Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Костровец Лариса Борисовна
Должность: директор
Дата подписания: 02.10.2026 00:12:43
Уникальный программный ключ:
6882606104c36dbde41c4ab93a65382136a292d6

Приложение 4
к образовательной программе

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Б1.О.02.12 Основы программной инженерии
(индекс, наименование дисциплины в соответствии с учебным планом)

09.03.03 Прикладная информатика
(код, наименование направления подготовки/специальности)

Прикладная информатика в управлении корпоративными информационными системами
(наименование образовательной программы)

Очная форма обучения
(форма обучения)

Год набора – 2026
Донецк

Автор(ы)-составитель(и) РПД:

Верзилов Александр Николаевич, канд. физ.-мат. наук, доцент, доцент кафедры информационных технологий

Заведующий кафедрой:

Брадул Наталья Валерьевна, канд. физ.-мат. наук, заведующий кафедрой информационных технологий

Рабочая программа дисциплины Б1.О.02.12 Основы программной инженерии одобрена на заседании кафедры информационных технологий факультета государственной службы и управления Донецкого филиала РАНХиГС.

Протокол № 7 от «05 » марта 2026 г.

СОДЕРЖАНИЕ

1. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы

2. Объем и место дисциплины в структуре образовательной программы

3. Содержание и структура дисциплины

4. Типы оценочных материалов, показатели и критерии их оценивания

5. Формы аттестации, типовые оценочные материалы для текущего контроля успеваемости обучающихся, критерии и шкалы оценивания по контрольным точкам

6. Формы промежуточной аттестации, критерии и шкала оценивания, типовые оценочные материалы по дисциплине

7. Методические материалы по освоению дисциплины

8. Учебная литература и ресурсы информационно-телекоммуникационной сети «Интернет»

9. Материально-техническая база, информационные технологии, программное обеспечение и информационные справочные системы

1. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы

Дисциплина Б1.О.02.12 Основы программной инженерии обеспечивает формирование у обучающихся следующих общепрофессиональных компетенций*:

ОТФ/ТФ и реквизиты ПС *(при наличии)***	**Код компетенции ****	**Наименование компетенции****	**Код индикатора достижения компетенций****	**Наименование индикатора достижения компетенций ****	**Образовательный результат ****
-	ОПК-2	Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности	ОПК-2.2	Использует принципы работы современных информационных систем и методы проектирования программных средств, в том числе отечественного производства в своей профессиональной деятельности	**Знает** технологию разработки программного обеспечения и методы выявления требований на программный продукт, основные этапы и содержание работ на каждом этапе разработки прикладных программных систем. **Умеет** использовать современные технологии программирования, тестирования и документирования программных комплексов

** Дисциплина может формировать компетенцию полностью или частично.*
***Должно соответствовать Приложению 1 к образовательной программе*

2. Объем и место дисциплины в структуре образовательной программы

Общий объем дисциплины:

3,00 з.е., 108 ак.час

Контактная работа обучающихся с преподавателем по видам учебных занятий: 44 ак. час на контактную работу с преподавателем, из них 18 ак. час на лекции и 36 ак. час на практические занятия. 45 ак. час на самостоятельную работу обучающихся.

Б1.О.02.12 Основы программной инженерии реализуется на 5-м семестре 3-го курса после изучения дисциплин:

- Информационные системы и технологии;
- Проектирование информационных систем;
- Архитектура вычислительных систем.

.

.

3. Содержание и структура дисциплины

3.1. Структура дисциплины

Очная форма обучения

№ п/п	Наименование тем и (или) разделов	ВСЕГО	Объем дисциплины, ак.час												Форма текущего контроля успеваемости, промежуточной аттестации
			Контактная работа обучающихся с преподавателем по видам учебных занятий									Самостоятельная работа			
			Период теоретического обучения						Период промежуточной аттестации (сессия)						
			Занятия лекционного типа		Занятия семинарского типа		ИК	КСР	КЭ	Каттэк	Контроль	СРкр	СРэк	СР	
			Л	ВЛ	ЛР	ПЗ									
РАЗДЕЛ 1. Введение и жизненный цикл ПО															
Тема 1	Введение в предмет. Основные понятия.	11	2	0	0	4	0	0	0	0	0	0	0	5	Устный опрос, контрольные задания, КТ№1
Тема 2	Классические модели ЖЦ	22	4	0	0	8	0	0	0	0	0	0	0	10	Устный опрос, контрольные задания, КТ№1
РАЗДЕЛ 2. Работа с требованиями и проектирование															
Тема 3	Сбор и анализ требований. Спецификация	11	2	0	0	4	0	0	0	0	0	0	0	5	Устный опрос, контрольное задание, КТ№2

Тема 4	Основы архитектур-ного проекти-рования	22	4	0	0	8	0	0	0	0	0	0	0	10	Устный опрос, контрольные задания, КТ№2
РАЗДЕЛ 3. Реализация, тестирование и качество															
Тема 5	Стандарты кодирования. Виды тестирования	22	4	0	0	8	0	0	0	0	0	0	0	10	Устный опрос, контрольное задание, КТ№3
Тема 6	Отладка и логирование. Качество ПО	11	2	0	0	4	0	0	0	0	0	0	0	5	Устный опрос, контрольное задание, КТ№3
Промежуточная аттестация		9	0	0	0	0	0	0	0	9	0	0	0	0	Зачет
Итого		**108**	**18**	**0**	**0**	**36**	**0**	**0**	**0**	**9**	**0**	**0**	**0**	**45**	

Используемые сокращения:

Л – лекции - занятия, предусматривающие преимущественную передачу учебной информации обучающимся педагогическими работниками организации и (или) лицами, привлекаемыми организацией к реализации образовательных программ на иных условиях,).

ВЛ – видео лекции.

ЛР – лабораторные работы.

ПЗ – практические занятия (за исключением лабораторных работ).

ИК – индивидуальные консультации.

КСР – контроль самостоятельной работы

КЭ – консультации перед экзаменом

Каттэк – контактная работа на аттестацию в период экзаменационных сессий

Контроль - контактная работа на аттестацию в период экзаменационных сессий для заочной формы обучения

СРкр – самостоятельная работа на подготовку курсовой работы/ курсового проекта.

СРэк – самостоятельная работа на подготовку к экзамену.

СР – самостоятельная работа в семестре на подготовку к учебным занятиям.

3.2. Содержание дисциплины

РАЗДЕЛ 1. ВВЕДЕНИЕ И ЖИЗНЕННЫЙ ЦИКЛ ПО

Тема 1. Введение в предмет. Основные понятия. ОПК-2.2

Определение программной инженерии, отличие от программирования. Отличие программной инженерии от системотехники.

Предпосылки и история возникновения программной инженерии.

Характеристики ПО. Процесс создания ПО. Методологии разработки ПО.

Понятие жизненного цикла ПО (ЖЦ).

Тема 2. Классические модели ЖЦ. ОПК-2.2

Классические модели ЖЦ: каскадная, V-модель, итеративные (инкрементная, спиральная).

Современные подходы: Agile (обзор Scrum, Kanban без глубокого управления).

Роли в команде (обзорно).

РАЗДЕЛ 2. РАБОТА С ТРЕБОВАНИЯМИ И ПРОЕКТИРОВАНИЕ

Тема 3. Сбор и анализ требований. Спецификация. ОПК-2.2

Сбор и анализ требований: функциональные / нефункциональные.

Анализ предметной области. Методы выделения требований. Методы описания и систематизации требований. Использование различных видов графических диаграмм при описании требований.

Анализ требований к программному обеспечению. Основные понятия методов формальной спецификации.

Тема 4. Основы архитектурного проектирования. ОПК-2.2

Основы архитектурного проектирования: компоненты, модули, интерфейсы. Нисходящее и восходящее проектирование.

UML: диаграммы вариантов использования, классов, последовательности (основы).

Параллельное программирование. Структурное проектирование. Объектно-ориентированный анализ и проектирование. Компонентно-базированная разработка.

Разработка ПО для повторного использования.

РАЗДЕЛ 3. РЕАЛИЗАЦИЯ, ТЕСТИРОВАНИЕ И КАЧЕСТВО

Тема 5. Стандарты кодирования. Виды тестирования. ОПК-2.2

Стандарты кодирования, ревью кода. Виды тестирования: модульное, интеграционное, системное, приемочное. Основы тест-дизайна (эквивалентность, граничные значения).

Планирование аттестационного тестирования. Основные методы построения тестов. Тестирование по методу «черного ящика» и методу

«белого ящика».

Тестирование модулей, интеграция модулей и проверка правильности интеграции, тестирование системы. Объектно-ориентированное тестирование. Инспектирование.

Тема 6. Отладка и логирование. Качество ПО. ОПК-2.2

Понятие отладки и логирования.

Понятие качества программного обеспечения и его основные характеристики. Методы контроля качества программного обеспечения.

Различные техники экспертизы.

Проверка моделей. Дедуктивная верификация.

4. Типы оценочных материалов, показатели и критерии оценивания

4.1. Оценочные материалы по дисциплине Б1.О.02.12 Основы программной инженерии входят в состав оценочных материалов по образовательной программе. Совокупность оценочных материалов по всем дисциплинам (модулям) образовательной программы составляет фонд оценочных средств (далее – ФОС). ФОС используется при проведении текущего контроля успеваемости и промежуточной аттестации обучающихся с целью оценивания достижения обучающимися планируемых результатов обучения.

4.2. ФОС разработан как комплекс проверочных заданий различного типа и уровня сложности, включает критерии и шкалы оценивания, а также «ключи» правильных ответов. ФОС формируется как отдельный документ и хранится в электронном виде, доступ к ФОС предоставлен ограниченному кругу лиц.

4.3. Для самостоятельной работы обучающихся при подготовке к текущему контролю успеваемости и промежуточной аттестации в рабочих программах дисциплин размещены типовые проверочные задания, которые можно условно разделить на задания закрытого, комбинированного и открытого типов.

Задания закрытого типа – это тестовые задания, в которых каждый вопрос сопровождается готовыми вариантами ответов, из которых необходимо выбрать один или несколько правильных.

Задания комбинированного типа – это тестовые задания, в которых каждый вопрос сопровождается готовыми вариантами ответов, из которых необходимо выбрать один или несколько правильных и обосновать свой

выбор.

Задания открытого типа – это задания, в которых на каждый вопрос должен быть предложен развернутый обоснованный ответ.

В зависимости от типа задания рекомендованы определенная последовательность выполнения и система оценивания выполнения заданий.

4.4. Типы заданий, сценарии выполнения, критерии оценивания

ТИП ЗАДАНИЯ	ИНСТРУКЦИЯ	СЦЕНАРИИ ВЫПОЛНЕНИЯ	КРИТЕРИИ ОЦЕНИВАНИЯ
Задание закрытого типа с выбором одного правильного ответа из нескольких вариантов предложенных	Прочитайте текст, выберите правильный ответ	1. Внимательно прочитать текст задания и понять, что в качестве ответа ожидается только один из предложенных вариантов. 2. Внимательно прочитать предложенные вариант-ты ответа. 3. Выбрать один верный ответ. 4. Записать только номер (или букву) выбранного варианта ответа (например, 3 или В).	Ответ считается верным, если правильно указана цифра или буква
Задание закрытого типа на установление соответствия	Прочитайте текст и установите соответствие	1. Внимательно прочитать текст задания и понять, что в качестве ответа ожидаются пары элементов. 2. Внимательно прочитать оба списка: список 1 – вопросы, утверждения, факты, понятия и т.д.; список 2 – утверждения, свойства объектов и т.д. 3. Сопоставить элементы списка 1 с элементами списка 2, сформировать пары элементов. 4. Записать попарно буквы и цифры (в зависимости от задания) вариантов ответа (например, А1 или Б4).	Ответ считается верным, если правильно указаны цифры или буквы

Задание закрытого типа с выбором нескольких правильных ответов из нескольких вариантов предложенных	Прочитайте текст, выберите правильные ответы	1. Внимательно прочитать текст задания и понять, что в качестве ответа ожидается несколько правильных ответов из предложенных вариантов. 2. Внимательно прочитать предложенные вариант-ты ответа. 3. Выбрать несколько правильных ответов. 4. Записать только номера (или буквы) выбранного варианта ответа (например, 1 4 или А Г).	Ответ считается верным, если правильно установлены все соответствия (позиции из одного столбца верно сопоставлены с позициями другого)

Задание закрытого типа на установление последовательности	Прочитайте текст и установите последовательность	1. Внимательно прочитать текст задания и понять, что в качестве ответа ожидается последовательность элементов. 2. Внимательно прочитать предложенные варианты ответа. 3. Построить верную последовательность из предложенных элементов. 4. Записать буквы/цифры (в зависимости от задания) вариантов ответа в нужной последовательности (например, БВА или 135).	Ответ считается верным, если правильно указана вся последовательность цифр
Задание комбинированного типа с выбором одного правильного ответа из предложенных и обоснованием выбора	Прочитайте текст, выберите правильный ответ и запишите аргументы, обосновывающие выбор ответа	1. Внимательно прочитать текст задания и понять, что в качестве ответа ожидается только один из предложенных вариантов. 2. Внимательно прочитать предложенные варианты ответа. 3. Выбрать один верный ответ. 4. Записать только номер (или букву) выбранного варианта ответа. 5. Записать аргументы, обосновывающие выбор ответа (например, 4 текст обоснования).	Ответ считается верным, если правильно указана цифра или буква и приведены корректные аргументы, используемые при выборе ответа

Задание открытого типа с развернутым ответом	Прочитайте текст и запишите развернутый обоснованный ответ	1. Внимательно прочитать текст задания и понять суть вопроса. 2. Продумать логику и полноту ответа. 3. Записать ответ, используя четкие компактные формулировки. 4. В случае расчетной задачи, записать решение и ответ	Ответ считается верным: 1. Отсутствие фактических ошибок. 2. Раскрытие объема используемых понятий (полнота ответа). 3. Обоснованность ответа (наличие аргументов). 4. Логическая последовательность излагаемого материала.

4.5. Общая шкала оценивания результатов текущего контроля успеваемости и промежуточной аттестации обучающихся с применением БРС

Оценка по шкале ECTS	**Сумма баллов за все виды учебной деятельности**	**Оценка по государственной шкале**	**Определение**
A	90 – 100	«Отлично»	отличное выполнение с незначительным количеством неточностей
B	80 – 89	«Хорошо»	в целом правильно выполненная работа с незначительным количеством ошибок (до 10%)
C	75 – 79		в целом правильно выполненная работа с незначительным количеством ошибок (до 15%)
D	70 – 74	«Удовлетворительно»	неплохо, но со значительным количеством недостатков
E	60 – 69		выполнение удовлетворяет минимальные критерии
FX	35 – 59	«Не удовлетворительно»	с возможностью повторной сдачи
F	0 – 34		с обязательным повторным изучением дисциплины (выставляется комиссией)

Соотношение баллов за текущий контроль успеваемости и промежуточную аттестацию, а также повторную промежуточную аттестацию:

Максимальная сумма баллов за текущий контроль успеваемости	Максимальная сумма баллов за промежуточную аттестацию	Максимальная итоговая балльная оценка	Максимальная сумма баллов за повторную промежуточную аттестацию
100 баллов	100 баллов	100 баллов	100 баллов

5. Формы аттестации, типовые оценочные материалы для текущего контроля успеваемости обучающихся, критерии и шкалы оценивания по контрольным точкам

5.1. В ходе реализации дисциплины Б1.О.02.12 Основы программной инженерии используются следующие формы текущего контроля успеваемости обучающихся (в том числе, задания к контрольным точкам): доклад, устный опрос, тестирование, контрольные задания.

Таблица 5.1.

Распределение баллов по видам учебной деятельности (БРС)

Раздел/Темы	Формы текущего контроля			КТ
	УО	КЗ	Д	
Р-1. / Т-1	5		6	13
Р-1. / Т-2	5	5		
Р-2. / Т-3	5	5		13
Р-2. / Т-4	5	5		
Р-3. / Т-5	5	5		13
Р-3. / Т-6	5	5		
Итого: 100 б	30	25	6	39

УО – устный опрос;
ТЗ – тестовое задание;
КЗ – контрольные задания;
КТ – контрольная точка;
ПЗ – практическое занятие;
Д – доклад;
КЗР – контрольные работы по разделу.

Критерии оценивания опроса:

Балы	Описание критерия
5	Обучающийся полно излагает материал (отвечает на вопрос), дает правильное определение основных понятий; обнаруживает понимание материала, может обосновать свои суждения, применить знания на практике, привести необходимые примеры не только из учебника, но и самостоятельно составленные; излагает материал последовательно и правильно с точки зрения норм литературного языка.
3-4	Обучающийся дает ответ, удовлетворяющий тем же требованиям, что и для оценки «отлично», но допускает 1–2 ошибки, которые сам же исправляет, и 1–2 недочета в последовательности и языковом оформлении излагаемого.
1-2	Обучающийся обнаруживает знание и понимание основных положений данной темы, но излагает материал неполно и допускает неточности в определении понятий или формулировке правил; не умеет достаточно глубоко и доказательно обосновать свои суждения и привести свои примеры; излагает материал непоследовательно и допускает ошибки в языковом оформлении излагаемого.
0	Обучающийся обнаруживает незнание вопроса, допускает ошибки в формулировке определений и правил, искажающие их смысл, беспорядочно и неуверенно излагает материал.

0* - в журнал академической группы не выставляется

Критерии оценивания контрольных заданий:

Балы	Описание критерия	
5	Свыше 90% правильных ответов.	Обучающийся демонстрирует глубокое познание в освоенном материале.
3-4	Свыше 70% правильных ответов.	Обучающимся материал освоен полностью, без существенных ошибок.
1-2	Реализовано более 50% поставленных задач	Обучающимся материал освоен не полностью, имеются значительные пробелы в знаниях.
0	Реализовано менее 30% поставленных задач.	Обучающимся материал не освоен, знания обучающегося ниже базового уровня.

0* - в журнал академической группы не выставляется

Критерии оценивания контрольных точек:

Балы	Описание критерия
9-10	Обучающимся задание выполнено без ошибок и в полном объеме.
6-8	Обучающимся в целом задание выполнено, имеются отдельные неточности или недостаточно полные ответы, не содержащие ошибок.
3-5	Обучающимся допущены отдельные ошибки при выполнении задания
0-2	У обучающегося отсутствуют ответы на большинство вопросов задачи, задание не выполнено или выполнено не верно.

0* - в журнал академической группы не выставляется

Критерии оценивания доклада:

Критерии оценки	Диапазон баллов	Описание критерия
Содержание и раскрытие темы	2	Детальное, последовательное описание всех этапов с конкретными примерами
Грамотность изложения	1	Соблюдены все правила грамматики, орфографии и пунктуации
Стилистика	1	Единый стиль изложения, точные формулировки, уместное использование терминов, лаконичность
Логика изложения	1	Чёткая последовательность изложения, логические связи между частями текста, аргументы подтверждают выводы
Оригинальность	1	Уникальный подход к теме, нестандартные решения, инновационные идеи, собственная позиция автора
Итого максимально:	6	

5.2. Типовые оценочные материалы для текущего контроля успеваемости обучающихся (вне контрольных работы):

РАЗДЕЛ 1. ВВЕДЕНИЕ И ЖИЗНЕННЫЙ ЦИКЛ ПО

Тема 1. Введение в предмет. Основные понятия.

Контрольные вопросы для проведения опроса:

1. Дайте определение программной инженерии. Чем она отличается от программирования?
2. Перечислите основные компоненты программной инженерии как дисциплины.
3. Что такое программная инженерия?
4. В чем отличие программной инженерии от информатики?
5. В чем отличие программной инженерии от системотехники?
6. Приведите примеры дисциплин информатики и программной инженерии (дисциплины не путать с учебными предметами).
7. Перечислите характеристики ПО по Бруксу и кратко характеризуйте каждую.
8. С какими иными видами человеческой деятельности соотносится создание ПО в данном разделе?
9. Что такое процесс создания ПО?
10. Расскажите о причинах отсутствия универсального процесса разработки ПО.

Перечень тем для доклада:

1. История возникновения программной инженерии.
2. Проблемы разработки сложного программного обеспечения.
3. Кризис программного обеспечения.
4. Современные методы инженерного творчества.
5. Системная инженерия. Междисциплинарный подход.

Тема 2. Классические модели ЖЦ

Контрольные вопросы для проведения опроса:

1. Что такое жизненный цикл программного обеспечения?
2. Опишите каскадную модель ЖЦ. Назовите её преимущества и недостатки.
3. В чём особенности V-модели жизненного цикла?
4. Чем итеративная модель отличается от каскадной?
5. Что такое спиральная модель? Для каких проектов она подходит?
6. Назовите основные принципы Agile (без глубокого управления проектами).
7. В чём ключевые отличия Scrum от Kanban на уровне базовых практик?
8. Перечислите типичные роли в команде разработки ПО.
9. Какова роль бизнес-аналитика в жизненном цикле?
10. Что такое техническое задание и как оно связано с ЖЦ?

11. Почему классические модели часто критикуют за жёсткость?

Контрольные задания:

Задание открытого типа с развернутым ответом.

Задание 1.

Разработайте схему выбора модели жизненного цикла (каскадная, V-модель, итеративная, спиральная) для трёх заданных типов проектов:

- система управления космическим аппаратом;
- мобильное приложение для доставки еды с еженедельными обновлениями;
- научный прототип с меняющимися требованиями.

Обоснуйте выбор для каждого проекта в письменной форме (1–2 абзаца на проект)

Задание 2.

Выберите открытую спецификацию любого API (например, OpenAPI для публичного сервиса погоды). Проанализируйте её с точки зрения полноты документации. Напишите замечания (3–5 пунктов) и предложите улучшения.

РАЗДЕЛ 2. РАБОТА С ТРЕБОВАНИЯМИ И ПРОЕКТИРОВАНИЕ

Тема 3. Сбор и анализ требований. Спецификация.

Контрольные вопросы для проведения опроса:

1. Чем функциональные требования отличаются от нефункциональных? Приведите примеры.
2. Что входит в процесс сбора и анализа требований?
3. Что такое спецификация требований к ПО (SRS)? Какие разделы она должна содержать?
4. Назовите основные признаки качественного требования (например, по стандарту IEEE).
5. Почему важно отделять требования от деталей реализации?
6. Что такое «заморозка требований» и когда она применяется?

Контрольное задание:

Задание открытого типа с развернутым ответом.

Задание 1.

Возьмите любой простой программный продукт (например, калькулятор, список задач, заметки). Составьте фрагмент спецификации требований (SRS), включающий:

- минимум 4 функциональных требования;
- минимум 3 нефункциональных требования (производительность, безопасность, удобство использования).

Оформите в виде нумерованного документа с заголовками

Задание 2.

Для системы «Управление очередями в поликлинике» (запись, отмена, вызов пациента, статистика) постройте две диаграммы:

- компонентную диаграмму (UML);
- диаграмму развёртывания (физические узлы, соединения).

Обоснуйте выбранное распределение компонентов по узлам

Тема 4. Основы архитектурного проектирования

Контрольные вопросы для проведения опроса:

1. Что такое архитектурное проектирование ПО?
2. Дайте определение компонента и модуля. В чём разница?
3. Что такое интерфейс в программной инженерии?
4. В чём суть нисходящего и восходящего проектирования? Сравните их.
5. Для чего используется язык UML?
6. Что показывает диаграмма вариантов использования?
7. Какая информация содержится в диаграмме классов?
8. Что моделирует диаграмма последовательности (sequence diagram)?
9. Как связаны между собой диаграммы вариантов использования и диаграммы классов?

Контрольные задания:

Задание открытого типа с развернутым ответом.

Задание 1.

Для системы «Онлайн-библиотека» (выдача книг, возврат, поиск, регистрация читателя) постройте три диаграммы:

- диаграмму вариантов использования (не менее 4 актёров, 6 прецедентов);
- диаграмму классов (минимум 4 класса, атрибуты, методы, связи);
- диаграмму последовательности для сценария «Пользователь берёт книгу».

Инструмент — любой (ручной рисунок, draw.io, PlantUML)

Задание 2.

Возьмите код любого простого алгоритма (сортировка, поиск, калькулятор). Определите, какие архитектурные паттерны из списка GoF можно применить для его рефакторинга. Выберите один паттерн, опишите его реализацию на псевдокоде и объясните, как он улучшит сопровождаемость

РАЗДЕЛ 3. РЕАЛИЗАЦИЯ, ТЕСТИРОВАНИЕ И КАЧЕСТВО

Тема 5. Стандарты кодирования. Виды тестирования.

Контрольные вопросы для проведения опроса:

1. Что такое стандарты кодирования?

2. Приведите примеры элементов стандарта.
3. Для чего проводится ревью кода?
4. Какие виды ревью существуют?
5. Чем модульное тестирование отличается от интеграционного?
6. Что такое системное тестирование? Когда оно выполняется?
7. Опишите приемочное тестирование. Кто обычно его проводит?
8. Что такое тест-дизайн? Назовите два основных метода: эквивалентное разбиение и граничные значения.
9. Какова роль автоматического тестирования в модульном тестировании?
10. Что такое тестовое покрытие? Почему 100% покрытие не гарантирует отсутствие ошибок?
11. Почему тестирование не может доказать отсутствие дефектов?

Контрольное задание:

Задание открытого типа с развернутым ответом.

Задание 1.

Дана функция: вход — возраст (целое число от 0 до 120) и стаж работы (целое от 0 до 50). Программа возвращает «допущен»/«не допущен». Условие: допускаются лица от 18 до 60 лет включительно со стажем не менее 2 лет.

Примените метод эквивалентного разбиения и анализ граничных значений:

- выделите классы эквивалентности;
- составьте набор тестовых случаев (входные данные + ожидаемый результат), покрывающих границы.

Задание 2.

Дано описание функции: «проверка номера банковской карты по алгоритму Луна». Составьте план верификации и валидации:

- два метода статической верификации;
- тест-кейсы для валидации (не менее 5, включая граничные и некорректные номера).

Оформите как таблицу (ID входа, входные данные, ожидаемый результат, тип проверки)

Тема 6. Отладка и логирование. Качество ПО

Контрольные вопросы для проведения опроса:

1. В чём разница между отладкой (debugging) и логированием (logging)?
2. Для чего нужны логи в программном продукте?
3. Приведите примеры уровней логирования.
4. Дайте определение качества ПО.

5. Назовите основные характеристики качества (по ISO 25010).
6. Что такое метрики качества? Приведите 2–3 примера.
7. Как надёжность ПО связана с количеством дефектов?
8. Чем ошибка (error) отличается от дефекта (defect) и отказа (failure)

Контрольное задание:

Задание открытого типа с развернутым ответом.

Задание 1.

Дан фрагмент кода на знакомом вам языке (предоставлен преподавателем или выбран из открытых источников). Выполните:

- найдите 3–4 потенциальных проблемы (стиль, логика, безопасность);
- предложите, куда и с каким уровнем (debug, info, error) добавить логирование в этом фрагменте;
- напишите краткое заключение о качестве кода (5–7 предложений)

Задание 2.

Возьмите любой открытый баг-репорт из трекера (GitHub Issues, Jira public). Проанализируйте его качество:

- заполнены ли обязательные поля;
- воспроизводим ли сценарий;
- предложите дополнения (логи, окружение).

Напишите улучшенную версию этого баг-репорта (на основе вымышленного или реального, но изменённого)

5.3. Один или несколько тематических блоков дисциплины завершаются контрольной точкой по разделу (далее – КТ). Текущий контроль успеваемости по дисциплине предусматривает не менее 2 (двух) и не более 10 (десяти) КТ в течение периода освоения дисциплины.

Максимальное количество баллов за любой тип работ в рамках КТ составляет 100 (сто) баллов.

Распределение весовых коэффициентов по КТ в рамках текущего контроля успеваемости по дисциплине и формулы расчета:

Наименование контрольной точки	Максимальное количество баллов за работу в рамках КТ, которое может набрать студент	Коэффициент веса контрольной точки	Результат контрольной точки, участвующий в формировании итоговой балльной оценки по дисциплине
КТ№1	100	0,13	13
КТ№2	100	0,13	13
КТ№2	100	0,13	13
Итого:	x	0,39	39

Формула расчета результата контрольной точки:

Результат контрольной точки = Количество баллов за работу в рамках КТ X Коэффициент веса контрольной точки.

5.4. Формы текущего контроля успеваемости обучающихся в рамках КТ и типовые оценочные материалы:

Контрольная точка №1.

Задание 1.

Выберите один из трёх проектов:

- система управления лифтом в 50-этажном здании,
- мобильное приложение для фотофильтров,
- корпоративная CRM на 5 лет).

Обоснуйте выбор модели жизненного цикла (каскадная, V, итеративная, спиральная, Agile без глубокого управления).

Укажите 3 критерия выбора и 2 ограничения выбранной модели?

Задание 2.

Нарисуйте схему перехода между фазами каскадной модели и

отметьте, в каких случаях возможен возврат на предыдущую фазу. Опишите одну реальную ситуацию, когда возврат неизбежен.

Задание 3.

Сравните роли «архитектор ПО» и «тимлид» в традиционных (каскадных) и гибких (Agile) проектах.

Результат оформите в виде таблицы из 4 строк: область ответственности, подчинение, артефакты, участие в совещаниях?

Контрольная точка №2.

Задание 1.

Для системы «Умная домофония» (открытие двери по коду/ключу/приложению, видеозвонок, запись событий) напишите спецификацию из 6 требований в формате:

- 2 функциональных (с приоритетом «высокий»);
- 2 нефункциональных (безопасность, надёжность);
- 2 ограничения (технологические, бюджетные)

Задание 2.

Возьмите диаграмму классов простой системы (например, интернет-магазин: Товар, Корзина, Заказ, Пользователь).

Добавьте к ней недостающие атрибуты, методы, типы связей (ассоциация, агрегация, композиция, наследование).

Кратко объясните, почему вы выбрали именно такие отношения?

Задание 3.

Постройте диаграмму последовательности для сценария: «Пользователь сбрасывает пароль через электронную почту в веб-приложении».

Укажите объекты, их линии жизни, сообщения (синхронные/асинхронные), альтернативный поток (если письмо не пришло)

Контрольная точка №3.

Задание 1.

Дан фрагмент псевдокода (вычисление скидки в зависимости от суммы и статуса клиента).

Предложите набор тест-кейсов по методам эквивалентного разбиения и

граничных значений.

Оформите как таблицу: ID, входные данные (сумма, статус), ожидаемый результат, метод тест-дизайна.

Задание 2.

Выберите любую открытую библиотеку на GitHub. Проанализируйте её README и файлы документации.

Напишите список из трёх рекомендаций, как улучшить документацию для повышения качества и удобства использования.

Задание 3.

Представьте, что вы провели ревью кода и обнаружили три типовые ошибки:

- необрабатываемое исключение;
- жёстко закодированный пароль;
- метод длиной 250 строк.

Для каждой ошибки опишите:

- почему это снижает качество;
- как исправить;
- какой вид тестирования мог бы её выявить раньше

6. Формы промежуточной аттестации, критерии и шкала оценивания, типовые оценочные материалы по дисциплине

6.1. Промежуточная аттестация проводится в форме *зачета* в шестом семестре в устной форме (устный опрос). Обучающемуся предлагаются 1-2 вопроса по каждой теме. Необходимо дать ответ в устной форме, подробно изложив ход мыслей. Ответ может быть дополнен схемами, диаграммами, графиками и т.д., изображаемых письменно.

6.2. Типовые оценочные материалы промежуточной аттестации.

РАЗДЕЛ 1. ВВЕДЕНИЕ И ЖИЗНЕННЫЙ ЦИКЛ ПО

Тема 1. Введение в предмет. Основные понятия. ОПК-2.2

Вопросы для устного опроса:

1. Дайте определение программной инженерии. Чем она отличается от программирования?
2. Перечислите основные компоненты программной инженерии как дисциплины.
3. Что такое программная инженерия?
4. В чем отличие программной инженерии от информатики?
5. В чем отличие программной инженерии от системотехники?
6. Приведите примеры дисциплин информатики и программной инженерии (дисциплины не путать с учебными предметами).
7. Перечислите характеристики ПО по Бруксу и кратко характеризуйте каждую.
8. С какими иными видами человеческой деятельности соотносится создание ПО в данном разделе?
9. Что такое процесс создания ПО?
10. Расскажите о причинах отсутствия универсального процесса разработки ПО

Тема 2. Классические модели ЖЦ. ОПК-2.2

Вопросы для устного опроса:

1. Что такое жизненный цикл программного обеспечения?
2. Опишите каскадную модель ЖЦ. Назовите её преимущества и недостатки.
3. В чём особенности V-модели жизненного цикла?
4. Чем итеративная модель отличается от каскадной?

5. Что такое спиральная модель? Для каких проектов она подходит?
6. Назовите основные принципы Agile (без глубокого управления проектами).
7. В чём ключевые отличия Scrum от Kanban на уровне базовых практик?
8. Перечислите типичные роли в команде разработки ПО.
9. Какова роль бизнес-аналитика в жизненном цикле?
10. Что такое техническое задание и как оно связано с ЖЦ?
11. Почему классические модели часто критикуют за жёсткость?

РАЗДЕЛ 2. РАБОТА С ТРЕБОВАНИЯМИ И ПРОЕКТИРОВАНИЕ

Тема 3. Сбор и анализ требований. Спецификация. ОПК-2.2

Вопросы для устного опроса:

1. Чем функциональные требования отличаются от нефункциональных? Приведите примеры.
2. Что входит в процесс сбора и анализа требований?
3. Что такое спецификация требований к ПО (SRS)? Какие разделы она должна содержать?
4. Назовите основные признаки качественного требования (например, по стандарту IEEE).
5. Почему важно отделять требования от деталей реализации?
6. Что такое «заморозка требований» и когда она применяется

Тема 4. Основы архитектурного проектирования. ОПК-2.2

Вопросы для устного опроса:

1. Что такое архитектурное проектирование ПО?
2. Дайте определение компонента и модуля. В чём разница?
3. Что такое интерфейс в программной инженерии?
4. В чём суть нисходящего и восходящего проектирования? Сравните их.
5. Для чего используется язык UML?
6. Что показывает диаграмма вариантов использования?
7. Какая информация содержится в диаграмме классов?
8. Что моделирует диаграмма последовательности (sequence diagram)?
9. Как связаны между собой диаграммы вариантов использования и диаграммы классов?

РАЗДЕЛ 3. РЕАЛИЗАЦИЯ, ТЕСТИРОВАНИЕ И КАЧЕСТВО

Тема 5. Стандарты кодирования. Виды тестирования. ОПК-2.2

Вопросы для устного опроса:

1. Что такое стандарты кодирования?

2. Приведите примеры элементов стандарта.
3. Для чего проводится ревью кода?
4. Какие виды ревью существуют?
5. Чем модульное тестирование отличается от интеграционного?
6. Что такое системное тестирование? Когда оно выполняется?
7. Опишите приемочное тестирование. Кто обычно его проводит?
8. Что такое тест-дизайн? Назовите два основных метода: эквивалентное разбиение и граничные значения.
9. Какова роль автоматического тестирования в модульном тестировании?
10. Что такое тестовое покрытие? Почему 100% покрытие не гарантирует отсутствие ошибок?
11. Почему тестирование не может доказать отсутствие дефектов?

Тема 6. Отладка и логирование. Качество ПО. ОПК-2.2

Вопросы для устного опроса:

1. В чём разница между отладкой (debugging) и логированием (logging)?
2. Для чего нужны логи в программном продукте?
3. Приведите примеры уровней логирования.
4. Дайте определение качества ПО.
5. Назовите основные характеристики качества (по ISO 25010).
6. Что такое метрики качества? Приведите 2–3 примера.
7. Как надёжность ПО связана с количеством дефектов?
8. Чем ошибка (error) отличается от дефекта (defect) и отказа (failure)

6.3. Критерии и шкала оценивания на основе БРС.

Соответствие государственной шкалы оценивания академической успеваемости и шкалы ECTS при экзамене

Оценка по шкале ECTS	**Сумма баллов за все виды учебной деятельности**	**Оценка по государственной шкале**	**Определение**
A	90 – 100	«Отлично»	отличное выполнение с незначительным количеством неточностей
B	80 – 89	«Хорошо»	в целом правильно выполненная работа с незначительным количеством ошибок (до 10%)
C	75 – 79		в целом правильно выполненная работа с незначительным количеством ошибок (до 15%)
D	70 – 74	«Удовлетворительно»	неплохо, но со значительным количеством недостатков
E	60 – 69		выполнение удовлетворяет минимальные критерии
FX	35 – 59	«Не удовлетворительно»	с возможностью повторной сдачи
F	0 – 34		с обязательным повторным изучением дисциплины (выставляется комиссией)

6.4. Описание дополнительных материалов и оборудования, необходимых для выполнения проверочных заданий

Компьютер с операционной системой РЕД ОС или MS Windows, на котором установлены Р7-Офис или MS Office.

7. Методические материалы по освоению дисциплины

Получение углубленных знаний по изучаемой дисциплине достигается за счет дополнительных часов к аудиторной работе самостоятельной работы студентов. Выделяемые часы целесообразно использовать для знакомства с дополнительной научной литературой по проблематике дисциплины, анализа научных концепций и современных подходов к осмыслению рассматриваемых проблем. К самостоятельному виду работы студентов относится работа в библиотеках, в электронных поисковых системах и т.п. по сбору материалов, необходимых для проведения практических занятий или выполнения конкретных заданий преподавателя по изучаемым темам. Студенты могут установить диалог с преподавателем, получать консультации по выполнению заданий. В качестве оценочных средств на протяжении семестра используются тестовые и иные задания.

Обучение по дисциплине «Основы программной инженерии» предполагает изучение курса на аудиторных занятиях (лекции, практические занятия) и самостоятельную работу студентов. Практические занятия дисциплины предполагают их проведение в различных формах с целью выявления полученных знаний, умений, навыков и компетенций с проведением контрольных мероприятий. С целью обеспечения успешного обучения студент должен готовиться к лекции, поскольку она является важнейшей формой организации учебного процесса, поскольку:

– знакомит с новым учебным материалом;
– разъясняет учебные элементы, трудные для понимания;
– систематизирует учебный материал;
– ориентирует в учебном процессе.

Работа обучающегося на лекции:

Слушание и запись лекций – сложный вид вузовской аудиторной работы. Внимательное слушание и конспектирование лекций предполагает интенсивную умственную деятельность обучающегося. Краткие записи лекций, их конспектирование помогает усвоить учебный материал. Конспект является полезным тогда, когда записано самое существенное, основное и сделано это самим обучающимся.

Подготовка к практическим занятиям:

Подготовку к каждому практическому занятию каждый обучающийся должен начать с ознакомления с планом, который отражает содержание предложенной темы. Тщательное продумывание и изучение вопросов плана основывается на проработке текущего материала лекции, а затем изучения обязательной и дополнительной литературы, рекомендованную к данной теме. Если программой дисциплины предусмотрено выполнение практического задания, то его необходимо выполнить с учетом предложенной инструкции. Все новые понятия по изучаемой теме необходимо внести в глоссарий, который целесообразно вести с самого начала изучения курса. Результат такой работы должен проявиться в способности обучающегося свободно ответить на теоретические вопросы практического занятия, его выступлении и участии в коллективном обсуждении вопросов изучаемой темы, правильном выполнении практических заданий и контрольных работ.

Структура практического занятия:

В зависимости от содержания и количества отведенного времени на изучение каждой темы может практическое занятие состоять из четырех-пяти частей:

1. Устный опрос.

2. Обсуждение теоретических вопросов, определенных программой дисциплины.

3. Выполнение практических заданий с последующим разбором полученных результатов или обсуждение практического задания, выполненного дома.

4. Подведение итогов занятия.

Работа с литературными источниками:

В процессе подготовки к практическим занятиям, обучающимся необходимо обратить особое внимание на самостоятельное изучение рекомендованной учебно-методической (а также научной и популярной) литературы. Самостоятельная работа с учебниками, учебными пособиями, научной, справочной и популярной литературой, материалами периодических изданий и Интернета, статистическими данными является наиболее эффективным методом получения знаний, позволяет значительно активизировать процесс овладения информацией, способствует более глубокому усвоению изучаемого материала, формирует у обучающихся свое отношение к конкретной проблеме. Более глубокому раскрытию вопросов способствует знакомство с дополнительной литературой, рекомендованной преподавателем, что позволяет обучающимся проявить свою индивидуальность в рамках выступления на занятиях, выявить широкий спектр мнений по изучаемой проблеме.

8. Учебная литература и ресурсы информационно-телекоммуникационной сети Интернет

8.1. Основная литература

1. Парамонов, А. И. Основы программной инженерии : учебно-методическое пособие / А. И. Парамонов, Н. В. Лапицкая, С. Н. Нестеренков. — Минск : БГУИР, 2023. — 121 с. — ISBN 978-985-543-699-8. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/479501 (дата обращения: 18.05.2026). — Режим доступа: для авториз. пользователей.

2. Основы системной и программной инженерии : учебное пособие / К. В. Гусев, А. Н. Миронов, Е. А. Чернов, М. Б. Туманова. — Москва : РТУ МИРЭА, 2023 — Часть 1 — 2023. — 140 с. — ISBN 978-5-7339-1761-0. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/368930 (дата обращения: 18.05.2026). — Режим доступа: для авториз. пользователей.

3. Ехлаков, Ю. П. Основы программной инженерии : учебное пособие / Ю. П. Ехлаков. — Москва : ТУСУР, 2019. — 128 с. — ISBN 978-5-4332-0280-1. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/313592 (дата обращения: 18.05.2026). — Режим доступа: для авториз. пользователей.

4. Маран, М. М. Программная инженерия : Учебное пособие для вузов / М. М. Маран. — 3-е изд., стер. — Санкт-Петербург : Лань, 2022. — 196 с. — ISBN 978-5-8114-9323-4. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/189470 (дата обращения: 18.05.2026). — Режим доступа: для авториз. пользователей.

8.2. Дополнительная литература

5. Турнецкая, Е. Л. Программная инженерия. Интеграционный подход к разработке : учебник для вузов / Е. Л. Турнецкая, А. В. Аграновский. — 2-е изд., стер. — Санкт-Петербург : Лань, 2025. — 216 с. — ISBN 978-5-507-50848-8. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/480161 (дата обращения: 18.05.2026). — Режим доступа: для авториз. пользователей.

6. Минакова, О. В. Программная инженерия. Основные принципы, методы и инструменты : учебник для вузов / О. В. Минакова. — 2-е изд., стер. — Санкт-Петербург : Лань, 2026. — 212 с. — ISBN 978-5-507-55013-5. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/515097 (дата обращения: 18.05.2026). — Режим доступа: для авториз. пользователей.

7. Турнецкая, Е. Л. Программная инженерия. Тестирование и

контроль качества программного обеспечения : учебное пособие для вузов / Е. Л. Турнецкая, А. В. Аграновский. — Санкт-Петербург : Лань, 2025. — 172 с. — ISBN 978-5-507-51677-3. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/455672 (дата обращения: 18.05.2026). — Режим доступа: для авториз. пользователей.

8. Мейер, Б. Объектно-ориентированное программирование и программная инженерия : учебное пособие / Б. Мейер. — 2-е изд. — Москва : ИНТУИТ, 2016. — 285 с. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/100271 (дата обращения: 18.05.2026). — Режим доступа: для авториз. пользователей.

8.3.Нормативные правовые документы и иная правовая информация

1. Конституция Российской Федерации. – Текст : электронный // Сайт Президента Российской Федерации. – URL: http://www.kremlin.ru/acts/constitution

8.4 Интернет-ресурсы

1. Академия Microsoft: Введение в программную инженерию (ИНТУИТ) – URL: https://new2.intuit.ru/studies/courses/497/353/info

2. ТПУ: Программирование и создание ИТ-продуктов – URL: http://web.tpu.ru/webcenter/portal/oit/page16/page150

3. Carnegie Mellon University: Foundations of Software Engineering (15-313) – URL: https://www.cs.cmu.edu/~ckaestne/15313/2016/index.html

4. Электронно-библиотечная система «Лань». – URL: http://e.lanbook.com

5. Основы программной инженерии (БНТУ, 2023) – URL: https://library.bntu.by/books/paramonov-a-i-osnovy-programmnoj-inzhenerii/

9. Материально-техническая база, информационные технологии, программное обеспечение и информационные справочные системы

Материально-техническое обеспечение дисциплины включает в себя:

- лекционные аудитории, оборудованные видеопроекционным оборудованием для презентаций, средствами звуковоспроизведения, экраном;
- помещения для проведения семинарских и практических занятий, оборудованные учебной мебелью.

Дисциплина поддержана соответствующими программными продуктами с открытой лицензией: РЕД ОС, Р7-Офис.

Вуз обеспечивает каждого обучающегося рабочим местом в

компьютерном классе в соответствии с объемом изучаемых дисциплин, обеспечивает выход в сеть Интернет.

Помещения для самостоятельной работы обучающихся включают следующую оснащенность: столы аудиторные, стулья, доски аудиторные, компьютеры с подключением к локальной сети института (для компьютерных аудиторий) и Интернет. Для изучения учебной дисциплины используются автоматизированная библиотечная информационная система и электронные библиотечные системы.