

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение
высшего образования

«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
АЭРОКОСМИЧЕСКОГО ПРИБОРОСТРОЕНИЯ»

ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ

**Методические указания к выполнению
курсового проекта**

Специальности:

01.03.02 - Прикладная математика и информатика

02.03.03 - Математическое обеспечение и администрирование информационных систем

09.03.04 - Программная инженерия

Санкт-Петербург 2021г.

Составитель: Е.О. Шумова

Рецензент

В методических указаниях приведены варианты задания и общие методические указания для выполнения курсового проекта по дисциплине «Объектно-ориентированное программирование».

Предназначены для студентов очной и заочной форм обучения, обучающихся по направлениям подготовки 01.03.02, 02.03.03 и 09.03.04.

Подготовлены кафедрой компьютерных технологий и программной инженерии и рекомендованы к изданию редакционно-издательским советом Санкт-Петербургского государственного университета аэрокосмического приборостроения.

© ФГАОУ ВО СПбГУАП, 2021

Подписано к печати	Формат 60x84 1/16. Бумага офсетная. Печать офсетная
Усл. печ. л	Усл. кр.-отт. 0,00. Уч.- изд. л
Тираж	экз. Заказ №

Редакционно-издательский отдел
Отдел электронных публикаций и библиографии библиотеки
Отдел оперативной полиграфии
СПбГУАП
190000, Санкт-Петербург, ул. Б. Морская, 67

1. Цели и задачи курсового проектирования

Целями курсового проекта являются:

- закрепление знаний, полученных в ходе изучения дисциплины «Объектно-ориентированное программирование»;
- приобретение навыков практического программирования с использованием объектно-ориентированной парадигмы;
- подготовка к выполнению выпускной квалификационной работы.

Задачей курсового проекта является разработка иерархии типов в заданной предметной области, включающей в себя:

- разработку иерархии классов
- разработку приложения
- подготовку презентации.

2. Тематика курсового проекта

В ходе выполнения курсового проекта студенты должны практически освоить общий методологический подход, используемый при проектировании и программной реализации системы классов, соответствующей объектно-ориентированной парадигме программирования. Создаваемая система классов описывает (моделирует) определённую предметную область и может служить основой для полноценной информационной системы, решающей задачи данной области.

Спроектированная система классов должна быть не только реализована в виде программы, но и протестирована. Для тестирования классов студент должен разработать приложение, функциональные возможности которого должны демонстрировать различные варианты использования объектов созданных классов.

3. Задание на курсовое проектирование

В ходе курсового проекта необходимо разработать приложение, позволяющее протестировать взаимодействие объектов классов, спроектированных и реализованных студентом для решения конкретной задачи при заданных критериях.

При разработке программного обеспечения следует использовать *шаблоны проектирования*. Также предпочтение должно быть отдано *графическому приложению*.

Не допускается в качестве курсового проекта по объектно-ориентированному программированию разрабатывать веб-приложения, а также мобильные приложения.

При этом подлежат разработке следующие вопросы:

- а) анализ задания;
- б) разработка классов
- в) разработка тестового приложения;
- г) оформление пояснительной записки по результатам выполнения проекта
- д) создание презентации к проекту

Вариант задания на курсовое проектирование должен быть согласован с преподавателем.

Каждый из вариантов определяет предметную область, для моделирования которой должна быть разработана система классов (не менее 10 классов).

Примерный список вариантов задания на курсовое проектирование приведён в Приложении 1. Допустим выбор иных вариантов тем курсового проекта.

4. Сроки выполнения курсового проекта в семестре

Этап выполнения	Крайний срок исполнения
Согласование и утверждение темы курсового проекта с преподавателем	20 сентября 2021
Разработка структуры классов, описывающих предметную область (для своего варианта)	15 октября 2021
Разработка приложения, реализующего заданные функции	20 ноября 2021
Оформление пояснительной записки	10 декабря 2021
Подготовка презентации для защиты	20 декабря 2021
Защита курсового проекта	Согласно графика защит курсовых проектов

5. Структура и содержание курсового проекта

Курсовой проект включает в себя:

- 1) демонстрационную версию программного продукта
- 2) пояснительную записку
- 3) компьютерную презентацию.

Составными частями пояснительной записки являются:

- 1) Титульный лист
- 2) Задание на курсовое проектирование
- 3) Содержание
- 4) Введение
- 5) Основная часть
- 6) Заключение
- 7) Список использованных источников
- 8) Приложения.

Во введении обосновывается актуальность проекта, раскрываются его цель и задачи, приводится краткое описание содержания последующей основной части пояснительной записки.

Основная часть пояснительной записки включает в себя постановку задачи, проектирование классов, разработку приложения, тестирование.

Рекомендуемая структура разделов и подразделов основной части:

1. Постановка задачи
 - 1.1. Анализ предметной области
 - 1.2. Формулировка технического задания
2. Проектирование классов
 - 2.1. Классы сущностей
 - 2.2. Управляющие классы
 - 2.3. Интерфейсные классы
 - 2.4. Используемые паттерны проектирования
3. Разработка приложения
 - 3.1. Разработка интерфейса приложения
 - 3.2. Реализация классов
 - 3.3. Разработка тестового приложения
4. Тестирование

В разделе «Постановка задачи» приводится анализ предметной области на основании задания, согласованного с преподавателем, осуществляется выделение классов, выявляются требования к проекту, составляется техническое задание. В техническом задании следует перечислить все функциональные требования, которые должны быть выполнены в данном проекте.

В разделе «Проектирование классов», на основе анализа предметной области, осуществляется разработка иерархии классов (или несколько иерархий), обосновывается использование тех или иных паттернов проектирования. Результатом проектирования является представленная диаграмма классов (или несколько диаграмм).

В разделе «Разработка приложения» осуществляется формирование «структуры меню» (интерфейса приложения), выполняется реализация всех

методов и свойств, заявленных при проектировании классов. Для сложных методов приводятся алгоритмы. Также должно быть уделено внимание каждому пункту, заявленному в меню программы.

Раздел «Тестирование» содержит наглядное представление доказательства (в виде скриншотов) работоспособности программы.

Заключение должно содержать выводы по выполнению задания на проект и соответствовать введению в смысле достижения указанных в нем поставленной цели и задач проектирования. Следует отметить преимущества, связанные с реализацией проектных предложений, отметить недостатки работы, дать практические рекомендации по совершенствованию объекта проектирования, охарактеризовать перспективы дальнейшего развития работы.

6. Выполнение курсового проекта

6.1. Постановка задачи

В разделе «Постановка задачи» осуществляется анализ предметной области на основании задания, согласованного с преподавателем, определяются цели проектирования и круг решаемых задач. Используя литературные и иные источники информации по предметной области, нужно выявить:

- чем является предметная область, какие понятия и термины используются в ее рамках, какие объекты и субъекты составляют ее основу, как взаимодействуют субъекты, как используются объекты и т.п.

- из чего состоит словарь предметной области, выделив список существующих и связанных с ними глаголов. Существительные в конечном итоге определяют названия объектов и свойств, глаголы определяют возможные действия

- функциональные требования к разрабатываемому программному продукту. В их основе лежат потребности условного заказчика, которые необходимо соотнести с техническими возможностями.

Результатом анализа является составление технического задания, которое (в рамках курсового проекта) является подразделом раздела «Постановка задачи». Этот подраздел должен описывать требования к конечному продукту, очерчивать границы реализуемых функций, определять состав конечного продукта.

6.2. Проектирование классов

Следующий этап соответствует разделу «Проектирование классов» пояснительной записки. Используя принципы объектно-ориентированного анализа и проектирования, здесь следует с помощью объектно-ориентированной декомпозиции и абстрагирования выделить существующие классы и действующие объекты, свойства и методы классов, выполнить инкапсуляцию, обосновать наследование, выявить полиморфизм, определить существующие отношения между классами.

На этом этапе разрабатывается иерархия классов (или несколько иерархий классов) в виде диаграммы, определяются классы сущностей, управляющие классы и интерфейсные классы. Классы сущностей описывают сущности предметной области. Управляющие классы предназначены для организации взаимодействия классов сущностей и интерфейсных классов. Интерфейсные классы обеспечивают взаимодействие программного модуля с внешней средой, например, с пользователем. Следует привести обоснования тех или иных проектных решений, используемых паттернов проектирования.

Результатом этапа проектирования является документированное описание программного продукта.

6.3. Разработка приложения

В рамках данного раздела следует разработать интерфейс приложения (структура меню), выполнить генерацию иерархии классов на выбранном языке программирования с получением основных классов и структур данных, сформировать архитектуру программного модуля или модулей, определить алгоритмы методов. Раздел является наиболее объемным и практически важным. Здесь описывается выполнение основной части работы, связанное непосредственно с программированием и формированием структуры

программного продукта. Каждое действие в рамках разработки программного модуля (модулей) должно быть обосновано и задокументировано.

Все классы должны быть описаны с указанием свойств и методов, для сложных методов, если такие есть, должны быть приведены алгоритмы. Все действия разработчика в этой части должны выполняться в соответствии с результатами проектирования. Рекомендуется уделить внимание каждому пункту, представленному в меню приложения. Допускается вставлять небольшие фрагменты программного кода. Полный текст программных модулей приводится в приложении.

6.4. Тестирование

В данном разделе следует показать работоспособность созданного приложения, а также выявить ошибки, недочеты и недостатки. Здесь рекомендуется, в виде скриншотов и пояснений к ним, продемонстрировать все функциональные возможности приложения, определенные в меню программы.

6.5. Разработка компьютерной презентации

Презентация выполняется в Microsoft Power Point. Состоит из последовательности слайдов. Рекомендуется 10-15 слайдов. Шрифт, используемый на слайдах не должен быть меньше 28. Каждый слайд должен иметь заголовок.

Первый слайд содержит тему курсового проекта, фамилию студента. Последующие слайды посвящаются разработке проекта. Нужно представить иерархии классов, структуру программных модулей, продемонстрировать разработку интерфейса программы. Завершают презентацию слайды с результатами тестирования. Скриншоты рекомендуется использовать только при демонстрации результатов тестирования. **На слайдах не должно быть текста выступления.**

7. Защита курсового проекта

Студент должен:

- продемонстрировать работающую программу*
- оформить пояснительную записку к курсовому проекту*
- непосредственно защита с использованием презентации*

В случае получения положительного отзыва курсовой проект допускается к защите.

Защита курсового проекта включает в себя доклад студента на основе компьютерной презентации и демонстрации программного продукта.

ОЦЕНКА В ЗАЧЕТКУ ПРОСТАВЛЯЕТСЯ ТОЛЬКО ПОСЛЕ ЗАГРУЗКИ ПОЯСНИТЕЛЬНОЙ ЗАПИСКИ В ЛИЧНЫЙ КАБИНЕТ.

Примерный список вариантов задания на курсовое проектирование:

- 1) Разработка иерархии классов, реализующих графические примитивы на плоскости и операции над ними (*Графические примитивы – линии; прямоугольники; окружности, эллипсы, дуги, сегменты и секторы. Операции – поворот, перемещение и т.д.*)
- 2) Разработка иерархии классов, реализующих графические примитивы на плоскости с реализацией операций над множествами. (*Графические примитивы – прямоугольники; окружности, эллипсы. Операции над множествами – пересечение, объединение и разность*)
- 3) Разработка иерархии классов, реализующих графические примитивы в трёхмерном пространстве (Open GL или DirectX 3D) (*Графические примитивы – куб, шар, прямоугольник.*)
- 4) Разработка иерархии классов, реализующих комплексную арифметику. (*Стандартная модель комплексных чисел. Действия над комплексными числами – сложение, вычитание, умножение, деление и сравнение. Отражение истории вычислений*)
- 5) Разработка иерархии классов, реализующих операции над векторами. (*Понятие и модель вектора. Операции над векторами – сложение, разность, произведение вектора на вещественное число.*)
- 6) Разработка иерархии классов, реализующих различные типы графов и операции над ними (*Ориентированный и смешанный граф. Операции над графами: пересечение и объединение.*)
- 7) Разработка системы классов для обеспечения работы с абонентами телефонной компании (*учет абонентов, тарифов, контроль оплаты*)
- 8) Разработка системы классов для обеспечения работы деканата (*учет студентов, контроль успеваемости и т.д.*)

- 9) Разработка системы классов, описывающих сотрудников предприятия / организации с их функциями (сотрудник, менеджер, ...) .
- 10) Разработка системы классов, описывающих различные транспортные средства, реализуемые дилерским центром (*учет продаж новых автомобилей, обмен подержанных, прием старых на утилизацию и д.р.*)
- 11) Разработка системы классов для обеспечения работы библиотеки (*В системе должны поддерживаться режимы поиска книги по заданному критерию (автор, название), заказа книги, учета клиентов и книг в книгохранилище, выдачи отчетов по запросам (местонахождение книги в архиве или ее отсутствие).*
- 12) Разработка иерархии классов для обеспечения работы магазина (*В системе должны поддерживаться режимы заказа товара (продовольственных товаров), покупки и учета товаров, анализа покупаемости товара, анализа покупаемости продуктов в зависимости от времени дня и дня недели).*
- 13) Разработка системы классов для обеспечения работы гостиницы (*в системе должны поддерживаться учет постояльцев, распределения их по номерам, контроль занятости номеров, подсчет доходов и др.*)
- 14) Разработка системы классов для обеспечения работы театра (*В системе должны поддерживаться режимы учета спектаклей, актеров, играющих в спектаклях, концертов, распределения мест и стоимости билетов, анализа популярности спектаклей по различным критериям, подсчет доходов от реализации билетов)*
- 15) Разработка системы классов «Успеваемость студентов на факультете» (*В системе должны поддерживаться режимы учета учащихся и результатов сдачи экзаменов, анализа сессии по семестрам, по факультетам, специальностям, генерации отчетов отличников и двоечников.)*
- 16) Разработка системы классов для обеспечения работы центра занятости (*В системе должны поддерживаться режимы учета безработных, их стажа, квалификации, желания работать по определенной специальности, места расположения и заработной платы, учета уже стоящих на учете в центре занятости, анализа занятости от времени, специальности, учет требуемых вакансий от предприятий, учет устроившихся на работу)*

- 17) Разработка системы классов для обеспечения работы больницы (*В системе должны поддерживаться режимы учета больных по отделениям, заболеваниям, сложности заболевания и количеству заболеваний у одного человека, продолжительности болезни, количеству койко-мест и анализа заболеваемости по районам и категориям*)
- 18) Разработка системы классов для обеспечения работы школы (*В системе должны поддерживаться режимы учета классов и учеников в них, регистрации нового ученика, учета посещаемости занятий и оценок учащихся, генерации отчетов по успеваемости учеников*).
- 19) Разработка системы классов «Расписание занятий в университете» (*В системе должны поддерживаться режимы поиска занятия по заданному критерию (время, преподаватель), регистрации занятий, учета занятий по типу, генерации расписаний*)
- 20) Разработка системы классов для обеспечения работы аптеки (*В системе должны поддерживаться режимы поиска лекарства по заданному критерию (название, болезнь, цена), заказа, покупки и учета лекарств, анализа спроса на лекарства в зависимости от стоимости, времени года и т. п., выдачи соответствующих отчетов*).
- 21) Разработка системы классов для обеспечения работы ресторана (*В системе должны поддерживаться режимы заказа и поиска блюда по заданному критерию (название, цена), анализа спроса на различные блюда в зависимости от цены, времени дня, дня недели, времени года и т. п., выдачи соответствующих отчетов, генерации меню*).
- 22) Разработка системы классов для обеспечения работы поликлиники (*В системе должны поддерживаться режимы учета докторов и пациентов, поиска пациента по заданному критерию, анализа заболеваемости по районам, по месяцам*).
- 23) Разработка системы классов «Автобусный парк» (*В системе должны поддерживаться режимы учета транспортных средств в автопарке, маршрутов, водителей, учета доходов и расходов (оплата за проезд, ремонт и т. п.), выдачи отчетов по запросам*).

- 24) Разработка класса «Банковский счет» (*В системе должны поддерживаться ввод/вывод информации о держателе счета, приход и списание средств, привязка/отвязка банковской карточки к/от счету/счета и т. д.*
- 25) Игра «Snake». *Правила игры: Чтобы выиграть в Snake, вам нужно съесть все яблоки в комнате и выйти через дверь, которая откроется вверх. С каждым съеденным яблоком вы становитесь длиннее. Если вы врежетесь в стену, вы умрете. Если вы врежетесь в себя, вы умрете. Если вам в голову попадет мяч, вы умрете. Если вы слишком долго не будете есть яблоки, появятся новые.*
- 26) Игра «Шахматы». *Известная игра. Необходимо реализовать разумное подмножество международных шахматных правил, как они определены FIDE. Учить игре в шахматы компьютер не нужно, достаточно реализовать игру двух человек за одним компьютером.*
- 27) Игра «Шарики». *Разрабатываемое приложение представляет собой программную реализацию известной логической игры. Цель игры состоит в том чтобы набрать максимальное количество очков. Суть игры состоит в следующем: на игровом поле отображаются круги разного цвета, игроку предлагается создать линию из кругов одинакового цвета, расположенных по горизонтали или по вертикали, созданная линия сгорает. Причем линия может сгореть, только в том случае если количество элементов в ней равно 3. После этой процедуры пустые места на игровом поле заполняются новыми элементами. Линию можно создать путем перестановки по горизонтали или по вертикали соседних элементов. Игра заканчивается тогда, когда невозможно создать ни одной линии. Приложение должно обеспечивать возможность задания количества цветов элементов (кругов). Следует учесть, что цвета кругов выбираются произвольным образом исходя из заданного количества. Кроме того, линии (3 и более элементов), получаемые при произвольной расстановки элементов, “сгорают”.*
- 28) Игра «Тетрис» *Программа представляет собой Тетрис на поле произвольного размера. Программа должна предоставлять возможность выбирать размер фигур (4, 5, 6, 7 клеток). Скорость падения управляется автоматически в зависимости от времени игры.*

Пример оформления задания на курсовой проект

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное автономное образовательное учреждение высшего образования
«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
АЭРОКОСМИЧЕСКОГО ПРИБОРОСТРОЕНИЯ»

КАФЕДРА КОМПЬЮТЕРНЫХ ТЕХНОЛОГИЙ И ПРОГРАММНОЙ ИНЖЕНЕРИИ

**Задание
на курсовой проект по дисциплине
«Объектно-ориентированное программирование»**

Студенту группы _____
№ группы _____ Ф.И.О. _____

Тема «Разработка приложения для организации взаимодействия объектов
при заданных критериях»

Исходные данные: (указывается Ваша конкретная предметная область,
согласованная с преподавателем)

Проект должен содержать:

- анализ предметной области
- разработку классов
- разработку тестового приложения
- оформление пояснительной записки по результатам выполнения проекта
- создание презентации к проекту

Срок сдачи законченного проекта _____

Руководитель проекта _____ ст.преп. Е.О.Шумова

Дата выдачи задания 01.09.2021 г.

Пример оформления пояснительной записки

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное автономное образовательное учреждение высшего образования
«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
АЭРОКОСМИЧЕСКОГО ПРИБОРОСТРОЕНИЯ»

КАФЕДРА КОМПЬЮТЕРНЫХ ТЕХНОЛОГИЙ И ПРОГРАММНОЙ ИНЖЕНЕРИИ

КУРСОВОЙ ПРОЕКТ
ЗАЩИЩЕН С ОЦЕНКОЙ
РУКОВОДИТЕЛЬ

Ст.преподаватель		Е.О. Шумова
должность, уч. степень, звание	подпись, дата	инициалы, фамилия

**ПОЯСНИТЕЛЬНАЯ ЗАПИСКА
К КУРСОВОМУ ПРОЕКТУ**

Разработка приложения для организации взаимодействия объектов
при заданных критериях

по дисциплине: ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ

РАБОТУ ВЫПОЛНИЛ

СТУДЕНТ ГР.	№№№№		
		подпись, дата	инициалы, фамилия

Санкт-Петербург 2021

Содержание

1. Цели и задачи курсового проектирования.....	3
2. Тематика курсового проекта.....	3
3. Задание на курсовое проектирование	4
4. Сроки выполнения курсового проекта в семестре	5
5. Структура и содержание курсового проекта.....	5
6. Выполнение курсового проекта.....	7
7. Защита курсового проекта.....	10
Приложение 1	11
Приложение 2	15
Приложение 3	16