

Министерство сельского хозяйства Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ АГРАРНЫЙ УНИВЕРСИТЕТ
ИМЕНИ ИМПЕРАТОРА ПЕТРА I»

УТВЕРЖДАЮ

Декан экономического факультета

 А.Н. Черных

«21» мая 2024г.



РАБОЧАЯ ПРОГРАММА ПО ДИСЦИПЛИНЕ

Б1.В.10 Инструменты и методы программной инженерии

Направление: 09.03.03 Прикладная информатика

Профиль: Информационные системы и технологии в менеджменте АПК

Квалификация выпускника: бакалавр

Кафедра Информационного обеспечения и моделирования агроэкономических систем

Разработчик рабочей программы:

Должность:

Ученая степень:

Ученое звание:

Кузнецова Елена Дмитриевна

доцент

кандидат экономических наук

доцент

Воронеж-2024

Рабочая программа разработана в соответствии с Федеральным государственным образовательным стандартом высшего образования по направлению подготовки 09.03.03 Прикладная информатика (уровень бакалавриата) (утвержден приказом Министерства образования и науки РФ от 19 сентября 2017 № 922).

Рабочая программа утверждена на заседании кафедры Информационного обеспечения и моделирования агроэкономических систем (протокол № 8 от 26.04.2024 г.)

Заведующий кафедрой:



Р.В. Подколзин

Рабочая программа рекомендована к использованию в учебном процессе на заседании методической комиссии экономического факультета (протокол №9 от 21.05.2024 г.).

Председатель методической комиссии:



Л.В. Брянцева

Рецензент: начальник отдела информационно-коммуникационных технологий ООО "Овощ-Прод-Холдинг" А.П. Сухоедов

Содержание рабочей программы

1. Общая характеристика дисциплины
 - 1.1. Цель дисциплины
 - 1.2. Задачи дисциплины
 - 1.3. Предмет дисциплины
 - 1.4. Место в образовательной программе
 - 1.5. Связь с другими дисциплинами
 - 1.6. Особенности реализации дисциплины для инвалидов и лиц с ограниченными возможностями здоровья
2. Планируемые результаты изучения дисциплины
3. Объем дисциплины и виды учебной работы
 - 3.1. Очная форма обучения
 - 3.2. Заочная форма обучения
4. Содержание дисциплины
 - 4.1. Содержание дисциплины в разрезе разделов и подразделов
 - 4.2. Распределение контактной и самостоятельной работы по подразделам
 - 4.3. Перечень тем и учебно-методического обеспечения для самостоятельной работы обучающихся
5. Фонд оценочных средств
 - 5.1. Этапы формирования компетенций
 - 5.2. Шкалы и критерии оценивания достижения компетенций
 - 5.2.1. Шкалы академических оценок освоения дисциплины
 - 5.2.2. Критерии оценки достижения компетенций в ходе освоения дисциплины
 - 5.3. Материалы для оценки достижения компетенций
 - 5.3.1. Вопросы к экзамену
 - 5.3.2. Задания к экзамену
 - 5.3.3. Вопросы к зачету с оценкой
 - 5.3.4. Вопросы к зачету
 - 5.3.5. Темы курсового проекта (работы) и вопросы к защите
 - 5.3.4.1. Темы курсового проекта (работы)
 - 5.3.4.2. Вопросы к защите курсового проекта (работы)
 - 5.3.6. Вопросы тестов
 - 5.3.7. Вопросы для устного опроса
 - 5.3.8. Задания для проверки формирования умений и навыков
 - 5.4. Система оценивания достижения компетенций
 - 5.4.1. Оценка достижения компетенций в ходе промежуточной аттестации
 - 5.4.2. Оценка достижения компетенций в ходе текущего контроля
6. Учебно-методическое и информационное обеспечение дисциплины
 - 6.1. Рекомендуемая литература
 - 6.2. Ресурсы сети Интернет
 - 6.2.1. Электронные библиотечные системы
 - 6.2.2. Профессиональные базы данных и информационные системы
 - 6.2.3. Сайты и информационные порталы
7. Материально-техническое и программное обеспечение дисциплины
 - 7.1. Помещения для ведения образовательного процесса и оборудование
 - 7.2. Программное обеспечение
 - 7.2.1. Программное обеспечение общего назначения
 - 7.2.2. Специализированное программное обеспечение
8. Междисциплинарные связи

1. Общая характеристика дисциплины

1.1. Цель дисциплины:

формирование знаний, умений и навыков использования современной и эффективной методологии и инструментария разработки программного обеспечения

1.2. Задачи дисциплины:

изучение специфики методологии программной инженерии, рассмотрение формальных методов и методов прототипирования;

изучение современных эвристических методов программной инженерии;

формирование знаний, умений и навыков использования инструментов управления требованиями;

формирование знаний, умений и навыков применения инструментов и программного обеспечения проектирования и конструирования ПО;

формирование знаний, умений и навыков использования методологии, инструментов и программного обеспечения для тестирования и сопровождения ПО;

формирование знаний, умений и навыков использования методологий и инструментов верификации и оценки качества ПО;

формирование знаний, умений и навыков применения инструментов конфигурационного управления ПО.

1.3. Предмет дисциплины:

современные методологии и инструменты разработки программного обеспечения

1.4. Место в образовательной программе:

часть, формируемая участниками образовательных отношений

1.5. Взаимосвязь с другими дисциплинами:

Б1.О.24 Программная инженерия

Б1.В.12 Управление IT-проектами

1.6. Особенности реализации дисциплины для инвалидов и лиц с ограниченными возможностями здоровья

Особенности реализации дисциплины для инвалидов и лиц с ограниченными возможностями здоровья определяются в индивидуальном порядке исходя из специфики заболевания и требований, указанных в Основной образовательной программе

2. Планируемые результаты изучения дисциплины

Компетенция		Индикатор достижения компетенции	
Код	Содержание	Код	Содержание
ПК-01	Способность проводить обследование организаций, выявлять информационные потребности пользователей, формировать требования к информационной системе	З1	подходы к разработке, функционированию и сопровождению программного обеспечения
		У1	выявлять и формулировать требования к структуре и функционалу информационных систем
		Н1	применения инструментов и методов программной инженерии при формировании требований к информационной системе
ПК-02	Способность разрабатывать и адаптировать прикладное программное обеспечение	З1	инструменты и методы прототипирования пользовательского интерфейса
		У1	применять инструменты и методы прототипирования пользовательского интерфейса
		Н1	выдачи экспертных заключений по предложенным решениям по реализации интерфейсов и форматов обмена данными
ПК-07	Способность настраивать, эксплуатировать и сопровождать информационные системы и сервисы	З2	инструменты и методы верификации и интеграции ИС
		З9	функции систем управления конфигурациями
		У2	оценивать параметры информационных систем
		У9	документировать изменения конфигураций
		Н2	разработки системы оценки качества информационных систем
		Н10	работы со средствами конфигурационного управления
ПК-08	Способность проводить тестирование	З1	инструменты и методы тестирования информационных систем
		У1	тестировать информационные системы на соответствие параметрам качества
		Н1	проведения тестирования, обработки и использования их результатов

3. Объем дисциплины и виды учебной работы

3.1. Очная форма обучения

Показатели	Семестр	Всего
	7	
Общая трудоёмкость, з.е./ч	4 / 144	4 / 144
Общая контактная работа, ч	40,75	40,75
Общая самостоятельная работа, ч	103,25	103,25
Контактная работа при проведении учебных занятий, в т.ч. (ч)	40,00	40,00
лекции	14	14,00
практические-всего	26	26,00
в т.ч. практическая подготовка	6	6,00
Самостоятельная работа при проведении учебных занятий, ч	85,50	85,50
Контактная работа при проведении промежуточной аттестации обучающихся, в т.ч. (ч)	0,75	0,75
групповые консультации	0,50	0,50
экзамен	0,25	0,25
Самостоятельная работа при промежуточной аттестации, в т.ч. (ч)	17,75	17,75
подготовка к экзамену	17,75	17,75
Форма промежуточной аттестации	экзамен	экзамен

3. Объем дисциплины и виды учебной работы

3.2. Заочная форма обучения

Показатели	Курс	Всего
	4	
Общая трудоёмкость, з.е./ч	4 / 144	4 / 144
Общая контактная работа, ч	22,75	22,75
Общая самостоятельная работа, ч	121,25	121,25
Контактная работа при проведении учебных занятий, в т.ч. (ч)	22,00	22,00
лекции	10	10,00
практические-всего	12	12,00
в т.ч. практическая подготовка	2	2,00
Самостоятельная работа при проведении учебных занятий, ч	103,50	103,50
Контактная работа при проведении промежуточной аттестации обучающихся, в т.ч. (ч)	0,75	0,75
групповые консультации	0,50	0,50
экзамен	0,25	0,25
Самостоятельная работа при промежуточной аттестации, в т.ч. (ч)	17,75	17,75
подготовка к экзамену	17,75	17,75
Форма промежуточной аттестации	экзамен	экзамен

4. Содержание дисциплины

4.1. Содержание дисциплины в разрезе разделов и подразделов

Раздел 1.

Методы программной инженерии

Подраздел 1.1.

Формальные методы и методы прототипирования

метод программной инженерии; понятие и виды формальных методов; понятие и виды прототипирования ПО

Подраздел 1.2.

Эвристические методы

понятие эвристических методов; структурные методы; методы, ориентированные на данные; объектно-ориентированные методы; ориентированные на область применения методы

Раздел 2.

Инструменты программной инженерии

Подраздел 2.1.

Инструменты работы с требованиями

понятие и свойства требования; инструменты работы с требованиями по SWEBOOK; операции с требованиями; ПО управления требованиями

Подраздел 2.2.

Инструменты проектирования ии конструирования

краткая характеристика языков BPMN, SDL, MSC; классификация диаграмм языка UML; проектирование пользовательского интерфейса; программные средства визуального моделирования; классификация и характеристика инструментов конструирования по SWEBOOK

Подраздел 2.3.

Инструменты и методы тестирования и сопровождения ПО

классификация инструментов тестирования ПО по SWEBOOK и их краткая характеристика; методы и техники тестирования ПО; программные средства тестирования; инструменты сопровождения согласно SWEBOOK; реинжиниринг и рефакторинг ПО; программным средствам обнаружения ошибок

Подраздел 2.4.

Инструменты верификации и обеспечения качества

методы верификации; характеристики качества ПО; оценки значений показателей качества; математические модели надежности

Подраздел 2.5.

Инструменты конфигурационного управления

инструменты конфигурационного управления согласно SWEBOOK; понятие, виды и содержание систем управления версиями; бэктрекинг системы; программы автоматизации сборки ПО

Практическая подготовка по дисциплине включает проведение лекций, практических занятий по дисциплине "Инструменты и методы программной инженерии" на профильных предприятиях (организациях) с использованием их материально-технической базы (ООО «Цифровые технологии») или в структурных подразделениях Университета (компьютерные аудитории №120) в объеме, указанном в таблицах 3.1. и 3.2.

Практическая подготовка осуществляется по следующим темам (разделам) дисциплины: Подраздел 2.2 Инструменты проектирования ии конструирования. Подраздел 2.5. Инструменты конфигурационного управления

4.2. Распределение контактной и самостоятельной работы по подразделам
Очная форма обучения

Разделы, подразделы дисциплины	Контактная работа		СР
	лекции	ПЗ	
Методы программной инженерии			
Формальные методы и методы прототипирования	0,7	1,3	4,3
Эвристические методы	1,4	1,3	8,6
Инструменты программной инженерии			
Инструменты работы с требованиями	2,1	5,2	12,8
Инструменты проектирования и конструирования	2,8	6,5	17,1
Инструменты и методы тестирования и сопровождения ПО	2,8	5,2	17,1
Инструменты верификации и обеспечения качества	2,8	5,2	17,1
Инструменты конфигурационного управления	1,4	1,3	8,6

4.2. Распределение контактной и самостоятельной работы по подразделам
Заочная форма обучения

Разделы, подразделы дисциплины	Контактная работа		СР
	лекции	ПЗ	
Методы программной инженерии			
Формальные методы и методы прототипирования	0,5	0,6	5,2
Эвристические методы	1,0	0,6	10,4
Инструменты программной инженерии			
Инструменты работы с требованиями	1,5	2,4	15,5
Инструменты проектирования и конструирования	2,0	3,0	20,7
Инструменты и методы тестирования и сопровождения ПО	2,0	2,4	20,7
Инструменты верификации и обеспечения качества	2,0	2,4	20,7
Инструменты конфигурационного управления	1,0	0,6	10,4

4.3. Перечень тем и учебно-методического обеспечения для самостоятельной работы обучающихся

Разделы, подразделы дисциплины	Учебно-методическое обеспечение	Объем часов СР	
		очная	заочная
Методы программной инженерии			
Формальные методы и методы прототипирования	Волк В. К. Практическое введение в программную инженерию [Электронный ресурс] / В. К. Волк - Санкт-Петербург: Лань, 2022 - 100 с. [ЭИ] [ЭБС Лань] URL: https://e.lanbook.com/book/249848	4,3	5,2
Эвристические методы	Волк В. К. Практическое введение в программную инженерию [Электронный ресурс] / В. К. Волк - Санкт-Петербург: Лань, 2022 - 100 с. [ЭИ] [ЭБС Лань] URL: https://e.lanbook.com/book/249848	8,6	10,4
Инструменты программной инженерии			
Инструменты работы с требованиями	Маран М. М. Программная инженерия [Электронный ресурс] / М. М. Маран - Санкт-Петербург: Лань, 2022 - 196 с. [ЭИ] [ЭБС Лань] URL: https://e.lanbook.com/book/189470	12,8	15,5
Инструменты проектирования и конструирования	Маран М. М. Программная инженерия [Электронный ресурс] / М. М. Маран - Санкт-Петербург: Лань, 2022 - 196 с. [ЭИ] [ЭБС Лань] URL: https://e.lanbook.com/book/189470	17,1	20,7
Инструменты и методы тестирования и сопровождения ПО	Маран М. М. Программная инженерия [Электронный ресурс] / М. М. Маран - Санкт-Петербург: Лань, 2022 - 196 с. [ЭИ] [ЭБС Лань] URL: https://e.lanbook.com/book/189470 Косицин Д. Ю. Язык программирования Python [Электронный ресурс] / Д. Ю. Косицин - Минск: БГУ, 2019 - 136 с. [ЭИ] [ЭБС Лань] URL: https://e.lanbook.com/book/180546	17,1	20,7
Инструменты верификации и обеспечения качества	Антипов В. А. Введение в программную инженерию [электронный ресурс]: Учебник / В. А. Антипов, А. А. Бубнов, А. Н. Пылькин, В. К. Столчнев - Москва: ООО "КУРС", 2019 - 336 с. [ЭИ] [ЭБС Знаниум] URL: https://znanium.com/catalog/document?id=342955	17,1	20,7
Инструменты конфигурационного управления	Антипов В. А. Введение в программную инженерию [электронный ресурс]: Учебник / В. А. Антипов, А. А. Бубнов, А. Н. Пылькин, В. К. Столчнев - Москва: ООО "КУРС", 2019 - 336 с. [ЭИ] [ЭБС Знаниум] URL: https://znanium.com/catalog/document?id=342955 Полупанов Д. В. Программирование в Python 3 [Электронный ресурс]: учебное пособие / Д. В. Полупанов, С. Р. Абдошева, А. М. Ефимов - Уфа: БашГУ, 2020 - 164 с. [ЭИ] [ЭБС Лань] URL: https://e.lanbook.com/book/179915	8,6	10,4
Итого		85,5	103,5

5. Фонд оценочных средств

5.1. Этапы формирования компетенций

Разделы, подразделы дисциплины	Компетенции и ИД			
	ПК-01	ПК-02	ПК-07	ПК-08
Методы программной инженерии				
Формальные методы и методы прототипирования	31			
Эвристические методы	31			
Инструменты программной инженерии				
Инструменты работы с требованиями	31, У1, Н1			
Инструменты проектирования	31, У1, Н1	31, У1, Н1		
Инструменты и методы тестирования и сопровождения ПО	31			31, У1, Н1
Инструменты верификации и обеспечения качества			32, У2, Н2	
Инструменты конфигурационного управления			39, У9, Н10	

5.2. Шкалы и критерии оценивания достижения компетенций

5.2.1. Шкалы академических оценок освоения дисциплины

Вид оценки	Оценки			
Академическая оценка по 4-х балльной шкале	неудовлетворительно	удовлетворительно	хорошо	отлично

Вид оценки	Оценки	
Академическая оценка по 2-х балльной шкале	не зачетно	зачтено

5.2.2. Критерии достижения компетенций в ходе освоения дисциплины

Критерии оценки на экзамене

Оценка, уровень достижения	Описание критериев
Отлично, высокий	Студент показал полные и глубокие знания материала, логично и аргументировано ответил на все вопросы экзаменационного билета, а также на дополнительные вопросы, способен самостоятельно решать сложные задачи дисциплины
Хорошо, продвинутый	Студент твердо знает материал, грамотно его излагает, не допускает существенных неточностей в ответе, достаточно полно ответил на вопросы экзаменационного билета и дополнительные вопросы, способен самостоятельно решать стандартные задачи дисциплины
Удовлетворительно, пороговый	Студент показал знание только основ материала, усвоил его поверхностно, но не допускал грубых ошибок или неточностей, требует наводящих вопросов для правильного ответа, не ответил на дополнительные вопросы, способен решать стандартные задачи дисциплины с помощью преподавателя
Неудовлетворительно, компетенции не освоены	Студент не знает основ материала, допускает грубые ошибки в ответе, не способен решать стандартные задачи дисциплины даже с помощью преподавателя

5.3. Материалы для оценки достижения компетенций

5.3.1. Вопросы к экзамену

№	Содержание	Компетенция	ИД
1	Методы программной инженерии: понятие и классификация	ПК-01	31
2	Характеристика формальных методов программной инженерии	ПК-01	31
3	Характеристика методов прототипирования программной инженерии	ПК-01	31
4	Характеристика эвристических методов программной инженерии	ПК-01	31
5	Требования к ПО: понятие, свойства и виды требований	ПК-01	31
6	Инструменты моделирования требований	ПК-01	31
7	Управление требованиями: понятие, методы и программные средства	ПК-01	31
8	Характеристика методология sadt idef0	ПК-01	31
9	Характеристика ER-моделей	ПК-01	31
10	Характеристика диаграмм в языке UML	ПК-01	31
11	Классификация средств визуального моделирования	ПК-01	31
12	Пользовательский интерфейс: понятие, стили, этапы создания	ПК-02	31
13	Инструменты прототипирования пользовательского интерфейса	ПК-02	31
14	Инструменты конструирования: классификация, примеры	ПК-01	31
15	Классификация методов тестирования	ПК-08	31
16	Классификация инструментов тестирования согласно SWEBOK	ПК-08	31
17	Характеристик техник тестирования согласно SWEBOK	ПК-08	31
18	Генерация тестов.понятие, способы, примеры генераторов	ПК-08	31
19	Инструменты анализа производительности тестов	ПК-08	31
20	Характеристика программ автоматизации тестирования	ПК-08	31
21	Характеристика программ управления тестированием	ПК-08	31
22	Метрики оценки работ по сопровождению	ПК-01	31
23	Инструменты сопровождения согласно SWEBOK	ПК-01	31
24	Реинжиниринг и рефакторинг ПО	ПК-01	31
25	Классификация программ обнаружения ошибок	ПК-01	31
26	Классификация инструментов конфигурационного управления согласно SWEBOK	ПК-07	39
27	Системы управления версиями: понятие и состав	ПК-07	39
28	Классификация систем управления версиями	ПК-07	39
29	Автоматизация сборки программного обеспечения: понятие, содержание, программы	ПК-07	39
30	Аспекты и уровни качества ПО	ПК-07	32
31	Основные характеристики качества ПО	ПК-07	32
32	Основные характеристики надежности ПО	ПК-07	32
33	Инструменты обеспечения качества согласно SWEBOK	ПК-07	32
34	Аналитические динамические модели надежности ПО	ПК-07	32
35	Аналитические статистически модели надежности ПО	ПК-07	32
36	Эмпирические модели надежности ПО	ПК-07	32
37	Верификация ПО: понятие и классификация методов	ПК-07	32
38	Формальные методам верификации ПО	ПК-07	32
39	Динамические методы верификации ПО	ПК-07	32
40	Синтетическим методы верификации ПО	ПК-07	32

5.3.2. Задания к экзамену

№	Содержание	Компетенция	ИД
1	Сформулируйте функциональные требования к ПО заданной предметной области	ПК-01	У1
2	Определите требования к ИС заданной предметной области методом А.Джекобсона	ПК-01	Н1
3	Создайте диаграммы классов на языке UML согласно предметной области	ПК-01	У1
4	Опишите с помощью языка UML предметную область по базе данных заказов на продажу продукции	ПК-01	Н1
5	Создайте прототипа пользовательского интерфейса заданной предметной области в MS PowerPoint	ПК-02	У1
6	Оцените полноту схемы описания информационного взаимодействия компонентов ИС	ПК-02	Н1
7	Оцените параметры надежности представленной ИС по заданным метрикам	ПК-07	У2
8	Разработайте систему показателей и метрик качества для ИС заданной предметной области	ПК-07	Н2
9	Протестируйте заданное ПО на соответствие представленным требованиям заказчика.	ПК-08	У1
10	Провести тестирования представленной блок-схемы, по необходимости внести изменения	ПК-08	Н1
11	Определите на основе данных среды разработки ПО перечень конфигураций и оформите в виде списка	ПК-07	У9
12	Создайте новую конфигурацию проекта на основе представленных данных	ПК-07	Н10

5.3.3. Вопросы к зачету с оценкой

Не предусмотрено

5.3.4. Вопросы к зачету

Не предусмотрено

5.3.5. Темы курсового проект (работы) и вопросы к защите

Не предусмотрено

5.3. Материалы для оценки достижения компетенций

5.3.6. Вопросы тестов

№	Содержание	Компетенция	ИД
1	Методы ПИ согласно SWEBOK делятся на группы: 1. формальные 2. эвристические 3. прототипирования 4. эволюционные 5. прогностические	ПК-01	31
2	Последовательность предписаний или процедур обработки информации, выполняемая с целью поиска более рациональных и новых конструктивных решений называются: 1. эвристическими методами 2. методами прототипирования 3. эволюционными методами	ПК-01	31
3	К эвристическим методам ПИ относят: 1. структурные 2. ориентированные на данные 3. объектно-ориентированные 4. ориентированные на область применения 5. формальные	ПК-01	31
4	Построение программных систем, начиная с высокоуровневого понимания поведения системы с постепенным уточнением низко-уровневых деталей, используется при использовании: 1. структурного метода 2. формального метода 3. метода прототипирования	ПК-01	31
5	Подход, при котором основой являются структуры данных, которыми манипулирует создаваемое программное обеспечение при вторичной роли функций, называется методами: 1. ориентированными на данные 2. ориентированными на область применения 3. ориентированными на требования заказчика	ПК-01	31
6	Если система рассматривается как коллекция объектов, а не функций, то такой метод называется: 1. объектно-ориентированным 2. объектно-зависимым 3. объектно-независимым	ПК-01	31
7	К базовым принципам объектно-ориентированного подхода программирования относят: 1. полиморфизм 2. инкапсуляция 3. наследование 4. деструктуризация 5. глобализация	ПК-01	31
8	Свойство языка программирования, позволяющее объединить данные и код в объект и скрыть такое объединение от пользователя, называется: 1. инкапсуляция 2. полиморфизм 3. наследование	ПК-01	31
9	Взаимозаменяемость объектов с одинаковым интерфейсом называется: 1. инкапсуляция 2. полиморфизм 3. наследование	ПК-01	31
10	Данный принцип позволяет описать новый класс на основе уже существующего, приобретая его свойства и функциональность: 1. инкапсуляция 2. полиморфизм 3. наследование 4. поглощение	ПК-01	31
11	Множество объектов, обладающих одинаковыми свойствами, операциями, отношениями и семантикой, в объектно-ориентированной методологии называется: 1. класс 2. кластер 3. уровень	ПК-01	31
12	Абстрактный образ с поведением, которое обусловлено его характеристиками и взаимоотношениями с другими объектами, в объектно-ориентированной методологии называется: 1. объект 2. класс 3. атрибут	ПК-01	31

13	К Agile-технологиями относятся: 1. Scram-технологии 2. XP-технологии 3. RUP-технологии	ПК-01	31
14	Методы, разрабатываемые с учетом специфики решаемых задач, называются: 1. ориентированными на область применения 2. специфическими 3. прикладными	ПК-01	31
15	Методы, подразумевающие создание математического описания системы, его анализ и доказательство, называются: 1. моделируемыми 2. формальными 3. аналитическими	ПК-01	31
16	Какие из перечисленные методов относятся к формальным: 1. VDM 2. RAISE 3. XP	ПК-01	31
17	Методы с созданием моделей разрабатываемой системы на основе уже существующих систем называются: 1. прототипирования 2. тиражирования 3. копирования	ПК-01	31
18	Макет, пробная версии программы, с целью проверки пригодности предлагаемых для применения концепций, архитектурных и/или технологических решений, называется: 1. прототипом 2. релизом 3. образцом	ПК-01	31
19	К основным типам прототипирования ПО относят: 1. быстрое 2. эволюционное 3. массовое 4. повторяемое	ПК-01	31
20	Прототип, который на каком-то этапе будет оставлен и не станет частью готовой системы, применяется при: 1. быстром прототипировании 2. эволюционном прототипировании 3. «черновом» прототипировании	ПК-01	31
21	Прототипирование, при котором последовательно создаются макеты системы, которые будут все ближе и ближе к реальному продукту, называется: 1. эволюционным 2. реальным 3. полноценным	ПК-01	31
22	Свойства программного обеспечения, которые должны быть надлежащим образом представлены в нём для решения конкретных практических задач, называются: 1. требованиями 2. условиями программирования 3. характеристики программы	ПК-01	31
23	Атомарность требований обозначает: 1. невозможность разделить требование на более мелкие требования 2. комплексный характер используемых требований 3. многослойность требований в сложных системах	ПК-01	31
24	Единичность требований обозначает: 1. требование описывает одну характеристику или единицу 2. требования не связаны между собой 3. требование должно быть одно на всю разрабатываемую систему	ПК-01	31
25	Если требование полностью определено в одном месте и вся необходимая информация присутствует, то такое требование: 1. завершенное 2. абсолютное 3. полноценное	ПК-01	31
26	Требования к информационной системе, отвечающие на вопрос «Что система должна делать?», называются требованиями.	ПК-01	31
27	Требования к информационной системе, отвечающие на вопрос «С соблюдением каких условий система должна работать?», называются требованиями.	ПК-01	31
28	Вариант использования, где описываются все действия, которые пользователь может произвести, и реакцию системы на эти действия, называется: 1. Use Case 2. User Story 3. User Variable	ПК-01	31

29	User Story - это: 1. описания требований пользователя в краткой форме 2. история работы пользователя с системой 3. пользовательский интерфейс пользователя	ПК-01	31
30	Анализ влияния изменений в требованиях на компоненты системы и другие требования называется: 1. трассированием требований 2. валидацией требований 3. верификацией требований	ПК-01	У1
31	Проверка требований системы на соответствие пожеланиям заказчика называется: 1. валидацией требований 2. трассированием требований 3. верификацией требований	ПК-01	У1
32	Проверка требований на соответствие правилам, оформленным в документации, называется: 1. Верификацией требований 2. Трассированием требований 3. Валидацией требований	ПК-01	У1
33	Процесс формализованного описания функциональных и нефункциональных требований, требований к характеристикам качества, называется: 1. спецификация 2. адаптация 3. моделирование	ПК-01	У1
34	Инструменты работы с требованиями по SWEBOK разделяются на: 1. средства моделирования (управления) 2. средства компилирования 3. средства трассирования 4. средства сопровождения	ПК-01	Н1
35	Метод инженерии требований А.Джекобсона является: 1. методом с последовательным выявлением объектов, существенных для домена проблемной области 2. метод в параллельном выявлении объектов, существенных для домена проблемной области 3. метод с математическим выявлением объектов, существенных для домена проблемной области	ПК-01	Н1
36	Инструменты моделирования требований предназначены для: 1. для извлечения, анализа, специфицирования и проверки программных требований 2. для тестирования и выявления ошибок в программных требованиях 3. для формализации программных требований	ПК-01	Н1
37	К специализированным программам управления требованиями относят: 1. IBM Rational DOORS; 2. Visure 3. Micro Focus Dimensions RM 4. Microsoft Visio 5. Git	ПК-01	Н1
38	Концептуальная модель предметной области это: 1. модель без ориентации на программные и технические средства 2. модель структуры программных модулей системы 3. модель технических средств реализации модели	ПК-01	31
39	При данном виде проектирования формируются спецификации алгоритмов задач, построении баз данных и программного обеспечения системы: 1. концептуальном 2. архитектурном 3. техническом 4. детальном рабочем	ПК-01	31
40	Визуальное моделирование это: 1. способ создания программы путём манипулирования графическими объектами 2. способ создания программы путем выделения разными цветами объектов программирования 3. способ формализации требований к программе в табличном виде	ПК-01	31
41	К языкам визуального моделирования относят: 1. UML 2. BPMN 3. SADT/IDEF0 4. C+	ПК-01	31
42	Для визуального моделирования бизнес-процессов используют язык: 1. SADT/IDEF0 2. UML 3. BPMN 4. DFD	ПК-01	31

43	Для проведения структурного анализа используются диаграммы: 1. SADT 2. ERD 3. UML 4. DFD 5. BPMN	ПК-01	31
44	Множество иерархически взаимосвязанных диаграмм, которые представляют сложную систему отдельными ее составными частями, называется: 1. моделью SADT 2. моделью ERD 3. моделью DFD	ПК-01	31
45	Основными компонентами ER-диаграмм являются: 1. сущность 2. связь 3. класс объектов 4. поток информации	ПК-01	31
46	Язык UML состоит из: 1. семантика языка 2. графические нотации 3. тесты 4. спецификации	ПК-01	31
47	Назовите три основных типа диаграмм языка UML: 1. структурные 2. поведения 3. взаимодействия 4. программирования 5. тестирования	ПК-01	У1
48	К структурным диаграммам языка UML относят: 1. диаграммы пакетов 2. диаграммы классов 3. диаграммы компонентов 4. диаграммы развертывания 5. диаграммы кооперации	ПК-01	У1
49	К диаграммам взаимодействия языка UML относят: 1. диаграммы кооперации 2. диаграммы последовательности 3. диаграммы классов 4. диаграммы состояния	ПК-01	У1
50	К диаграммам поведения языка UML относят: 1. диаграммы вариантов использования 2. диаграммы состояния 3. диаграммы деятельности 4. диаграммы классов 5. диаграммы последовательности	ПК-01	У1
51	Данная диаграмма в языке UML показывает, как ПО размещается на элементах аппаратного комплекса: 1. диаграмма развертывания 2. диаграммы состояния 3. диаграммы кооперации	ПК-01	У1
52	Данная диаграмма в языке UML показывает работу системы с точки зрения пользователей: 1. диаграмма вариантов использования 2. диаграммы классов 3. диаграммы состояния 4. диаграммы кооперации	ПК-01	У1
53	Какие приложения предназначены для создания SADT-диаграмм: 1. Design/IDEF 2. Dia 3. Visio 4. Lucidchart	ПК-01	Н1
54	Какие приложения предназначены для создания UML-диаграмм: 1. Visual Paradigm 2. Diagrams.net 3. Umler 4. Erwin	ПК-01	Н1

55	В каких программах можно создать ER-диаграммы: 1. Lucidchart 2. Gliffy 3. Diagrams.net 4. yUML	ПК-01	Н1
56	Элементы и компоненты программы, которые способны оказывать влияние на взаимодействие пользователя с программным обеспечением, называются: 1. интерфейсом пользователя 2. интегрированная среда разработки 3. аккаунт пользователя	ПК-02	31
57	Пользовательский интерфейс может быть: 1. командной строки 2. графический 3. речевой 4. биометрический 5. семантический 6. мультимедийный	ПК-02	31
58	Согласно ГОСТ Р ИСО 9241-161-2016, составляющими интерфейса пользователя являются: 1. интерактивная 2. информационная 3. декоративная 4. управляющая	ПК-02	31
59	Конструкторы пользовательского интерфейса делятся на: 1. конструкторы интегрированных сред разработки ПО 2. автономные конструкторы 3. конструкторы систем тестирования	ПК-02	У1
60	Какие из перечисленных программ являются автономными конструкторами интерфейсов: 1. NetBeans 2. Visual Studio 3. Glade	ПК-02	У1
61	Конструкторы пользовательского интерфейса позволяют осуществлять: 1. добавление, удаление, размещение компонентов на экране 2. редактирование свойств компонентов 3. генерация исходного кода инициализации компонентов 4. редактирование взаимосвязей между модулями системы	ПК-02	У1
62	Согласно SWEBOOK, к инструментам конструирования относят: 1. редакторы 2. компиляторы 3. интерпретаторы 4. отладчики 5. тестировщики	ПК-01	31
63	Для создания и модификации программ используются: 1. редакторы 2. компиляторы 3. интерпретаторы 4. отладчики	ПК-01	31
64	Какие из представленных относятся к редакторам IDE: 1. редактор Microsoft Visual Studio 2. редактор PyCharm 3. Редактор Sublime Text 3 4. Редактор Atom	ПК-01	31
65	IDE среды включают: 1. текстовый редактор 2. транслятор 3. средства автоматизации сборки 4. отладчик 5. багтрекинг-система	ПК-01	31
66	К IDE средам относят: 1. IntelliJ 2. CLion 3. PyCharm 4. Microsoft Visual Studio 5. Jira	ПК-01	31
67	К специализированным текстовым редакторам кода относят: 1. Notepad++ 2. Sublime Text 3 3. Atom 4. MS Excel 5. Make	ПК-01	31

68	<p>Данный транслятор полностью обрабатывают весь текст программы, прежде чем запускать ее на исполнение:</p> <ol style="list-style-type: none"> 1. компилятор 2. редактор связей 3. генератор кода 	ПК-01	31
69	<p>Данный транслятор берет очередной оператор языка из текста программы, анализирует его структуру и затем сразу исполняет, только после того как текущий оператор успешно выполнен, он перейдет к следующему:</p> <ol style="list-style-type: none"> 1. интерпретатор 2. редактор связей 3. генератор кода 	ПК-01	31
70	<p>Результатом компиляции является:</p> <ol style="list-style-type: none"> 1. объектный код 2. исполнимый файл 3. библиотека файлов 	ПК-01	31
71	<p>Кросс-компиляторы это:</p> <ol style="list-style-type: none"> 1. компилятор, который создает программу, предназначенную запускаться на другой системе 2. компилятор, который способен компилировать код, написанный на разных языках программирования 3. компилятор, который способен максимально быстро компилировать код 	ПК-01	31
72	<p>Программа, транслирующая исполняемый модуль в относительно эквивалентный исходный код на языке программирования высокого уровня, называется:</p> <ol style="list-style-type: none"> 1. декомпилятором 2. интерпретатором 3. багтрекингом 	ПК-01	31
73	<p>Выявление и устранение неправильной работы программы называется:</p> <ol style="list-style-type: none"> 1. отладкой 2. валидацией 3. трассировкой 	ПК-01	31
74	<p>Под запуском программы на разных наборах исходных данных и анализом полученных результатов с целью обнаружения ошибок понимается</p>	ПК-08	31
75	<p>Укажите уровни функционального тестирования (выберите три правильных варианта ответа):</p> <ol style="list-style-type: none"> 1. дымовое 2. критического пути 3. расширенное 4. неограниченное 	ПК-08	31
76	<p>Под отклонением фактического результата от ожидаемого в программе понимается.....</p>	ПК-08	31
77	<p>Документ, описывающий весь объем работ по тестированию, начиная с описания объекта, стратегии, расписания, критериев начала и окончания тестирования, до необходимого в процессе работы оборудования, специальных знаний, а также оценки рисков с вариантами их разрешения называется тест-.....</p>	ПК-08	31
78	<p>Атрибут, характеризующий влияние бага на общую функциональность тестируемого продукта, называется бага (русскими буквами)</p>	ПК-08	31
79	<p>Bug report - это:</p> <ol style="list-style-type: none"> 1. документ с отчетом о недостатках в программе 2. скомпилированный файл 3. план тестирования ПО 	ПК-08	31
80	<p>Набор входных данных, условий выполнения и ожидаемых результатов, разработанный с целью проверки того или иного свойства или поведения программного средства называется тест-....</p>	ПК-08	31
81	<p>К методам тестирования «черного ящика» относят (выберите три правильных варианта ответа):</p> <ol style="list-style-type: none"> 1. метод эквивалентного разбиения 2. метод граничных значений 3. метод функциональных диаграмм 4. метод Делфи 	ПК-08	31
82	<p>К этапам тестирования относят (выберите четыре правильных варианта ответа):</p> <ol style="list-style-type: none"> 1. планирование тестов 2. генерация сценариев тестирования и разработка тестового окружения 3. выполнение тестов 4. анализ результатов тестирования 5. редактирование кода 	ПК-08	31

83	<p>К тестовым работам относят (выберите три правильных варианта ответа):</p> <ol style="list-style-type: none"> 1. генерация сценариев тестирования 2. формализация требований 3. выполнение тестов 4. анализ результатов тестирования 5. проектирование предметной области 	ПК-08	31
84	<p>Баг, не влияющий на работу программы, является:</p> <ol style="list-style-type: none"> 1. Trivial (Тривиальным) 2. Minor (Незначительным) 3. Major (Серьезным) 	ПК-08	31
85	<p>Баг, при котором программа не работает в принципе, является:</p> <ol style="list-style-type: none"> 1. Major (Серьезный) 2. Critical (Критический) 3. Blocker (Блокирующий) 	ПК-08	31
86	<p>Баг, который мешает приложению выполнять основные функции, является</p> <ol style="list-style-type: none"> 1. Major (Серьезный) 2. Critical (Критический) 3. Blocker (Блокирующий) 	ПК-08	31
87	<p>Какой из статусов не относится к жизненному циклу бага?</p> <ol style="list-style-type: none"> 1. Открыт 2. В работе 3. Исправлен 4. Закрыт 5. Уничтожен 	ПК-08	31
88	<p>К атрибутам отчета о дефекте относят (выберите три правильных варианта ответа):</p> <ol style="list-style-type: none"> 1. описание дефекта 2. фактический результат 3. ожидаемый результат 4. анализ требований 	ПК-08	У1
89	<p>В состав тест-кейса входит (выберите три правильных варианта ответа):</p> <ol style="list-style-type: none"> 1. шаги тестирования 2. ожидаемые результаты 3. фактические результаты 4. экономический эффект 	ПК-08	У1
90	<p>Техника тестирования поведения продукта на крайних (граничных) значениях входных данных, называется:</p> <ol style="list-style-type: none"> 1. техника эквивалентного разбиения 2. техника граничных значений 3. техника сценариев использования 	ПК-08	У1
91	<p>Техника формирования наборов тестовых данных, при которой каждое тестируемое значение каждого из проверяемых параметров хотя бы раз сочетается с каждым из тестируемых значений всех остальных проверяемых параметров, называется:</p> <ol style="list-style-type: none"> 1. техника попарного тестирования 2. техника параллельного тестирования 3. техника пересеченного тестирования 	ПК-08	У1
92	<p>Техника тестирования, при которой требуется написание такого количества тестов, чтобы при выполнении их всех каждый оператор был выполнен хотя бы один раз, называется:</p> <ol style="list-style-type: none"> 1. техника покрытия операторов 2. техника попарного тестирования 3. техника граничных значений 	ПК-08	У1
93	<p>Тестирование, основанное на опыте, при котором тестер спонтанно разрабатывает и выполняет тестирования на основе существующих соответствующих знаний тестера, называется:</p> <ol style="list-style-type: none"> 1. исследовательским 2. эмпирическим 3. интуитивным 	ПК-08	У1
94	<p>Функциональные техники тестирования строятся на основе:</p> <ol style="list-style-type: none"> 1. спецификации 2. кода 3. дефектов 	ПК-08	У1
95	<p>Структурные техники тестирования строятся на основе:</p> <ol style="list-style-type: none"> 1. спецификации 2. кода 3. дефектов 	ПК-08	У1
96	<p>К инструментам тестирования относят (выберите два правильных ответа):</p> <ol style="list-style-type: none"> 1. генераторы тестовых данных 2. компиляторы кода 3. средства управления тестированием 4. средства анализа требований 	ПК-08	Н1

97	Специализированный программный инструмент, который генерирует ложные или фиктивные данные для использования при тестировании программных приложений называется тестовых данных.	ПК-08	Н1
98	Эти средства обеспечивают среду исполнения тестовых сценариев в контролируемом окружении, позволяющем отслеживать поведение объекта, подвергаемого тестированию: 1. генераторы тестов 2. средства выполнения тестов 3. средства оценки тестов	ПК-08	Н1
99	Системы управления тестированием предназначены для: 1. обеспечения поддержки всех аспектов тестирования 2. количественной и качественной оценки тестирования 3. формирования итоговых документов тестирования	ПК-08	Н1
100	К программам управления тестам относятся: 1. TestRail 2. «Кайман» 3. Zephyr 4. MicrosoftTest	ПК-08	Н1
101	К генераторам тестовых данных относят (выберите два правильных варианта ответа): 1. Test Data Manager 2. Mockaroo.com 3. Visual Studio 4. CVS	ПК-08	Н1
102	Под TMS в программной инженерии понимается система управления	ПК-08	Н1
103	К функциям системы управления тестированием относят (выберите три правильных варианта ответа): 1. управление тестовыми сценариями 2. управление автоматизированным тестированием 3. управление конфигурацией 4. поддержка версионности тестовых сценариев 5. изменение требований к системе	ПК-08	Н1
104	Модификацию программного продукта после передачи в эксплуатацию для устранения сбоев, улучшения его характеристик адаптацию для использования в модифицированном окружении называется: 1. сопровождением 2. отладкой 3. тестированием	ПК-08	31
105	Сопровождение в виде модификации программного продукта, выполняемое уже после передачи в эксплуатацию для устранения сбоев называется: 1. корректирующее 2. адаптирующее 3. совершенствующее 4. профилактическое	ПК-08	31
106	Сопровождение в виде модификации программного продукта на этапе эксплуатации для обеспечения продолжения его использования с заданной эффективностью в изменившемся или находящемся в процессе изменения окружении, называется: 1. корректирующее 2. адаптирующее 3. совершенствующее 4. профилактическое	ПК-08	31
107	Сопровождение в виде модификации программного продукта на этапе эксплуатации для повышения характеристик производительности и удобства сопровождения, называется: 1. корректирующее 2. адаптирующее 3. совершенствующее 4. профилактическое	ПК-08	31
108	Сопровождение в виде модификации программного продукта программного продукта на этапе эксплуатации для идентификации и предотвращения скрытых дефектов до того, когда они приведут к реальным сбоям, называется: 1. корректирующее 2. адаптирующее 3. совершенствующее 4. профилактическое	ПК-08	31
109	Детальная оценка и перестройка программного обеспечения для формирования понимания, воссоздания и дальнейшей реализации его функций в новой форме, называется: 1. реинжиниринг 2. рефакторинг 3. аутсорсинг	ПК-08	31

110	<p>Процесс получения нового компонента на базе существующего, включает операции модификации (изменение, замещение, расширение) компонентов и интерфейсов, называется:</p> <ol style="list-style-type: none"> 1. реинжиниринг 2. рефакторинг 3. рецессинг 	ПК-08	31
111	<p>Восстановление спецификации по полученному коду системы для наблюдения за ней на более высоком уровне, восстановление идентификации программных компонентов и связей между ними для обеспечения перепрограммирования системы к новой форме, называется:</p> <ol style="list-style-type: none"> 1. реверсная инженерия 2. реинжиниринг 3. рефакторинг 4. реверсная адаптация 	ПК-08	31
112	<p>Аспекты качества в программной инженерии делятся на:</p> <ol style="list-style-type: none"> 1. качество процесса 2. качество продукта 3. качество сопровождения 4. качество персонала 	ПК-07	32
113	<p>Набор свойств продукта, которые характеризуют его способность удовлетворить установленные или предполагаемые потребности заказчика, называется:</p> <ol style="list-style-type: none"> 1. условия системы 2. качество системы 3. уровень системы 	ПК-07	32
114	<p>К основным характеристикам качества ПО относят следующие характеристики:</p> <ol style="list-style-type: none"> 1. функциональность 2. надежность 3. удобство 4. эффективность 5. сопровождаемость 6. переносимость 7. минимальность 	ПК-07	32
115	<p>Совокупность свойств, определяющих способность ПО выполнять в заданной среде перечень функций в соответствии с требованиями к обработке и общесистемным средствам, называется:</p> <ol style="list-style-type: none"> 1. функциональность 2. удобство 3. эффективность 4. переносимость 	ПК-07	32
116	<p>Совокупность атрибутов, которые определяют способность ПО преобразовывать исходные данные в результаты при условиях, зависящих от периода времени жизни, называется:</p> <ol style="list-style-type: none"> 1. функциональность 2. надежность 3. удобство 4. эффективность 5. переносимость 	ПК-07	32
117	<p>Атрибуты, которые определяют взаимосвязь между уровнем выполнения ПО, количеством используемых ресурсов и услуг, выполняемых штатным обслуживающим персоналом, называются:</p> <ol style="list-style-type: none"> 1. функциональность 2. удобство 3. эффективность 4. переносимость 	ПК-07	32
118	<p>Множество показателей, указывающих на способность ПО приспосабливаться к работе в новых условиях среды выполнения, называются:</p> <ol style="list-style-type: none"> 1. функциональность 2. удобство 3. эффективность 4. переносимость 	ПК-07	32
119	<p>К инструментам обеспечения качества согласно SWEBOK относят:</p> <ol style="list-style-type: none"> 1. инструменты инспектирования 2. инструменты анализа 3. инструменты кодирования 	ПК-07	32
120	<p>К внутренним метрикам ПО относят:</p> <ol style="list-style-type: none"> 1. свойства продукта, видимые пользователю 2. свойства продукта, видимые только команде разработчиков 3. свойства продукта, видимые только руководителю программного проекта 	ПК-07	32

121	<p>К внешним метрикам ПО относят:</p> <ol style="list-style-type: none"> 1. свойства продукта, видимые пользователю 2. свойства продукта, видимые только команде разработчиков 3. свойства продукта, видимые только руководителю программного проекта 	ПК-07	32
122	<p>К внутренним метрикам ПО относят:</p> <ol style="list-style-type: none"> 1. надежность 2. функциональность 3. размер 4. сложность 	ПК-07	32
123	<p>К внешним метрикам ПО относят:</p> <ol style="list-style-type: none"> 1. надежность 2. функциональность 3. размер 4. сложность 	ПК-07	32
124	<p>Для оценки значений показателей качества используются методы:</p> <ol style="list-style-type: none"> 1. измерительный 2. регистрационный 3. расчетный 4. экспертный 5. критический 	ПК-07	32

125	Согласно стандарту ISO/IES 9126-2, для оценки качества ПО применяются 5 видов шкал: 1. номинальная 2. порядковая 3. интервальная 4. относительная 5. абсолютная 6. параллельная	ПК-07	32
126	Совокупность организационной структуры и ответственных лиц, а также процедур, процессов и ресурсов для планирования и управления достижением качества ПС, называется: 1. управление качеством 2. контроль качества 3. аудит качества	ПК-07	32
127	Проверка того, правильно ли система работает в соответствии с ее спецификацией и заданными требованиями, называется: 1. верификация 2. валидация 3. трассировка	ПК-07	32
128	Метод проверки соответствия спроектированного ПО требованиям и потребностям заказчика называется: 1. верификация 2. валидация 3. трассирование	ПК-07	32
129	Математические модели надежности делятся на: 1. аналитические 2. эмпирические 3. поведенческие	ПК-07	У2
130	Аналитические модели надежности разделяются: 1. статические 2. динамические 3. регрессионные	ПК-07	У2
131	К аналитическим динамическим дискретным моделям надежности относят: 1. модель Шумана 2. модель Ла Падула 3. модель Мусса	ПК-07	У2
132	К аналитическим динамическим непрерывным моделям надежности относят: 1. модель Джелинского-Моранды 2. модель Шика-Волвертона 3. модель Мусса 4. модель переходных вероятностей 5. модель Шумана	ПК-07	У2
133	К аналитическим статистическим моделям надежности относят: 1. модель Миллса 2. модель Липова 3. модель Нельсона 4. модель Шумана	ПК-07	У2
134	Методы верификации ПО разделяют на: 1. экспертиза 2. статический анализ 3. формальные методы 4. динамические 5. синтетические 6. семантические	ПК-07	У2
135	Инструменты конфигурационного управления делятся на категории согласно SWEBOOK: 1. инструменты отслеживания дефектов, расширений и проблем 2. инструменты управления версиями 3. инструменты сборки и выпуска 4. инструменты формирования требований	ПК-07	39
136	Багтрекинг-системы предназначены: 1. выявления, регистрации и отслеживании ошибок в программном обеспечении 2. разработки спецификаций требований после тестирования программного обеспечения 3. оптимизации кода и структуры программного обеспечения	ПК-07	39
137	Дефект (bug)-это: 1. отклонение фактического результата от ожидаемого 2. ошибка синтаксиса в коде 3. неправильно оформленное требование	ПК-07	39
138	Bug herog- это: 1. документ с отчетом о недостатке в программе 2. скомпилированный файл 3. план тестирования ПО	ПК-07	39

139	<p>Под автоматизацией сборки программного обеспечения понимается (выберите четыре правильных варианта ответа):</p> <ol style="list-style-type: none"> 1. компиляция исходного кода в объектный модуль 2. сборка бинарного кода в исполняемый файл 3. развёртывание программы в целевой среде 4. написание сопроводительной документации 5. формализация требований 6. написание кода модулей 	ПК-07	39
140	<p>Специальное программное обеспечение для хранения версий документа и возвращения к более ранним версиям, определения сроков и источника изменения, называется:</p> <ol style="list-style-type: none"> 1. системы управления версиями 2. системы управления конфигурациями 3. система управления сборками 	ПК-07	39
141	<p>По способу хранения системы контроля версиями делятся на (выберите три правильных ответа):</p> <ol style="list-style-type: none"> 1. локальные 2. централизованные 3. распределенные 4. глобальные 	ПК-07	39
142	<p>Если основной репозиторий программного проекта хранится на отдельном сервере, то такая система контроля версий является системой.</p>	ПК-07	39
143	<p>Автоматизация сборки программы может быть (выберите три правильных ответа):</p> <ol style="list-style-type: none"> 1. по запросу 2. запланированная автоматизация 3. условная автоматизация 4. регрессионная 	ПК-07	39
144	<p>Ветвь в системах управления версиями это:</p> <ol style="list-style-type: none"> 1. направление разработки проекта, независимое от других 2. главная версия разрабатываемого проекта 3. неиспользуемый программный код 	ПК-07	У9
145	<p>Commit в системах управления версиями это:</p> <ol style="list-style-type: none"> 1. фиксация изменений 2. выход из системы 3. создание новой версии 	ПК-07	У9
146	<p>Check-out в системах управления версиями это:</p> <ol style="list-style-type: none"> 1. извлечение конкретных версий для редактирования 2. стоимостная оценка конкретной версии 3. закрытие версии для редактирования 	ПК-07	У9
147	<p>Метка (Tag) в системах управления версиями это:</p> <ol style="list-style-type: none"> 1. именованная версия проекта в определенный момент 2. закладка для разработчика на активной версии 3. главная неизменная версия продукта 	ПК-07	У9
148	<p>Какие программы относятся к системам контроля версий (выберите три правильных варианта ответа):</p> <ol style="list-style-type: none"> 1. CVS 2. SVN 3. Git 4. BugZilla 5. Visual Studio Code 	ПК-07	Н10
149	<p>Для автоматизации сборки используется утилита:</p> <ol style="list-style-type: none"> 1. Mak 2. CVS 3. UML 4. Jira 	ПК-07	Н10
150	<p>К багтрекинговым системам относится:</p> <ol style="list-style-type: none"> 1. Visual Studio Code 2. Jira 3. Microsoft Visio 4. Visual Paradigm 	ПК-07	Н10

5.3. Материалы для оценки достижения компетенций

5.3.7. Вопросы для устного опроса

№	Содержание	Компетенция	ИД
1	Что понимается под методологией и методом в программной инженерии?	ПК-01	31
2	Охарактеризуйте компоненты методологии программной инженерии	ПК-01	31
3	Охарактеризуйте содержание методов прототипирования	ПК-01	31
4	Назовите виды эвристических методов программной инженерии	ПК-01	31
5	Назовите основные направления формальных методов программной инженерии	ПК-01	31
6	Дайте определение требованию и назовите его свойства	ПК-01	31
7	Назовите виды требований к ПО	ПК-01	31
8	Охарактеризуйте классификацию инструментов управления требованиями по SWEBOK	ПК-01	31
9	Охарактеризуйте метод инженерии требований А.Джекобса	ПК-01	31
10	Назовите современные инструменты моделирования требований и их назначение	ПК-01	31
11	Назовите и охарактеризуйте функции программ управления требованиями	ПК-01	31
12	Охарактеризуйте методологию sadt idef0	ПК-01	31
13	Назовите специфику и элементы ER-моделей	ПК-01	31
14	Перечислите языки визуального проектирования	ПК-01	31
15	Опишите назначение и виды структурных диаграмма языка UML	ПК-01	31
16	Охарактеризуйте диаграммы поведения языка UML	ПК-01	31
17	Охарактеризуйте диаграммы классов языка UML	ПК-01	31
18	Охарактеризуйте диаграммы компонентов языка UML	ПК-01	31
19	Перечислите средства, реализующие языки и методы визуального моделирования	ПК-01	31
20	Дайте определение пользовательскому интерфейсу, определите основные его виды	ПК-02	31
21	Назовите состав пользовательского графического интерфейса	ПК-02	31
22	Перечислите последовательность разработки пользовательского интерфейса	ПК-02	31
23	Назовите виды прототипирования пользовательского интерфейса	ПК-02	31
24	Назовите инструментам прототипирования пользовательского интерфейса	ПК-02	31
25	Назовите инструменты конструирования согласно SWEBOK	ПК-01	31
26	Какие методы тестирования ПО существуют?	ПК-08	31
27	Назовите виды инструментов тестирования согласно SWEBOK	ПК-08	31
28	Дайте понятие генерации тестов и назовите инструменты для генерации	ПК-08	31
29	Охарактеризуйте назначение и примеры инструментов анализа производительности тестов	ПК-08	31
30	Дайте понятие багтрекинговым системам и охарактеризуйте их виды	ПК-08	31
31	Назовите и охарактеризуйте техники тестирования различают по SWEBOK	ПК-08	31
32	Перечислите инструменты сопровождения согласно SWEBOK	ПК-01	31
33	Дайте определение понятиям реинжиниринга и рефакторинга ПО	ПК-01	31
34	Перечислите инструменты конфигурационного управления согласно SWEBOK	ПК-07	39
35	Назовите состав систем управления версиями	ПК-07	39
36	Перечислите виды систем управления версиями	ПК-07	39
37	Перечислите программы автоматизации сборки ПО, их функционал	ПК-07	39
38	Назовите уровни и аспекты качества ПО	ПК-07	33
39	Определите составляющие качества ПО	ПК-07	33
40	Перечислите характеристики надежности ПО	ПК-07	33
41	Перечислите инструменты обеспечения качества согласно SWEBOK	ПК-07	32
42	Назовите классификацию методов оценки значений показателей качества	ПК-07	32
43	Охарактеризуйте аналитические динамические модели надежности ПО	ПК-07	32
44	Охарактеризуйте аналитические статические модели надежности ПО	ПК-07	32
45	Охарактеризуйте эмпирические модели надежности ПО	ПК-07	32
46	Дайте определение верификации и валидации ПО	ПК-07	32
47	Перечислите методы верификации ПО	ПК-07	32
48	Охарактеризуйте формальные методы верификации ПО	ПК-07	32
49	Охарактеризуйте динамические методы верификации ПО	ПК-07	32
50	Охарактеризуйте синтетические методы верификации ПО	ПК-07	32

5.3.8. Задания для проверки формирования навыков

№	Содержание	Компетенция	ИД
1	Сформулируйте нефункциональные требования к ПО заданной предметной области	ПК-01	У1
2	Сформулируйте требования к ПО заданной предметной области	ПК-01	Н1
3	Создайте диаграммы объектов/компонентов на языке UML согласно предметной области	ПК-01	У1
4	Опишите с помощью языка UML предметную область по заданной базе данных	ПК-01	Н1
5	Создайте бумажный прототип пользовательского интерфейса заданной предметной области	ПК-02	У1
6	Оцените полноту схемы описания информационного взаимодействия компонентов ИС	ПК-02	Н1
7	Оцените параметры качества представленной ИС по заданным метрикам	ПК-07	У2
8	Разработайте систему показателей качества для ИС заданной предметной области	ПК-07	Н2
9	Протестируйте заданное ПО на соответствие представленным требованиям заказчика.	ПК-08	У1
10	Провести тестирования представленной блок-схемы, по необходимости внести изменения	ПК-08	Н1
11	Определите на основе данных среды разработки ПО перечень конфигураций и оформите в виде списка	ПК-07	У9
12	Создайте конфигурацию проекта на основе требований к составу проекта	ПК-07	Н10

5.3.9. Вопросы для контрольной (расчетно-графической) работы

Не предусмотрено

5.4. Система оценивания достижения компетенций

5.4.1. Оценка достижения компетенций в ходе промежуточной аттестации

Индикаторы дотижения компетенций		Номера вопросов и	
Код	Содержание	вопросы к экзамену	задачи к экзамену
ПК-01 Способность проводить обследование организаций, выявлять информационные потребности пользователей, формировать требования к информационной системе			
З1	подходы к разработке, функционированию и сопровождению программного обеспечения	1-11, 14, 22, 25	
У1	выявлять и формулировать требования к структуре и функционалу информационных систем		1,3
Н1	применения инструментов и методов программной инженерии при формировании требований к информационной системе		2,4
ПК-02 Способность разрабатывать и адаптировать прикладное программное обеспечение			
З1	инструменты и методы прототипирования пользовательского интерфейса	12, 13	
У1	применять инструменты и методы прототипирования пользовательского интерфейса		5
Н1	выдачи экспертных заключений по предложенным решениям по реализации интерфейсов и форматов обмена данными		6
ПК-07 Способность настраивать, эксплуатировать и сопровождать информационные системы и сервисы			
З2	инструменты и методы верификации и интеграции ИС	30-40	
З9	функции систем управления конфигурациями	26-29	
У2	оценивать параметры информационных систем		7
У9	документировать изменения конфигураций		11
Н2	разработки системы оценки качества информационных систем		8
Н10	работы со средствами конфигурационного управления		12
ПК-08 Способность проводить тестирование			
З1	инструменты и методы тестирования информационных систем	15-21	
У1	тестировать информационные системы на соответствие параметрам качества		9
Н1	проведения тестирования, обработки и использования их результатов		10

5.4. Система оценивания достижения компетенций
5.4.2. Оценка достижения компетенций в ходе текущего контроля

Индикаторы дотижения компетенций		Номера вопросов и задач		
Код	Содержание	вопросы тестов	вопросы устного опроса	задачи для проверки навыков
ПК-01 Способность проводить обследование организаций, выявлять информационные потребности пользователей, формировать требования к информационной системе				
31	подходы к разработке, функционированию и сопровождению программного обеспечения	1-29, 38-46, 62-73	1-19, 25, 32, 33	
У1	выявлять и формулировать требования к структуре и функционалу информационных систем	30-33,47-52		1,3
Н1	применения инструментов и методов программной инженерии при формировании требований к информационной системе	34-37,53-55		2,4
ПК-02 Способность разрабатывать и адаптировать прикладное программное обеспечение				
31	инструменты и методы прототипирования пользовательского интерфейса	56-58	20-24	
У1	применять инструменты и методы прототипирования пользовательского интерфейса	59-61		5
Н1	выдачи экспертных заключений по предложенным решениям по реализации интерфейсов и форматов обмена данными			6
ПК-07 Способность настраивать, эксплуатировать и сопровождать информационные системы и сервисы				
32	инструменты и методы верификации и интеграции ИС	112-128	38-50	
39	функции систем управления конфигурациями	135-143	34-37	
У2	оценивать параметры информационных систем	129-134		7
У9	документировать изменения конфигураций	144-147		11
Н2	разработки системы оценки качества информационных систем			8
Н10	работы со средствами конфигурационного управления	148-150		12
ПК-08 Способность проводить тестирование				
31	инструменты и методы тестирования информационных систем	74-87,104-111	26-31	
У1	тестировать информационные системы на соответствие параметрам качества	88-95		9
Н1	проведения тестирования, обработки и использования их результатов	96-103		10

6. Учебно-методическое обеспечение дисциплины

6.1. Рекомендуемая литература

№	Библиографическое описание	Вид издания
1	Волк В. К. Практическое введение в программную инженерию [Электронный ресурс] / В. К. Волк - Санкт-Петербург: Лань, 2022 - 100 с. [ЭИ] [ЭБС Лань] URL: https://e.lanbook.com/book/249848	Учебное
2	Маран М. М. Программная инженерия [Электронный ресурс] / М. М. Маран - Санкт-Петербург: Лань, 2022 - 196 с. [ЭИ] [ЭБС Лань] URL: https://e.lanbook.com/book/189470	Учебное
3	Антипов В. А. Введение в программную инженерию [электронный ресурс]: Учебник / В. А. Антипов, А. А. Бубнов, А. Н. Пылькин, В. К. Столчнев - Москва: ООО "КУРС", 2019 - 336 с. [ЭИ] [ЭБС Знаниум] URL: https://znanium.com/catalog/document?id=342955	Учебное
4	Косицин Д. Ю. Язык программирования Python [Электронный ресурс] / Д. Ю. Косицин - Минск: БГУ, 2019 - 136 с. [ЭИ] [ЭБС Лань] URL: https://e.lanbook.com/book/180546	Учебное
5	Полупанов Д. В. Программирование в Python 3 [Электронный ресурс]: учебное пособие / Д. В. Полупанов, С. Р. Абдюшева, А. М. Ефимов - Уфа: БашГУ, 2020 - 164 с. [ЭИ] [ЭБС Лань] URL: https://e.lanbook.com/book/179915	Учебное
6	Улезько А. В. Порядок оценивания результатов достижения компетенций [Электронный ресурс]: методические материалы для основной образовательной программы бакалавриата по направлению: 09.03.03 Прикладная информатика, профиль: Информационные системы и технологии в менеджменте АПК / [А. В. Улезько, С. А. Кулев, А. А. Толстых]; Воронежский государственный аграрный университет - Воронеж: Воронежский государственный аграрный университет, 2019 [ПТ] URL: http://catalog.vsau.ru/elib/metod/m153419.pdf	Методическое
7	Улезько А. В. Порядок формирования компетенций [Электронный ресурс]: методические материалы для основной образовательной программы бакалавриата по направлению 09.03.03 Прикладная информатика, профиль: Информационные системы и технологии в менеджменте АПК / [А. В. Улезько, С. А. Кулев, А. А. Толстых]; Воронежский государственный аграрный университет - Воронеж: Воронежский государственный аграрный университет, 2019 [ПТ] URL: http://catalog.vsau.ru/elib/metod/m153465.pdf	Методическое
8	Программирование: журнал / Учредители: Российская академия наук, Московский государственный университет - Москва: Российская академия наук, 2020 [ЭИ] URL: https://elibrary.ru/contents.asp?titleid=7966	Периодическое
9	Программные продукты и системы: международный научно-практический журнал / Издатель: Научно-исследовательский институт Центрпрограммсистем ; учредитель: В. П. Куприянов - Тверь: Научно-исследовательский институт Центрпрограммсистем, 2020 [ЭИ] URL: https://elibrary.ru/contents.asp?titleid=9834	Периодическое

6.2. Ресурсы сети Интернет

6.2.1. Электронные библиотечные системы

№	Название
1	Лань
2	ZNANIUM.COM
3	ЮРАЙТ
4	IPRbooks
5	E-library
6	Электронная библиотека ВГАУ

6.2.2. Профессиональные базы данных и информационные системы

№	Название	Размещение
1	Справочная правовая система Гарант	http://www.consultant.ru/
2	Справочная правовая система Консультант Плюс	http://ivo.garant.ru

6.2.3. Сайты и информационные порталы


№	Название	Размещение
1	"Программная инженерия. Инструменты и методы программной инженерии" (перевод. С.Орлик)	elck.ru/356XZd
2	Материалы Открытого национального института "ИНТУИТ" "Введение в программную инженерию"	https://www.intuit.ru/studies/mini_mba/3413/courses/353/info
3	Онлайн-сервис для программирования на python	https://www.online-python.com/
4	Веб-сервис для хостинга IT-проектов	https://github.com/
5	Коллективный блог по информационным технологиям, бизнесу и интернет "Харб"	https://habr.com/ru/

7. Материально-техническое и программное обеспечение дисциплины

7.1. Помещения для ведения образовательного процесса и оборудование

№	Наименование помещений для проведения всех видов учебной деятельности, предусмотренной учебным планом, в том числе помещения для самостоятельной работы, с указанием перечня основного оборудования, учебно-наглядных пособий и используемого программного обеспечения	Адрес (местоположение) помещений для проведения всех видов учебной деятельности, предусмотренной учебным планом (в случае реализации образовательной программы в сетевой форме дополнительно указывается наименование организации, с которой заключен договор)
1	Учебная аудитория для проведения занятий : комплект учебной мебели, демонстрационное оборудование, учебно-наглядные пособия в виде презентаций, программное обеспечение: MS Windows /Linux /Ред ОС	394087, Воронежская область, г. Воронеж, ул. Мичурина, д.1
2	Комплект учебной мебели, демонстрационное оборудование и учебно-наглядные пособия, компьютерная техника с возможностью подключения к сети "Интернет" и обеспечением доступа в электронную информационно-образовательную среду, используемое программное обеспечение MS Windows, Office MS Windows, DrWeb ES, 7-Zip, MediaPlayer Classic, Яндекс Браузер/ Mozilla Firefox / Internet Explorer, ALT Linux, LibreOffice	394087, Воронежская область, г. Воронеж, ул. Мичурина, д.1: 120
3	Комплект учебной мебели, демонстрационное оборудование и учебно-наглядные пособия, компьютерная техника с возможностью подключения к сети "Интернет" и обеспечением доступа в электронную информационно-образовательную среду, используемое программное обеспечение MS Windows, Office MS Windows, DrWeb ES, 7-Zip, MediaPlayer Classic, Яндекс Браузер/ Mozilla Firefox / Internet Explorer, ALT Linux, LibreOffice	394087, Воронежская область, г. Воронеж, ул. Мичурина, д.1, а.: 120 (с 16.00 до 20.00)

8. Междисциплинарные связи

Взаимосвязанные дисциплины		Кафедра, на которой преподается дисциплина	Подпись заведующего кафедрой
Код	Название		
Б1.О.24	Программная инженерия	Информационного обеспечения и моделирования агроэкономических систем	
Б1.В.12	Управление IT-проектами	Информационного обеспечения и моделирования агроэкономических систем	