРЖАТЕЛЬНОЙ ЧАСТИ ПОЯСНИТЕЛЬНОЙ ЗАПИСКИ

Пояснительная записка к курсовой работе должна состоят из следующих в изложенном порядке частей:

титульный лист;  
лист задания;  
определения, обозначения, сокращения;  
реферат  
содержание;  
введение;  
основная часть;  
заключение;  
используемые источники;  
приложения.

Основная часть работы содержит следующие обязательные разделы:  
предметная область и описание функциональных требований к программному продукту;  
объектно-ориентированная декомпозиция предметной области;  
проектирование интерфейса пользователя;  
обоснование выбора паттерна проектирования (при необходимости);  
реализация программного продукта;  
результаты тестирования разработанного программного продукта.

Общий объем пояснительной записки к курсовой работе должен составлять 30-40 страниц.

Текст пояснительной записки должен быть достаточно кратким, четким и не допускать различных толкований. В тексте записки не допускается наличие грамматических и синтаксических ошибок, поэтому рекомендуется при оформлении материала в текстовом редакторе включать режим проверки правописания.

Материал пояснительной записки должен быть логично построен, структурирован, сопровождается необходимым иллюстративным материалом. Каждый из разделов должен быть завершен выводами по разделу.

Форма титульного листа курсовой работы приведена в приложении А.

Вторым листом пояснительной записки является индивидуальное задание на курсовую работу, выданное руководителем. Форма листа «Задание» приведена в приложении Б.

Третий лист – РЕФЕРАТ. Форма данного листа приведена в приложении В.

На первых трех страницах пояснительной записки номер страницы не проставляется, но эти страницы входят в общую нумерацию документа.

Четвертым листом является содержание. Оно включает все разделы пояснительной записки: введение, наименование всех разделов, подразделов, заключение, библиографический список и наименование приложений с указанием номеров страниц, с которых начинаются эти элементы работы.

Слово «СОДЕРЖАНИЕ» записывается первой строкой страницы, выравнивание – по центру, заглавными буквами. Содержание включает введение, наименование всех разделов, подразделов, пунктов (если они имеют наименования), заключение, библиографический список и наименование приложений с указанием номеров страниц, с которых начинаются эти элементы работы.

Первым разделом является введение. Поскольку введение является самостоятельным разделом, его объем должен быть не менее одной страницы. Во введении дается краткая характеристика современного состояния изучаемой проблемы, содержатся цели и задачи курсового проектирования.

Раздел «Предметная область и описание функциональных требований к программному продукту» должен содержать: описание предметной области, в которой выполняется работа, показывается актуальность данной разработки, сфера ее применения. Далее приводится подробное описание функциона-



ла разрабатываемого программного продукта и особенности требований заказчика.

Раздел «Объектно-ориентированная декомпозиция предметной области» предполагает описание применение подходов объектно-ориентированного анализа и объектной композиции.

Объектно-ориентированный анализ – это методология, при которой требования к системе воспринимаются с точки зрения классов и объектов, выявленных в предметной области.

Объектная декомпозиция предполагает: выделение основных содержательных элементов задачи, разбиение их на типы (классы), определение свойств (данные) и поведения (методы, операции) для каждого класса, взаимодействия классов друг с другом. Объектная декомпозиция поддерживаются всеми современными объектно-ориентированными языками программирования.

Спецификация программного продукта выполняется с помощью языка графического описания для объектного моделирования UML. UML-диаграммы позволяют визуализировать, проектировать и документировать программные системы. Данное средство представления моделей имеет несколько способов описания модели в различных разрезах: диаграмму вариантов использования, диаграмму последовательности, диаграмму состояний, диаграмму активности, диаграмма компонентов, диаграмму классов и др. Непосредственно сами диаграммы в силу их объема могут быть расположены в приложениях.

Раздел «Проектирование интерфейса пользователя» начинается с анализа требований к будущему программному продукту. Затем следует описание дизайна экранных форм и описание возможностей и средств их реализации.

Раздел «Обоснование выбора паттерна проектирования» содержит непосредственное название, описание принципов выбранного шаблона про-

ектирования, особенностей его реализации в проекте.

В разделе «Реализация программного продукта» описываются алгоритмы программы; используемые методы; связи компонентов (модулей) программы с другими; приводится описание логической структуры программы с учетом текста программы на исходном языке; описание структуры программного проекта (дерево каталогов). Материал данного раздела может оперировать фрагментами кода программы. Фрагменты кода программы или ее модулей следует размещать в приложениях.

Отдельным обязательным подразделом данного раздела является «Описание классов и их экземпляров, созданных в проекте». Для удобства восприятия информации данного раздела целесообразно элементы программы свести в таблицы, например:

Таблица 1 – Элементы класса Array

<b>Поля класса</b>			
<b>Идентификатор</b>	<b>Спецификатор доступа</b>	<b>Тип</b>	<b>Смысловое содержание</b>
arr	private	int*	Указатель на выделяемую динамическую память
arr_size	private	int	Размер динамического массива
<b>Методы класса</b>			
<b>Имя</b>	<b>Описание параметров</b>	<b>Тип</b>	<b>Выполняемое действие</b>
Array	-	-	Инициализация нулем и nullptr полей класса
Array	int s – размер создаваемого массива		Создание динамического массива размера arr_size
Array	Array& copy – ссылка на другой экземпляр класса		Создание копии динамического массива
~Array	-	-	Деструктор класса
arrayrandom	int min, int max – диапазон генерации	void	Случайная генерация элементов массива в указанном диапазоне
arrayoutput	-	void	Вывод элементов массива в stdout
elementout	int i – индекс элемента массива	int	Возвращает элемент массива под индексом i
minindex	-	int	Возвращает индекс минимального элемента массива

Таблица 2 – Экземпляры класса Array

Идентификатор	Вид используемого конструктора
arr	Конструктор с параметром
arr1	Конструктор копии

В подразделе «Используемые технические средства» могут указываться типы ЭВМ и устройств, которые используются при работе программы.

В разделе «Результаты тестирования разработанного программного продукта» приводятся подготовленные тестовые значения исходных данных, а также прогнозируемые и полученные результаты работы программного продукта. Продемонстрированы возможности обработки исключительных ситуаций.

Заключение должно быть кратким и лаконичным с перечислением всех этапов выполнения задания курсового проектирования и полученных результатов. В нем следует приводить только такие выводы, которые согласуются с целью исследования, сформулированной в разделе «ВВЕДЕНИЕ».

Слово «ЗАКЛЮЧЕНИЕ» записывают в верхней части страницы, посередине прописными буквами

В библиографическом списке приводятся все литературные и нормативные источники, которые оформляются согласно правилам оформления письменных работ. На каждый источник должна быть хотя бы одна ссылка в тексте пояснительной записки.

## ТРЕБОВАНИЯ К ОФОРМЛЕНИЮ ТЕКСТА ПОЯСНИТЕЛЬНОЙ ЗАПИСКИ КУРСОВОЙ РАБОТЫ

Отчет должен быть оформлен и напечатан с использованием компьютера и принтера. Текст располагается на одной стороне листа белой бумаги формата А4. Цвет шрифта должен быть черным, размер – 14, гарнитура – Times New Roman, интервал – полуторный. Текст форматируется по ширине с включением автоматического переноса слов, интервал между абзацами не добавлять.

Размеры полей устанавливаются равными: правое – 10 мм, верхнее и нижнее – 20 мм, левое – 30 мм.

Текст пояснительной записки располагается в рамке, установленной требованиями СМК АмГУ.

Допускается исправление опечаток, ошибок и графических неточностей, обнаруженных в процессе подготовки отчета. Исправления наносятся закрашиванием белой краской, с нанесением на том же месте исправленного текста (графики) машинописным способом или черной пастой рукописным способом. Количество исправлений на одном листе – не более трех.

Наименования таких структурных элементов работы как: содержание, введение, заключение, библиографический список, приложения, следует писать по центру и прописными буквами.

Каждую структурную часть отчета следует начинать с нового листа и отделять от последующего текста двумя одинарными интервалами.

Основная часть работы делится на разделы, подразделы. Разделы, подразделы нумеруются арабскими цифрами без точки и записываются с абзацного отступа (1,25 см). Разделы должны иметь порядковую нумерацию в пределах всего текста, за исключением приложений. Номер подраздела включает номер раздела и порядковый номер подраздела, разделенные точкой (например, 2.1).

Названия разделов пишутся прописными буквами, выравнивание – по

левому краю. Названия подразделов оформляются жирными строчными буквами. Заголовок подраздела не отделяется от последующего текста пустой строкой.

Все иллюстрации пояснительной записки (чертежи, графики, схемы, компьютерные скриншоты, диаграммы, фотоснимки) называются рисунками. Рисунки следует располагать в работе непосредственно после текста, в котором они упоминаются впервые, или на следующей странице. На все иллюстрации в работе должны быть даны ссылки.

Иллюстрации должны иметь наименование. Слово «Рисунок» и наименование помещаются после самой иллюстрации с выравнением по центру следующим образом:

Рисунок 1 – Блок-схема алгоритма программы

Иллюстрации каждого приложения обозначают отдельной нумерацией арабскими цифрами с добавлением перед цифрой обозначения приложения. Например – Рисунок А.3.

При ссылках на иллюстрации следует писать «... в соответствии с рисунком 1» или «на рисунке 1 приведены ...».

Цифровой материал, как правило, оформляют в виде таблиц.

Таблицы применяют для лучшей наглядности и удобства сравнения показателей. Название таблицы, при его наличии, должно отражать ее содержание, быть точным, кратким. Название таблицы следует помещать над таблицей слева без абзачного отступа в одну строку с ее номером, через тире.

Пример оформления таблицы:

Таблица 1 – Основные переменные программы

Идентификатор	Тип	Хранимые данные
n	int	Размерность массива
X[10]	float	Одномерный массив с исходными данными
i	int	Счетчик цикла

Таблицу, в зависимости от ее размера, помещают под текстом, где впервые дана ссылка на нее, либо на следующей странице, а при необходимости – в приложении к документу. На все таблицы должны быть ссылки в работе. При ссылке следует писать слово «таблица» с указанием ее номера. Таблицу с большим количеством строк допускается переносить на другой лист (страницу). При этом слово «Таблица» и ее номер указывают один раз слева над первой частью таблицы, над другими частями пишут справа слово «Продолжение» и указывают номер таблицы. Например – «Продолжение таблицы 1». При переносе части таблицы на следующую страницу название помещают только над первой частью таблицы, нижнюю горизонтальную черту, ограничивающую таблицу, не проводят.

Библиографическое описание составляется в соответствии с ГОСТ 7.1-2003, ГОСТ 7.82-2001, ГОСТ 7.80–2000. Список нумеруется арабскими цифрами без точки. Наименования в нем располагаются в алфавитном порядке.

Примеры оформления библиографических описаний:

1 Михайлов, М. Н. Имитационное моделирование / М. Н. Михайлов, Т. В. Первозванская, Л. П. Вьюненко. – СПб: Юрайт, 2017. – 284 с.

2 Смоленцев, Н. Г. MATLAB. Программирование на C++, C#, Java и VBA / Н. Г. Смоленцев. – Москва: ДМК Пресс, 2015. – 498 с.

3 Рекомендации по выбору оборудования для работы 1С: Предприятие 8 [Электронный ресурс]. – Режим доступа: <http://v8.1c.ru/>. – 10.03.2020.

Приложения оформляют как продолжение отчета, включаются в общую нумерацию страниц. В тексте документа на все приложения должны быть ссылки. Каждое приложение следует начинать с новой страницы с указанием наверху по центру страницы слова «ПРИЛОЖЕНИЕ», его обозначения. Приложение может быть с названием и без названия. Если приложение имеет заголовок, то он записывается симметрично относительно текста, с прописной буквы отдельной строкой. Приложения обозначают заглавными буквами русского алфавита.

## ТРЕБОВАНИЯ К ОФОРМЛЕНИЮ МУЛЬТИМЕДИА ПРЕЗЕНТАЦИИ

В ходе защиты магистрант за короткое время довести до членов комиссии достаточно большой объем информации. При этом продолжительность доклада составляет 5-7 минут. Чтобы упростить восприятие работы, более четко структурировать ее изложение, сделать результаты более наглядными и убедительными, доклад необходимо сопровождать компьютерной презентацией.

Рекомендуемый объем презентации – не более 10 слайдов.

Первый слайд содержит тему курсовой работы и ФИО исполнителя. На второй слайд требуются вынести: цели и задачи работы, этапы ее выполнения, практическая значимость разработки. Большая часть материала остальных слайдов презентации – формулы, диаграммы, рисунки, схемы, экранные формы разработанного программного продукта. Количество текста на слайдах необходимо минимизировать, приводить его в тезисном варианте. практической значимости работы. Последовательность изложения материала презентации должна соответствовать последовательности изложения материала пояснительной записки. На последнем слайде приводится краткое изложение заключения, сформулированного в пояснительной записке к курсовой работе. Возможно включение в презентацию видео, демонстрирующего работу созданного программного обеспечения, но его продолжительность не должна превышать 30 секунд.

Если презентация содержит формулы, их необходимо пронумеровать. Слайды также необходимо сопроводить нумерацией.

## ЭТАПЫ РАЗРАБОТКИ ПРОГРАММНЫХ СИСТЕМ С ИСПОЛЬЗОВАНИЕМ ООП

Разработка программного обеспечения на основе объектно-ориентированного подхода включает следующие этапы: анализ и уточнение спецификаций, проектирование, реализация, модификация.

Анализ и уточнение спецификаций. Цель этапа – максимально полное описание задачи. На этом этапе выполняют анализ предметной области задачи, определяют важнейшие особенности поведения объектов (описание абстракций). По результатам анализа разрабатывают диаграмму объектов программного продукта, на которой показывают основные абстракции (объекты) предметной области и сообщения, передаваемые между ними. Все выявленные абстракции и отношения между ними описывают.

На этапе «Проектирование» различают:

- логическое проектирование, при котором принимаемые проектные решения практически не зависят от условий эксплуатации (операционной системы и используемого оборудования), заключается в разработке непосредственной структуры классов.
- физическое проектирование, включает объединение реализаций классов в модули, выбор схемы их подключения, определение способов взаимодействия с оборудованием, с операционной системой и/или другим программным обеспечением и т.д.

На этапе «Реализация системы» происходит поэтапное подключение классов к проекту. При этом начинают с создания основной программы или проекта будущего программного продукта. Затем реализуют и подключают классы так, чтобы создать первоначальный работающий прототип будущей системы. Например, таким прототипом может служить система, в которой реализован только интерфейс с пользователем (передача сообщений в отсутствующую пока часть системы не выполняется). Интерфейс тестируют и от-



лаживают. Полученный в результате работоспособный прототип продукта может быть, например, показан заказчику для уточнения требований.

На следующих этапах подключают классы, например, связанные с реализацией некоторого пункта меню. Полученный вариант системы также тестируют и отлаживают. Процесс продолжают до реализации всех возможностей системы. Использование поэтапной реализации существенно упрощает тестирование и отладку программного продукта.

Модификация. Это процесс добавления новых функциональных возможностей или изменение существующих свойств продукта. Как правило, изменения затрагивают реализацию класса, оставляя без изменения его интерфейс. Изменение интерфейса – также не очень сложная задача, но её решение может повлечь за собой необходимость согласования процессов взаимодействия объектов, что потребует изменения в других классах программы. Однако сокращение количества параметров в интерфейсной части по сравнению с модульным программированием существенно облегчает и этот процесс. Существенным также является то, что при использовании ООП объект или группа объектов могут разрабатываться отдельно, и, следовательно, их проектирование может находиться на различных этапах.

# ОБЪЕКТНАЯ ДЕКОМПОЗИЦИЯ И ОБЪЕКТНО-ОРИЕНТИРОВАННЫЙ АНАЛИЗ

Анализ предметной области необходим до начала написания программы. Объектный подход к разработке сложных программных систем предполагает необходимость этапов анализа и моделирования предметной области до непосредственного кодирования программы на каком-либо языке программирования.

Объектный подход применяется на всех основных стадиях жизненного цикла программного обеспечения и включает в себя несколько этапов.

Объектно-ориентированный анализ – методология анализа предметной области, который выполняется с целью выделения объектов и классов в качестве требования к проектируемой системе.

Объектно-ориентированное проектирование – методология проектирования, объединяющая в себе процесс декомпозиции объектов и приемы их представления логической и физической моделью проектируемой системы.

Объектно-ориентированное программирование – методология (парадигма программирования), в основе которой лежит построение программы в форме взаимодействия объектов, каждый из которых является экземпляром определенного класса.

Под объектом понимают некоторую сущность, обладающую определённым состоянием и поведением, имеющую заданные значения свойств (атрибутов) и операций над ними (методов).

Объект определяется через его внешнее отличие от других объектов. Внутренняя особенность объекта (его структура, внутренние характеристики) не влияет на внешнее отличие и для объектного моделирования значения не имеет.

Главная цель объектного анализа – представление предметной области в виде множества объектов со своими свойствами и характеристиками, кото-

рые достаточны для их определения и идентификации, а также для задания поведения объектов в рамках выбранной системы понятий и абстракций. Каждый объект – это уникальный элемент, имеющий, по крайней мере, одно свойство или характеристику и уникальную идентификацию во множестве объектов.

Предметная область может являться самостоятельным объектом или быть объектом в составе другой предметной области.

Класс представляет собой множество объектов, обладающих одинаковыми свойствами и операциями, отношениями и семантикой. Любой объект является экземпляром класса.

ПРИЛОЖЕНИЕ А  
Министерство образования и науки Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
АМУРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
(ФГБОУ ВО «АмГУ»)

Факультет математики и информатики  
Кафедра информационных и управляющих систем  
Направление подготовки 09.03.01 Информатика и вычислительная техника

**КУРСОВАЯ РАБОТА**

на тему: «\_\_\_\_\_»

по дисциплине: «Объектно-ориентированный анализ, программирование»

Исполнитель студент группы _____	_____	И.О. Фамилия
	(подпись, дата)	
Руководитель ученая степень, должность _____	_____	И.О. Фамилия
	(подпись, дата)	
Нормоконтроль ученая степень, должность _____	_____	И.О. Фамилия
	(подпись, дата)	

Благовещенск 202\_

**ПРИЛОЖЕНИЕ Б**  
**Министерство образования и науки Российской Федерации**  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
**АМУРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ**  
**(ФГБОУ ВО «АмГУ»)**

Факультет математики и информатики

Кафедра информационных и управляющих систем

Направление подготовки 09.03.01 Информатика и вычислительная техника

**З А Д А Н И Е**

К курсовой работе студента \_\_\_\_ группы \_\_\_\_\_  
(Фамилия И.О. студента)

1. Тема курсовой работы: \_\_\_\_\_

2. Срок сдачи студентом законченной работы: \_\_\_\_\_

3. Исходные данные к курсовому проектированию: предметная область, перечень литературных источников

4. Содержание курсовой работы (перечень вопросов, подлежащих разработке): предметная область и описание функциональных требований к программному продукту; объектно-ориентированная декомпозиция предметной области; проектирование интерфейса пользователя; обоснование выбора паттерна проектирования (при необходимости); разработка программного продукта; результаты тестирования разработанного программного продукта.

5. Дата выдачи задания: \_\_\_\_\_

Руководитель курсовой работы: доцент, канд. техн. наук Т.А. Галаган

Задание принял к исполнению: \_\_\_\_\_

## ПРИЛОЖЕНИЕ В

### РЕФЕРАТ

Курсовая работа содержит 42 страницы, 12 рисунков, 18 источников.

**ОБЪЕКТНО-ОРИЕНТИРОВАННЫЙ ПОДХОД, ПАТТЕРНЫ ПРОЕКТИРОВАНИЯ, UML-ДИАГРАММА, ПРОГРАММНЫЙ ПРОДУКТ, КЛАСС, ОБЪЕКТЫ, WINDOWS-FORM**

В курсовой работе выполнены: проект на основе объектно-ориентированного подхода и разработка программного продукта «Головоломка» на языке программирования С# в среде разработки Visual Studio 2019.

Целью курсовой работы является разработка, отладка и тестирование программного продукта с использованием объектно-ориентированной парадигмы программирования.

Задачами курсового проектирования являются: применение основных принципов объектно-ориентированного анализа и декомпозиции при проектировании компонент программного продукта; построение UML-диаграмм; применение возможностей языка программирования С# при разработке, отладке программного продукта; проектирование программных интерфейсов в среде разработки Visual Studio 2019; тестирование программы.

Преимуществами созданного программного продукта являются удобный интерфейс пользователя, эффективное использование памяти.

					<i>КР.175743.090301.ПЗ</i>			
<i>Изм.</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Подпись</i>	<i>Дата</i>				
<i>Разраб.</i>		<i>Дмитриева А.В.</i>			<i>Разработка игры «Головоломка» на языке программирования С#</i>	<i>Лит.</i>	<i>Лист</i>	<i>Листов</i>
<i>Проект.</i>		<i>Галаган Т.А.</i>					4	49
<i>Реценз.</i>								
<i>Н. Контр.</i>		<i>Галаган Т.А.</i>				<i>АмГУ кафедра ИиУС</i>		
<i>Зав. Каф.</i>		<i>Вишманов А.В.</i>						

## ПРИЛОЖЕНИЕ Г

### СОДЕРЖАНИЕ

Введение	5
1 Предметная область и описание функциональных требований к программному продукту	7
2 Объектно-ориентированная декомпозиция предметной области	10
3 Проектирование интерфейса пользователя	15
4 Реализация программного продукта	18
4.1 Описание классов	18
4.2 Содержание проекта	20
4.3 Тестирование программы	23
Заключение	27
Библиографический список	29
Приложение А. Экранные формы разработанного ПО	31
Приложение В. Диаграммы классов	35

					<i>КР.155497.090401.ПЗ</i>	Лист
Изм.	Листы	№ докум.	Подп.	Дата		3

## СОДЕРЖАНИЕ

ВВЕДЕНИЕ	3
ЦЕЛИ И ЗАДАЧИ КУРСОВОГО ПРОЕКТИРОВАНИЯ.	
ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ВЫПОЛНЕНИЯ КУРСОВОЙ РАБОТЫ	4
СОДЕРЖАНИЕ ЭТАПОВ КУРСОВОГО ПРОЕКТИРОВАНИЯ	5
ТРЕБОВАНИЯ К СОДЕРЖАТЕЛЬНОЙ ЧАСТИ ПОЯСНИТЕЛЬНОЙ ЗАПИСКИ	7
ТРЕБОВАНИЯ К ОФОРМЛЕНИЮ ТЕКСТА ПОЯСНИТЕЛЬНОЙ ЗАПИСКИ КУРСОВОЙ РАБОТЫ	12
ТРЕБОВАНИЯ К ОФОРМЛЕНИЮ МУЛЬТИМЕДИА ПРЕЗЕНТАЦИИ	15
ЭТАПЫ РАЗРАБОТКИ ПРОГРАММНЫХ СИСТЕМ С ИСПОЛЬЗОВАНИЕМ ООП	16
ОБЪЕКТНАЯ ДЕКОМПОЗИЦИЯ И ОБЪЕКТНО-ОРИЕНТИРОВАН- НЫЙ АНАЛИЗ	18

составитель Татьяна Алексеевна Галаган,  
*доцент кафедры ИиУС АмГУ*

Методические указания к курсовому проектированию по дисциплине «Объектно-ориентированный анализ, программирование» для направления подготовки 09.03.01 Информатика и вычислительная техника, 09.03.02 Информационные системы и технологии

---