Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Макаренко Елена Николаевна
Должность: Ректор
Дата подписания: 09.09.2026 16:03:22
Уникальный программный ключ:
c098bc0c1041cb2a4cf926cf171d6715d99a6ae00adc8e27b55cbe1e2dbd7c78

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение высшего образования
«Ростовский государственный экономический университет (РИНХ)»

УТВЕРЖДАЮ
Начальник
учебно-методического управления
Т.К. Платонова
«25» мая 2026 г.

Рабочая программа дисциплины
Проектирование и конструирование программного обеспечения

Направление подготовки
09.03.04 Программная инженерия

Направленность (профиль) программы бакалавриата
09.03.04.01 Системное и прикладное программное обеспечение

Для набора 2026 года

Квалификация
Бакалавр

КАФЕДРА **Информационные технологии и программирование**

Распределение часов дисциплины по семестрам / курсам

Семестр (<Курс>.<Семестр на курсе>)	**4 (2.2)**		**5 (3.1)**		Итого	
Недель	16 3/6		16 5/6			
Вид занятий	УП	РП	УП	РП	УП	РП
Лекции	16	16	16	16	32	32
Лабораторные	16	16	16	16	32	32
Практические	16	16	16	16	32	32
Итого ауд.	48	48	48	48	96	96
Контактная работа	48	48	48	48	96	96
Сам. работа	24	24	60	60	84	84
Часы на контроль			36	36	36	36
Итого	72	72	144	144	216	216

ОСНОВАНИЕ

Учебный план утвержден учёным советом Университета (протокол № 9 от 03.03.2026 г.).

Программу составил(и): к.э.н., доцент, Жилина Е.В.

Зав. кафедрой: к.э.н., доцент Е.В. Ефимова

Методический совет: д.э.н., профессор Е.Н. Тищенко

1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

1.1	Ознакомление обучающихся с основами теории и практики в области проектирования и конструирования программного обеспечения; изучение различных методов и технологий проектирования, ознакомление с современными программными средствами, применяемыми в области проектирования и конструирования; получение практических навыков проектирования и конструирования программного обеспечения.

2. ТРЕБОВАНИЯ К РЕЗУЛЬТАТАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ

ПК-2. способен читать, понимать и выделять главную идею прочитанного исходного кода, документации

ПК-6. способен моделировать инженерные процессы, использовать формальные методы конструирования и проектирования программного обеспечения

В результате освоения дисциплины обучающийся должен:

Знать:

основы представления программного кода, структуру и нормы составления документации по программному обеспечению (соотнесено с индикатором ПК-2.1);
основы моделирования и формальные методы конструирования программного обеспечения (соотнесено с индикатором ПК-6.1).

Уметь:

выявлять основные модули в программном коде, определять главную идею прочитанного кода, сформулировать результат анализа кода (соотнесено с индикатором ПК-2.2);
использовать формальные методы конструирования программного обеспечения (соотнесено с индикатором ПК-6.2).

Владеть:

навыками определения задачи, поставленной при разработке программного кода, основных шагов алгоритма, реализованного в программе (соотнесено с индикатором ПК-2.3);
методами формализации и моделирования программного обеспечения (соотнесено с индикатором ПК-6.3).

3. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

Раздел 1. Методологические аспекты проектирования программного обеспечения

№	Наименование темы, краткое содержание	Вид занятия / работы / форма ПА	Семестр / Курс	Количество часов	Компетенции
1.1	Тема 1.1 "Структура программного проекта" Понятие программного обеспечения (ПО), программного продукта, проекта. Жизненный цикл ПО. Этапы создания ПО. Модели жизненного цикла ПО: каскадная модель; спиральная модель. Содержание и взаимосвязь процессов жизненного цикла ПО.	Лекционные занятия	4	2	ПК-2 ПК-6
1.2	Тема 1.1 "Структура программного проекта" Разработка схемы архитектуры программного проекта.	Лабораторные занятия	4	2	ПК-2 ПК-6
1.3	Тема 1.1 "Структура программного проекта" Анализ технологий проектирования. Анализ архитектуры коммерческих программных продуктов.	Практические занятия	4	2	ПК-2 ПК-6
1.4	Тема 1.2 "Качество программного проекта" Требования к эффективности и надежности проектных решений. Критерии качества программного проекта. Стандарты качества. Регламентация процессов проектирования в отечественных и международных стандартах. Обзор современных стандартов и технологий создания программного обеспечения.	Лекционные занятия	4	2	ПК-2 ПК-6
1.5	Тема 1.2 "Качество программного проекта" Расчет показателей качества программного обеспечения	Лабораторные занятия	4	2	ПК-2 ПК-6
1.6	Тема 1.2 "Качество программного проекта" Сравнительный анализ отечественных стандартов качества программного обеспечения	Практические занятия	4	2	ПК-2 ПК-6
1.7	Тема 1.3 "Каноническое проектирование" Организация канонического проектирования. Стадии проекта. Разработка модели деятельности организации ("как есть" и "как должно быть"). Декомпозиция функций. Состав функциональных подсистем, комплексов задач и задач. Описание постановки задачи. Классификация информации. Понятия и основные требования к системе. Состав и содержание операций проектирования классификаторов. Внутримашинное информационное обеспечение. Процессы проектирования первичных (входных) и результатных (выходных) документов.	Лекционные занятия	4	4	ПК-2 ПК-6
1.8	Тема 1.3 "Каноническое проектирование" Разработка IDEF0 диаграмм	Лабораторные	4	4	ПК-2

	проекта (модель "как-должно быть"). Декомпозиция работ модели. Потоки данных (входящие, выходящие, управление, механизм). Тоннельность.	занятия			ПК-6
1.9	Тема 1.3 "Каноническое проектирование" Классификация информации. Системы кодификации. Разработка системы кодирования на предприятии.	Практические занятия	4	4	ПК-2 ПК-6

Раздел 2. Современные принципы проектирования архитектуры программного обеспечения

№	Наименование темы, краткое содержание	Вид занятия / работы / форма ПА	Семестр / Курс	Количество часов	Компетенции
2.1	Тема 2.1 "Архитектура программного обеспечения" Схема Захмана архитектуры программной системы. Иерархический принцип определения архитектуры. Модульность. Компонентная технология. Методы создания и использования компонентов. Взаимодействие компонентов. Распределенные системы. Принцип открытой архитектуры (SOA). Драйверы ODBC- стандарт открытого взаимодействия баз данных. Стандарты COM, ДCOM, CORBA и др. Интеграция моделей.	Лекционные занятия	4	4	ПК-2 ПК-6
2.2	Тема 2.1 "Архитектура программного обеспечения" Разработка DFD диаграмм проекта (модель "как-должно быть"). Документооборот. Внешние сущности. Хранилища данных. Разработка IDEF3 диаграмм проекта (модель "как-должно быть"). Логика функций. Справочники.	Лабораторные занятия	4	4	ПК-2 ПК-6
2.3	Тема 2.1 "Архитектура программного обеспечения" Разработка входящих/исходящих документов (шаблонов) для программного проекта	Практические занятия	4	4	ПК-2 ПК-6
2.4	Тема 2.2 "Методы и средства проектирования программного обеспечения" Классификация технологий, методов и средств проектирования. Выбор технологии проектирования. Моделирование как методологическая основа проектирования. Виды моделей и методов моделирования. Формы описания: абстрактные объекты, конечные автоматы.	Лекционные занятия	4	4	ПК-2 ПК-6
2.5	Тема 2.2 "Методы и средства проектирования программного обеспечения" Анализ организационной структуры предприятия. Разработка Char-диаграммы, Node-Tree-диаграммы. Разработка ролей системы.	Лабораторные занятия	4	4	ПК-2 ПК-6
2.6	Тема 2.2 "Методы и средства проектирования программного обеспечения" Импорт данных из структурной модели в ER-модель. Разработка логической модели данных. Разработка физической модели данных. Генерация скриптов.	Практические занятия	4	4	ПК-2 ПК-6
2.7	Разработка структурной модели программного обеспечения. Полный комплект структурных диаграмм IDEF0, DFD, IDEF3, альтернативные IDEF3, Char-диаграмма, Node-Tree-диаграмма. ER-модель данных. Имитационное моделирование как инструмент оценки качества модели программного проекта. ABC-анализ затрат на реализацию функций программного обеспечения. Разработка библиотеки анализа ABC-затрат	Самостоятельная работа	4	24	ПК-2 ПК-6
2.8	Подготовка к промежуточной аттестации	Зачет	4	0	ПК-2 ПК-6

Раздел 3. Современные технологии проектирования программного обеспечения

№	Наименование темы, краткое содержание	Вид занятия / работы / форма ПА	Семестр / Курс	Количество часов	Компетенции
3.1	Тема 3.1 "Организация предпроектной/проектной стадии проектирования программного обеспечения" Состав работ на предпроектной стадии, стадии технического и рабочего проектирования, стадии ввода в действие программного обеспечения, эксплуатации и сопровождения. Состав проектной документации. Предпроектная стадия создания. Объекты обследования. Методы организации обследования и сбора материалов обследования. Анализ материалов обследования. Разработка технико-экономического обоснования (ТЭО) проектирования программного обеспечения. Разработка технического задания (ТЗ) на программный проект.	Лекционные занятия	5	2	ПК-2 ПК-6
3.2	Тема 3.1 "Организация предпроектной/проектной стадии проектирования программного обеспечения" Разработка UML-модели программного обеспечения. Диаграммы Use-case (прецедентов). Декомпозиция. Пакеты. Ассоциации. Диаграммы Activity (деятельности).	Лабораторные занятия	5	2	ПК-2 ПК-6
3.3	Тема 3.1 "Организация предпроектной/проектной стадии проектирования программного обеспечения" Разработка технического задания на программный проект. Выбор	Практические занятия	5	2	ПК-2 ПК-6

	среды, обоснование языка разработки.				
3.4	Тема 3.2 "Использование CASE-технологий в проектировании программного обеспечения" Объектно-ориентированный подход к проектированию программного обеспечения. Язык моделирования UML/UML2. Диаграммы UML2.Типовое проектное решение (ТПР). Классы и структура ТПР. Моделирование информационных систем средствами Rational Software Architect.	Лекционные занятия	5	4	ПК-2 ПК-6
3.5	Тема 3.2 "Использование CASE-технологий в проектировании программного обеспечения" Разработка UML-модели программного обеспечения в Rational Software Architect. Диаграмма классов. Атрибуты, методы, свойства классов.	Лабораторные занятия	5	4	ПК-2 ПК-6
3.6	Тема 3.2 "Использование CASE-технологий в проектировании программного обеспечения". Стереотипы классов. Отношения между классами. Наследование. Разработка классов на языке C# в Visual Studio Code.	Практические занятия	5	4	ПК-2 ПК-6
3.7	Тема 3.3 "Методы прототипного проектирования программного обеспечения" Технология быстрого проектирования (RAD-технология). Содержание проектирования программного обеспечения с использованием RAD-технологии. Основные принципы методологии RAD . Экстремальное программирование.	Лекционные занятия	5	4	ПК-2 ПК-6
3.8	Тема 3.3 "Методы прототипного проектирования программного обеспечения" Разработка UML-модели программного обеспечения в Rational Software Architect. Диаграмма Sequence-последовательности действий. State-диаграмма. Диаграмма Component (компонентов). Диаграмма размещения (топологии).	Лабораторные занятия	5	4	ПК-2 ПК-6
3.9	Тема 3.3 "Методы прототипного проектирования программного обеспечения" Разработка пользовательских экранных веб-проекта форм.	Практические занятия	5	4	ПК-2 ПК-6
3.10	Разработка объектно-ориентированной модели веб-проекта. Полный комплект UML-диаграмм проектируемого программного обеспечения	Самостоятельная работа	5	20	ПК-2 ПК-6

Раздел 4. Паттерны проектирования

№	Наименование темы, краткое содержание	Вид занятия / работы / форма ПА	Семестр / Курс	Количество часов	Компетенции
4.1	Тема 4.1 "Классификация паттернов проектирования" Порождающие паттерны. Абстрактная фабрика. Строитель. Фабричный метод. Прототип. Одиночка.	Лекционные занятия	5	4	ПК-2 ПК-6
4.2	Тема 4.1 "Классификация паттернов проектирования" Разработка классов на языке C# в Visual Studio Code. Паттерны: абстрактная фабрика, строитель, фабричный метод, прототип, одиночка.	Лабораторные занятия	5	4	ПК-2 ПК-6
4.3	Тема 4.1 "Классификация паттернов проектирования" Классы. Агрегация. Композиция. Зависимость. Использование. Разработка классов на языке C# в Visual Studio Code.	Практические занятия	5	4	ПК-2 ПК-6
4.4	Тема 4.2 "Структурные паттерны. Поведенческие паттерны" Адаптер. Мост. Компоновщик. Фасад. Декоратор. Proxy. Команда. Итератор. Цепочка обязанностей. Посредник. Наблюдатель. Состояние. Шаблонный метод. Стратегия.	Лекционные занятия	5	2	ПК-2 ПК-6
4.5	Тема 4.2 "Структурные паттерны. Поведенческие паттерны" Разработка классов на языке C# в Visual Studio Code. Паттерны: адаптер, мост,компоновщик, фасад, декоратор, proxy, команда, итератор, цепочка обязанностей.	Лабораторные занятия	5	2	ПК-2 ПК-6
4.6	Тема 4.2 "Структурные паттерны. Поведенческие паттерны" Разработка классов на языке C# в Visual Studio Code. Паттерны: посредник, наблюдатель, состояние, шаблонный метод, стратегия.	Практические занятия	5	2	ПК-2 ПК-6
4.7	Курсовой проект. Перечень тем представлен в Приложении 1 к рабочей программе дисциплины.	Самостоятельная работа	5	40	ПК-2 ПК-6
4.8	Подготовка к промежуточной аттестации	Экзамен	5	36	ПК-2 ПК-6

4. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

Структура и содержание фонда оценочных средств для проведения текущего контроля и промежуточной аттестации представлены в Приложении 1 к рабочей программе дисциплины.

5. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

5.1. Учебные, научные и методические издания

	Авторы, составители	Заглавие	Издательство, год	Библиотека / Количество
1	Золотов С. Ю.	Проектирование информационных систем: учебное пособие	Томск: Эль Контент, 2013	ЭБС «Университетская библиотека онлайн»
2	Малышева Е. Н.	Проектирование информационных систем: учебное пособие	Кемерово: Кемеровский государственный университет культуры и искусств (КемГУКИ), 2009	ЭБС «Университетская библиотека онлайн»
3	Стасышин В. М.	Проектирование информационных систем и баз данных: учебное пособие	Новосибирск: Новосибирский государственный технический университет, 2012	ЭБС «Университетская библиотека онлайн»
4	Леоненков А.	Нотация и семантика языка UML: курс лекций	Москва: Национальный Открытый Университет «ИНТУИТ», 2016	ЭБС «Университетская библиотека онлайн»
5	Платёнкин А. В., Рак И. П., Терехов А. В., Чернышов В. Н.	Проектирование информационных систем. Проектный практикум: учебное пособие	Тамбов: Тамбовский государственный технический университет (ТГТУ), 2015	ЭБС «Университетская библиотека онлайн»
6		Информационные системы и технологии: журнал	Орел: Госуниверситет – УНПК, 2015	ЭБС «Университетская библиотека онлайн»
7	Хританков А. С., Полежаев В. А., Андрианов А. И.	Проектирование на UML: сборник задач: сборник задач и/или упражнений	Москва, Берлин: Директ-Медиа, 2018	ЭБС «Университетская библиотека онлайн»
8	Качановский, Ю. П., Широков, А. С.	Аппаратное и программное обеспечение персонального компьютера. Основы работы с операционной системой: методические указания к проведению лабораторной работы по курсу «информатика»	Липецк: Липецкий государственный технический университет, ЭБС АСВ, 2014	ЭБС «IPR SMART»
9		БИТ. Бизнес & Информационные технологии: журнал	Москва: Положевец и партнеры, 2019	ЭБС «Университетская библиотека онлайн»
10	Смирнов, А. А.	Прикладное программное обеспечение: учебное пособие	Москва: Евразийский открытый институт, 2011	ЭБС «IPR SMART»
11	Гунько, А. В.	Системное программное обеспечение: конспект лекций	Новосибирск: Новосибирский государственный технический университет, 2011	ЭБС «IPR SMART»

5.2. Профессиональные базы данных и информационные справочные системы

ИСС "КонсультантПлюс"
ИСС "Гарант"http://www.internet.garant.ru/
База данных научных и медицинских публикаций - ScienceDirect https://www.sciencedirect.com/

5.3. Перечень программного обеспечения

Операционная система РЕД ОС
Visual Studio Code

5.4. Учебно-методические материалы для обучающихся с ограниченными возможностями здоровья

При необходимости по заявлению обучающегося с ограниченными возможностями здоровья учебно-методические материалы предоставляются в формах, адаптированных к ограничениям здоровья и восприятия информации. Для лиц с нарушениями зрения: в форме аудиофайла; в печатной форме увеличенным шрифтом. Для лиц с нарушениями слуха: в форме электронного документа; в печатной форме. Для лиц с нарушениями опорно-двигательного аппарата: в форме электронного документа; в печатной форме.

6. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Помещения для всех видов работ, предусмотренных учебным планом, укомплектованы необходимой специализированной учебной мебелью и техническими средствами обучения:
- столы, стулья;
- персональный компьютер / ноутбук (переносной);
- проектор;
- экран / интерактивная доска.

Лабораторные занятия проводятся в компьютерных классах, рабочие места в которых оборудованы необходимыми лицензионными и/или свободно распространяемыми программными средствами и выходом в Интернет, и/или в

специализированных лабораториях, предусмотренных образовательной программой.
Помещения для самостоятельной работы обучающихся оснащены компьютерной техникой с возможностью подключения к сети "Интернет" и обеспечением доступа к электронной информационно-образовательной среде.

7. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

Методические указания по освоению дисциплины представлены в Приложении 2 к рабочей программе дисциплины.

Приложение 1

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

1 Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания

1.1 Показатели и критерии оценивания компетенций:

ЗУН, составляющие компетенцию	Показатели оценивания	Критерии оценивания	Средства оценивания
ПК-2: способен читать, понимать и выделять главную идею прочитанного исходного кода, документации			
З: основы представления программного кода, структуру и нормы составления документации по программному обеспечению	знает понятие программного продукта; правила оформления документов; возможности средств поиска информации в Интернет при подготовке к опросу, зачету и экзамену	сформировавшееся систематическое знание понятия программного продукта; правил оформления документов; возможности средств поиска информации в Интернет при ответе на вопросы опроса, зачета и экзамена	Вопросы к зачету (1-28), практико-ориентированные задания к зачету (раздел 1-2), вопросы к экзамену (1-46), практико-ориентированные задания к экзамену (раздел 3-4), лабораторные задания (4 семестр: раздел 1-2, 5 семестр: раздел 3-4), практические задания (4 семестр: раздел 1-2, 5 семестр: раздел 3-4), опрос (4 семестр: 1-28, 5 семестр: 1-46), курсовая работа (темы 1-34)
У. выявлять основные модули в программном коде, определять главную идею прочитанного кода, сформулировать результат анализа кода	применяет на практике международные и отечественные стандарты по обеспечение качества программного обеспечения при выполнении лабораторных, практических, практико-ориентированных заданий и курсового проекта	корректность применения на практике международных и отечественных стандартов по обеспечению качества программного обеспечения при выполнении лабораторных, практических, практико-ориентированных заданий и курсового проекта	Вопросы к зачету (1-28), практико-ориентированные задания к зачету (раздел 1-2), вопросы к экзамену (1-46), практико-ориентированные задания к экзамену (раздел 3-4), лабораторные задания (4 семестр: раздел 1-2, 5 семестр: раздел 3-4), практические задания (4 семестр: раздел 1-2, 5 семестр: раздел 3-4), опрос (4 семестр: 1-28, 5 семестр: 1-46), курсовая работа (темы 1-34)
В: навыками определения задачи, поставленной при разработке программного кода, основных шагов алгоритма, реализованного в программе	выполняет практические задания по обеспечение качества программного обеспечения при выполнении лабораторных, практических, практико-ориентированных заданий и курсового проекта	сформировавшееся систематическое владение навыками выполнения практических заданий по обеспечение качества программного обеспечения при выполнении лабораторных, практических, практико-ориентированных заданий и курсового проекта	Вопросы к зачету (1-28), практико-ориентированные задания к зачету (раздел 1-2), вопросы к экзамену (1-46), практико-ориентированные задания к экзамену (раздел 3-4), лабораторные задания (4 семестр: раздел 1-2, 5 семестр: раздел 3-4), практические задания (4 семестр: раздел 1-2, 5 семестр: раздел 3-4), опрос (4 семестр: 1-28, 5 семестр: 1-46), курсовая работа (темы 1-34)
ПК-6: способен моделировать инженерные процессы, использовать формальные методы конструирования и проектирования программного обеспечения			
З: основы моделирования и формальные методы конструирования программного обеспечения	знает методы формализации и моделирования программного обеспечения при подготовке к опросу, зачету и экзамену	сформировавшееся систематическое знание методов формализации и моделирования программного обеспечения при ответе на вопросы опроса, зачета и экзамена	Вопросы к зачету (1-28), практико-ориентированные задания к зачету (раздел 1-2), вопросы к экзамену (1-46), практико-ориентированные задания к экзамену (раздел 3-4), лабораторные задания (4 семестр: раздел 1-2, 5 семестр: раздел 3-4), практические задания (4 семестр: раздел 1-2, 5 семестр: раздел 3-4), опрос (4 семестр: 1-28, 5 семестр: 1-46), курсовая работа (темы 1-34)

У: использовать формальные методы конструирования программного обеспечения	использует формальные методы конструирования программного обеспечения при выполнении лабораторных, практических, практико-ориентированных заданий и курсового проекта	корректность использования формальных методов конструирования программного обеспечения при выполнении лабораторных, практических, практико-ориентированных заданий и курсового проекта	Вопросы к зачету (1-28), практико-ориентированные задания к зачету (раздел 1-2), вопросы к экзамену (1-46), практико-ориентированные задания к экзамену (раздел 3-4), лабораторные задания (4 семестр: раздел 1-2, 5 семестр: раздел 3-4), практические задания (4 семестр: раздел 1-2, 5 семестр: раздел 3-4), опрос (4 семестр: 1-28, 5 семестр: 1-46), курсовая работа (темы 1-34)
В: методами формализации и моделирования программного обеспечения	владеет методами формализации и моделирования программного обеспечения при выполнении лабораторных, практических, практико-ориентированных заданий и курсового проекта	сформировавшееся систематическое владение методами формализации и моделирования программного обеспечения при выполнении лабораторных, практических, практико-ориентированных заданий и курсового проекта	Вопросы к зачету (1-28), практико-ориентированные задания к зачету (раздел 1-2), вопросы к экзамену (1-46), практико-ориентированные задания к экзамену (раздел 3-4), лабораторные задания (4 семестр: раздел 1-2, 5 семестр: раздел 3-4), практические задания (4 семестр: раздел 1-2, 5 семестр: раздел 3-4), опрос (4 семестр: 1-28, 5 семестр: 1-46), курсовая работа (темы 1-34)

1.2 Шкалы оценивания:
Текущий контроль успеваемости и промежуточная аттестация осуществляется в рамках накопительной балльно-рейтинговой системы в 100-балльной шкале.
- 84-100 баллов (оценка «отлично»);
- 67-83 баллов (оценка «хорошо»);
- 50-66 баллов (оценка удовлетворительно);
- 0-49 баллов (оценка неудовлетворительно).

- 50-100 баллов (зачтено);
- 0-49 баллов (не зачтено).

2 Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы

4 сем.

Вопросы к зачету

1. Понятие программного обеспечения (ПО), программного продукта, программного проекта.
2. Жизненный цикл ПО.
3. Этапы создания ПО.
4. Модели жизненного цикла ПО.
5. Требования к эффективности и надежности проектных решений.
6. Критерии качества программного проекта.
7. Стандарты качества. Обзор современных стандартов и технологий создания программного обеспечения.
8. Организация канонического проектирования. Стадии проекта.
9. Разработка модели деятельности организации ("как есть" и "как должно быть"). Декомпозиция функций.
10. Классификация информации. Понятия и основные требования к системе. Состав и содержание операций проектирования классификаторов.
11. Внутримашинное информационное обеспечение. Процессы проектирования первичных (входных) и результатных (выходных) документов.
12. Схема Захмана архитектуры программной системы. Иерархический принцип определения архитектуры.
13. Классификация технологий, методов и средств проектирования.
14. Выбор технологии проектирования.
15. Модульность.
16. Компонентная технология. Методы создания и использования компонентов. Взаимодействие компонентов.
17. Распределенные системы. Принцип открытой архитектуры (SOA).
18. Драйверы ODBC- стандарт открытого взаимодействия баз данных. Стандарты COM, ДСОМ, CORBA и др. Интеграция моделей.

19. IDEF0-диаграмма проекта. Декомпозиция работ модели. Потоки данных (входящие, выходящие, управление, механизм). Тоннельность.
20. DFD-диаграмма проекта. Документооборот. Внешние сущности. Хранилища данных.
21. IDEF3-диаграмма проекта. Логика функций. Справочники.
22. Анализ организационной структуры предприятия. Char-диаграмма.
23. Node-Tree-диаграмма. Разработка ролей системы.
24. Импорт данных из структурной модели в ER-модель.
25. Разработка логической модели данных.
26. Разработка физической модели данных.
27. Генерация скриптов.
28. ABC-анализ затрат на реализацию функций программного обеспечения.

Практико-ориентированные задания к зачету

Раздел 1 «Методологические аспекты проектирования программного обеспечения».

Задание 1. Используя любой графический редактор, постройте схему взаимосвязей следующих терминов: «бизнес-процесс», «процесс», «результат процесса», «правила выполнения действий процесса», «вход процесса», «выход процесса», «участники процесса», «ресурсы», «материалы», «продукты», «данные», «основные процессы», «обеспечивающие процессы», «процессы управления», «процессы развития», «процесс верхнего уровня», «подпроцесс», «процедура», «функция (операция)», «транзакция», «модель». Для описания взаимосвязей используйте следующий набор отношений: *содержит, может быть, имеет отношение с, является, классифицируется*), *является образцом, имеет, является частью, синоним (предпочтительный термин), является родовым термином, является особенностью.*

Задание 2. Постройте модель верхнего уровня процессов, направленных на достижение данной цели. Выделите на ней основные и обеспечивающие процессы.

Задание 3. Используя визуально графический метод, изобразите процесс из повседневной жизни. Выделите проблемные участки, предложите пути решения.

Задание 4. Распишите процесс выполнения «домашнего задания» за компьютером, выявите проблемные участки, предложите пути улучшения, улучшите уже существующие процессы.

Задание 5. Разработайте модель процесса моделирования для организации.

Раздел 2. «Современные принципы проектирования архитектуры программного обеспечения».

Задание 1. Рассмотрим организацию работы call-центра. Операторы медленно отвечают на звонки, с трудом ищут информацию в базе данных, грубо говорят с клиентами. Из предложений по оптимизации поступили следующие, выбрать можно одно из них: - решение об усовершенствовании базы данных, но забыть об улучшении АТС - принятии этического регламента для сотрудников, об определении точных критериев качества (время ответа на звонок, время консультации, опрос абонентов об удовлетворенности услугой. Обоснуйте свой выбор.

Задание 2 Спроектировать базу данных программного обеспечения.

Задание 3. Разработать проект программного обеспечения.

Задание 4. Спроектировать интерфейс программного обеспечения.

Задание 5 Написать код для реализации определенной функциональности в программном обеспечении.

Критерии оценивания:

- 50-100 баллов («зачтено») – изложенный материал фактически верен, наличие глубоких исчерпывающих знаний в объеме пройденной программы дисциплины в соответствии с поставленной программой курса целью обучения; правильные, уверенные действия по применению полученных навыков и умений при решении практико-ориентированных заданий, грамотное и логически стройное изложение материала при ответе, усвоение основной и знакомство с дополнительной литературой;
- 0-49 баллов («не зачтено») – ответы не связаны с вопросами, наличие грубых ошибок в ответе, непонимание сущности излагаемого вопроса, неумение применять умения и навыки при решении практико-ориентированных заданий, неуверенность и неточность ответов на дополнительные и наводящие вопросы.

5 сем.

Вопросы к экзамену

1. Состав работ на предпроектной стадии при проектировании ПО.
2. Стадии технического и рабочего проектирования.
3. Разработка технического задания (ТЗ) на программный проект.
4. Объектно-ориентированный подход к проектированию программного обеспечения.
5. Язык моделирования UML/UML2.
6. Диаграммы UML/UML2.
7. Типовое проектное решение (ТПР).
8. Классы и структура ТПР.
9. Атрибуты, методы, свойства классов.
10. Стереотипы классов.

11. Отношения между классами. Наследование.
12. Отношения между классами. Агрегация. Композиция. Зависимость.
13. Разработка классов на языке C# в Visual Studio Code
14. Проектирование программного обеспечения средствами Rational Software Architect.
15. Диаграммы прецедентов (Use Case diagram)
16. Диаграммы деятельности (Activity Diagram)
17. Диаграмма классов (Class diagram)
18. Диаграммы состояний (Statechart diagram)
19. Диаграммы последовательности действий (Sequence diagram)
20. Диаграммы взаимодействий (Collaboration diagram)
21. Диаграммы компонентов (Component diagram)
22. Диаграммы топологии (Deployment diagram)
23. Технология быстрого проектирования (RAD-технология).
24. Экстремальное программирование.
25. Разработка пользовательских экранных форм.
26. Классификация паттернов проектирования.
27. Абстрактная фабрика.
28. Строитель.
29. Фабричный метод.
30. Прототип.
31. Одиночка.
32. Абстрактная фабрика.
33. Адаптер.
34. Мост.
35. Компоновщик.
36. Фасад.
37. Декоратор.
38. Proxy.
39. Команда.
40. Итератор.
41. Цепочка обязанностей.
42. Посредник.
43. Наблюдатель.
44. Состояние.
45. Шаблонный метод.
46. Стратегия.

Практико-ориентированные задания к экзамену

Раздел 3 «Современные технологии проектирования программного обеспечения».

Задание 1 Расчет показателей качества программного обеспечения

Задание 2 Работа в Case-средстве Data Modeler (ER Win). Разработка логической модели базы данных. Разработка физической модели.

Задание 3 ABC-анализ затрат на реализацию функций программного обеспечения.

Задание 4. Построить семантическую модель структуры данных информационной системы, используя диаграммы сущность-связь (ER - Entity-Relationship).

Задание 5. Построить диаграммы сущность-связь (ER - Entity-Relationship), используя нотацию Баркера.

Раздел 4 «Паттерны проектирования».

Задание 1. Реализовать две иерархии классов, в одну из которых входят абстрактный создатель Creator и два конкретных создателя ConcreteCreator1 и ConcreteCreator2, а в другую — абстрактный продукт Product и два конкретных продукта ConcreteProduct1 и ConcreteProduct2.

Задание 2 Реализовать иерархию классов-животных с абстрактным предком Animal, содержащим метод GetInfo, который возвращает строковое значение, и шестью конкретными потомками: Lion, Tiger, Leopard (кошачьи, cats), Gorilla, Orangutan, Chimpanzee (человекообразные обезьяны, apes). Каждый конкретный класс содержит строковое поле name (имя животного), которое определяется в его конструкторе с помощью одноименного параметра. Метод GetInfo возвращает имя класса и имя животного, разделенные пробелом, например, Lion Tom

Задание 3 Спроектировать и реализовать иерархию классов для игровых персонажей и разных типов вооружения. Каждый персонаж в любой момент времени использует только один вид оружия, но может свободно менять оружие в ходе игры.

Задание 4. Промоделировать чат на основе паттерна Observer.

Задание 5 Реализовать класс «Длинное целое». Обеспечить возможность выполнения арифметических операций с экземплярами класса.

Критерии оценивания:

- 84-100 баллов (оценка «отлично») - изложенный материал фактически верен, наличие глубоких исчерпывающих знаний в объеме пройденной программы дисциплины в соответствии с поставленными программой курса целями и задачами обучения; правильные, уверенные действия по применению полученных знаний на практике, грамотное и логически стройное изложение материала при ответе, усвоение основной и знакомство с дополнительной литературой;
- 67-83 баллов (оценка «хорошо») - наличие твердых и достаточно полных знаний в объеме пройденной программы дисциплины в соответствии с целями обучения, правильные действия по применению знаний на практике, четкое изложение материала, допускаются отдельные логические и стилистические погрешности, обучающийся усвоил основную литературу, рекомендованную в рабочей программе дисциплины;
- 50-66 баллов (оценка удовлетворительно) - наличие твердых знаний в объеме пройденного курса в соответствии с целями обучения, изложение ответов с отдельными ошибками, уверенно исправленными после дополнительных вопросов; правильные в целом действия по применению знаний на практике;
- 0-49 баллов (оценка неудовлетворительно) - ответы не связаны с вопросами, наличие грубых ошибок в ответе, непонимание сущности излагаемого вопроса, неумение применять знания на практике, неуверенность и неточность ответов на дополнительные и наводящие вопросы.

Лабораторные задания

1.Тематика лабораторных работ по разделам и темам

4 сем.

Раздел 1 «Методологические аспекты проектирования программного обеспечения».

Тема 1.1 " Структура программного проекта ".

Лабораторное задание 1 Разработка схемы архитектуры программного проекта.

Тема 1.2 «Качество программного проекта».

Лабораторное задание 2 Расчет показателей качества программного обеспечения

Тема 1.3 «Каноническое проектирование».

Лабораторное задание 3 Разработка IDEF0 диаграмм проекта (модель "как-должно быть"). Декомпозиция работ модели. Потоки данных (входящие, выходящие, управление, механизм). Тоннельность.

Раздел 2 «Современные принципы проектирования архитектуры программного обеспечения»

Тема 2.1. «Архитектура программного обеспечения».

Лабораторное задание 4 Разработка DFD диаграмм проекта (модель "как-должно быть"). Документооборот. Внешние сущности. Хранилища данных. Разработка IDEF3 диаграмм проекта (модель "как-должно быть"). Логика функций. Справочники.

Тема 2.2 «Методы и средства проектирования программного обеспечения».

Лабораторное задание 5 Анализ организационной структуры предприятия. Разработка Char-диаграммы, Node-Tree-диаграммы. Разработка ролей системы.

5 сем.

Раздел 3 «Современные технологии проектирования программного обеспечения».

Тема 3.1 «Организация предпроектной/проектной стадии проектирования программного обеспечения».

Лабораторное задание 1 Разработка UML-модели программного обеспечения. Диаграммы Use-case (прецедентов). Декомпозиция. Пакеты. Ассоциации. Диаграммы Activity (деятельности).

Тема 3.2 "Использование CASE-технологий в проектировании программного обеспечения".

Лабораторное задание 2 Разработка UML-модели программного обеспечения в Rational Software Architect. Диаграмма классов. Атрибуты, методы, свойства классов..

Тема 3.3 "Методы прототипного проектирования программного обеспечения"

Лабораторное задание 3 Разработка UML-модели программного обеспечения в Rational Software Architect. Диаграмма Sequence-последовательности действий. State-диаграмма. Диаграмма Component (компонентов). Диаграмма размещения (топологии).

Раздел 4 «Паттерны проектирования».

Тема 4.1 "Классификация паттернов проектирования".

Лабораторное задание 4 Разработка классов на языке C# в Visual Studio Code. Паттерны: абстрактная фабрика, строитель, фабричный метод, прототип, одиночка.

Тема 4.2 "Структурные паттерны. Поведенческие паттерны".

Лабораторное задание 5 Разработка классов на языке C# в Visual Studio Code. Паттерны: адаптер, мост, компоновщик, фасад, декоратор, proxy, команда, итератор, цепочка обязанностей..

2. Критерии оценивания:

За каждый сем.

Максимальное количество баллов: 40 баллов.

Каждое задание оценивается максимум в 8 баллов.

8 б. – задание выполнено верно;

7-6 б. – при выполнении задания были допущены неточности, не влияющие на результат;

5-3 б. – при выполнении задания были допущены ошибки;

2-1 б. – при выполнении задания были допущены существенные ошибки.

0 б. – задание не выполнено.

Практические задания

1.Тематика практических работ по разделам и темам

4 сем.

Раздел 1 «Методологические аспекты проектирования программного обеспечения».

Тема 1.1 " Структура программного проекта ".

Практическое задание 1 Анализ технологий проектирования. Анализ архитектуры коммерческих программных продуктов.

Тема 1.2 «Качество программного проекта».

Практическое задание 2 Сравнительный анализ отечественных стандартов качества программного обеспечения

Тема 1.3 «Каноническое проектирование».

Практическое задание 3 Классификация информации. Системы кодификации. Разработка системы кодирования на предприятии.

Раздел 2 «Современные принципы проектирования архитектуры программного обеспечения»

Тема 2.1. «Архитектура программного обеспечения».

Практическое задание 4 Разработка входящих/исходящих документов (шаблонов) для программного проекта

Тема 2.2 «Методы и средства проектирования программного обеспечения».

Практическое задание 5 Импорт данных из структурной модели в ER-модель. Разработка логической модели данных. Разработка физической модели данных. Генерация скриптов.

5 сем.

Раздел 3 «Современные технологии проектирования программного обеспечения».

Тема 3.1 «Организация предпроектной/проектной стадии проектирования программного обеспечения».

Практическое задание 1 Разработка технического задания на программный проект. Выбор среды, обоснование языка разработки.

Тема 3.2 "Использование CASE-технологий в проектировании программного обеспечения".

Практическое задание 2 Стереотипы классов. Отношения между классами. Наследование. Разработка классов на языке C# в Visual Studio Code.

Тема 3.3 "Методы прототипного проектирования программного обеспечения"

Практическое задание 3 Разработка пользовательских экранных веб-проекта форм.

Раздел 4 «Паттерны проектирования».

Тема 4.1 "Классификация паттернов проектирования".

Практическое задание **4** Классы. Агрегация. Композиция. Зависимость. Использование. Разработка классов на языке C# в Visual Studio Code.

Тема 4.2 "Структурные паттерны. Поведенческие паттерны".

Практическое задание 5 Разработка классов на языке C# в Visual Studio Code. Паттерны: посредник, наблюдатель, состояние, шаблонный метод, стратегия.

2. Критерии оценивания:

За каждый сем.

Максимальное количество баллов: 40 баллов.

Каждое задание оценивается максимум в 8 баллов.

8 б. – задание выполнено верно;

7-6 б. – при выполнении задания были допущены неточности, не влияющие на результат;

5-3 б. – при выполнении задания были допущены ошибки;

2-1 б. – при выполнении задания были допущены существенные ошибки.

0 б. – задание не выполнено.

Вопросы для опроса

4 сем.

1. Понятие программного обеспечения (ПО), программного продукта, программного проекта.
2. Жизненный цикл ПО.
3. Этапы создания ПО.
4. Модели жизненного цикла ПО.
5. Требования к эффективности и надежности проектных решений.
6. Критерии качества программного проекта.
7. Стандарты качества. Обзор современных стандартов и технологий создания программного обеспечения.
8. Организация канонического проектирования. Стадии проекта.
9. Разработка модели деятельности организации ("как есть" и "как должно быть"). Декомпозиция функций.
10. Классификация информации. Понятия и основные требования к системе. Состав и содержание операций проектирования классификаторов.

11. Внутримашинное информационное обеспечение. Процессы проектирования первичных (входных) и результатных (выходных) документов.
12. Схема Захмана архитектуры программной системы. Иерархический принцип определения архитектуры.
13. Классификация технологий, методов и средств проектирования.
14. Выбор технологии проектирования.
15. Модульность.
16. Компонентная технология. Методы создания и использования компонентов. Взаимодействие компонентов.
17. Распределенные системы. Принцип открытой архитектуры (SOA).
18. Драйверы ODBC- стандарт открытого взаимодействия баз данных. Стандарты COM, ДСОМ, CORBA и др. Интеграция моделей.
19. IDEF0-диаграмма проекта. Декомпозиция работ модели. Потоки данных (входящие, выходящие, управление, механизм). Тоннельность.
20. DFD-диаграмма проекта. Документооборот. Внешние сущности. Хранилища данных.
21. IDEF3-диаграмма проекта. Логика функций. Справочники.
22. Анализ организационной структуры предприятия. Char-диаграмма.
23. Node-Tree-диаграмма. Разработка ролей системы.
24. Импорт данных из структурной модели в ER-модель.
25. Разработка логической модели данных.
26. Разработка физической модели данных.
27. Генерация скриптов.
28. ABC-анализ затрат на реализацию функций программного обеспечения.

5 сем.

1. Состав работ на предпроектной стадии при проектировании ПО.
2. Стадии технического и рабочего проектирования.
3. Разработка технического задания (ТЗ) на программный проект.
4. Объектно-ориентированный подход к проектированию программного обеспечения.
5. Язык моделирования UML/UML2.
6. Диаграммы UML/UML2.
7. Типовое проектное решение (ТПР).
8. Классы и структура ТПР.
9. Атрибуты, методы, свойства классов.
10. Стереотипы классов.
11. Отношения между классами. Наследование.
12. Отношения между классами. Агрегация. Композиция. Зависимость.
13. Разработка классов на языке C# в Visual Studio Code
14. Проектирование программного обеспечения средствами Rational Software Architect.
15. Диаграммы прецедентов (Use Case diagram)
16. Диаграммы деятельности (Activity Diagram)
17. Диаграмма классов (Class diagram)
18. Диаграммы состояний (Statechart diagram)
19. Диаграммы последовательности действий (Sequence diagram)
20. Диаграммы взаимодействий (Collaboration diagram)
21. Диаграммы компонентов (Component diagram)
22. Диаграммы топологии (Deployment diagram)
23. Технология быстрого проектирования (RAD-технология).
24. Экстремальное программирование.
25. Разработка пользовательских экранных форм.
26. Классификация паттернов проектирования.
27. Абстрактная фабрика.
28. Строитель.
29. Фабричный метод.
30. Прототип.
31. Одиночка.
32. Абстрактная фабрика.
33. Адаптер.
34. Мост.
35. Компоновщик.
36. Фасад.
37. Декоратор.
38. Proxy.

39. Команда.
40. Итератор.
41. Цепочка обязанностей.
42. Посредник.
43. Наблюдатель.
44. Состояние.
45. Шаблонный метод.
46. Стратегия.

Критерии оценивания:

За каждый сем.

Максимальное количество баллов: 20 баллов.

Во время опроса обучаемому задаются 5 вопросов.

За один ответ обучаемый получает:

4 б. – за правильный ответ;

3 б. – при ответе были допущены неточности, не влияющие на результат;

2 б. – при ответе были допущены ошибки;

1 б. – при ответе были допущены существенные ошибки.

0 б. – не ответил на вопрос.

Темы курсовых проектов

1. Проектирование программного проекта для ведения домашней бухгалтерии
2. Проектирование программного проекта для автоматизации функций менеджера туристической фирмы
3. Проектирование программного проекта по учету и оплате телефонных разговоров
4. Проектирование программного проекта для заказа такси
5. Проектирование программного проекта бронирования авиабилетов
6. Проектирование программного проекта бронирования мест в гостинице
7. Проектирование программного проекта расчета заработной платы
8. Проектирование программного проекта по оплате коммунальных услуг
9. Проектирование программного проекта «Абитуриент»
10. Проектирование программного проекта для автоматизации начисления стипендий в деканате
11. Проектирование программного проекта «Деканат»
12. Проектирование программного проекта продажи авиабилетов
13. Проектирование программного проекта продажи железнодорожных билетов
14. Проектирование программного проекта «Скорая помощь»
15. Проектирование программного проекта продажи билетов в кинотеатре
16. Проектирование программного модуля заказа товаров
17. Проектирование программного модуля «Регистратура в поликлинике»
18. Проектирование программного проекта «Аптека»
19. Проектирование программного проекта для автоматизации страховой компании
20. Проектирование программного проекта для автоматизации риэлтерской компании
21. Проектирование «умного дома».
22. Разработка компонент «умного дома».
23. Разработка компонент нейронной сети для решения задачи «*».
24. Разработка компонент нечеткой логики для решения задачи «*».
25. Разработка проекта dll-библиотеки использования «генетических» алгоритмов.
26. Проектирование и разработка клиентского модуля для решения задачи «*» на основе Google Map API (Yandex Map API, OpenStreetMap API).
27. Проектирование desktop-приложения «*».
28. Проектирование web-системы (сервиса, портала) «*».
29. Проектирование web CRM-системы (сервиса, портала) «*».
30. Проектирование мобильного приложения «*».
31. Проектирование кроссплатформенного приложения «*».
32. Проектирование игровых алгоритмов «*» на движке Unity.
33. Проектирование систем автоматики с использованием Arduiono (Raspberry Pi).
34. Проектирование робототехники с использованием Arduiono (Raspberry Pi).

Критерии оценивания:

- 84-100 баллов (оценка «отлично») – курсовой проект выполнен верно, наличие глубоких исчерпывающих знаний в объеме пройденной программы дисциплины; правильные, уверенные действия по применению полученных навыков и умений при решении практико-ориентированных заданий, грамотное и логически стройное изложение материала при ответе, усвоение основной и знакомство с дополнительной литературой;

- 67-83 баллов (оценка «хорошо») - изложенный материал в курсовом проекте фактически верен, допускаются отдельные логические и стилистические погрешности; усвоена основная литература, рекомендованная в рабочей программе дисциплины;
- 50-66 баллов (оценка «удовлетворительно») - изложенный материал фактически верен, изложение материала с отдельными ошибками, уверенно исправленными после дополнительных вопросов; выполняются в целом корректные действия по применению знаний на практике;
- 0-49 баллов (оценка «неудовлетворительно») – курсовой проект выполнен с допущением грубых ошибок или не выполнен вообще, ответы не связаны с вопросами, наличие грубых ошибок в ответе, непонимание сущности излагаемого материала, неумение применять умения и навыки при решении практико-ориентированных заданий, неуверенность и неточность ответов на дополнительные и наводящие вопросы.

3 Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций

Процедуры оценивания включают в себя текущий контроль и промежуточную аттестацию.

Текущий контроль успеваемости проводится с использованием оценочных средств, представленных в п. 2 данного приложения. Результаты текущего контроля доводятся до сведения студентов до промежуточной аттестации.

Промежуточная аттестация проводится в форме зачета, экзамена, защиты курсового проекта.

Зачет проводится по расписанию промежуточной аттестации. Количество вопросов в задании – 3: два теоретических вопроса и одно практико-ориентированное задание. Объявление результатов производится в день зачета. Результаты аттестации заносятся в ведомость и зачетную книжку студента. Студенты, не прошедшие промежуточную аттестацию по графику, должны ликвидировать задолженность в установленном порядке.

Экзамен проводится по расписанию промежуточной аттестации в письменном виде. Количество вопросов в экзаменационном задании – 3: два теоретических вопроса и одно практико-ориентированное задание. Проверка ответов и объявление результатов производится в день экзамена. Результаты аттестации заносятся в ведомость и зачетную книжку студента. Студенты, не прошедшие промежуточную аттестацию по графику, должны ликвидировать задолженность в установленном порядке.

Защита курсового проекта проводится по расписанию промежуточной аттестации. Результаты защиты курсового проекта заносятся в ведомость и зачетную книжку студента.

Приложение 2

МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

Учебным планом предусмотрены следующие виды занятий:
- лекции;
- лабораторные занятия;
- практические занятия.

В ходе лекционных занятий рассматриваются основные теоретические вопросы, даются рекомендации для самостоятельной работы и подготовке к лабораторным и практическим занятиям.

В ходе практических занятий углубляются и закрепляются знания студентов по ряду рассмотренных на лекциях вопросов, развиваются навыки практической работы.

При подготовке к лабораторным и практическим занятиям каждый студент должен:
– изучить рекомендованную учебную литературу;
– изучить конспекты лекций;
– подготовить ответы на все вопросы по изучаемой теме.

В процессе подготовки к лабораторным занятиям студенты могут воспользоваться консультациями преподавателя.

Вопросы, не рассмотренные на лабораторных занятиях, должны быть изучены студентами в ходе самостоятельной работы. Контроль самостоятельной работы студентов над учебной программой курса осуществляется в ходе занятий методом опроса. В ходе самостоятельной работы каждый студент обязан прочитать основную и по возможности дополнительную литературу по изучаемой теме, дополнить конспекты недостающим материалом, выписками из рекомендованных первоисточников. Выделить непонятные термины, найти их значение в энциклопедических словарях.

Студент должен готовиться к предстоящему лабораторному занятию по всем, обозначенным в рабочей программе дисциплины вопросам.

Для подготовки к занятиям, текущему контролю и промежуточной аттестации студенты могут воспользоваться электронно-библиотечными системами. Также обучающиеся могут взять на дом необходимую литературу на абонементе университетской библиотеки или воспользоваться читальными залами.

Курсовой проект имеет следующую структуру:
а) титульный лист;
б) содержание;
в) введение;
г) основная часть;
д) заключение;
е) список использованных источников
ж) приложение.

Все части курсового проекта должны быть логично связанными между собой, без резких переходов из одной в другую.

Курсовой проект выполняется на листах формата А4. Текст печатается на одной стороне листа. Объем курсового проекта – 25 - 30 страниц (1,5 интервал). При использовании таблиц, схем и рисунков допускаются незначительные отклонения от нормы. Все графики и рисунки сопровождаются номером, названием и ссылкой на источник. Параметры абзаца: выравнивание текста по ширине – страницы; отступ первой строки – 1,25 мм.; межстрочный интервал – полуторный. Поля: верхнее – 2,5 см.; нижнее – 2 см.; левое – 3 см.; правое – 1 см. Нумерация страниц начинается с третьей станицы (титульный лист и содержание (оглавление) не нумеруются). На титульном листе указывается название вуза; тема курсового проекта; курс обучения, группа, ФИО автора; ФИО, учёное звание, степень преподавателя; город и год. Список литературы оформляется в алфавитном порядке в соответствии с ГОСТом.