


Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Алейник Станислав Николаевич
Должность: Ректор
Дата подписания: 19.02.2021 14:40:35
Уникальный программный ключ:
5258223550ea9fbeb23726a1609b644b33d8986ab6255891f288f913a1351fae

МИНИСТЕРСТВО СЕЛЬСКОГО ХОЗЯЙСТВА РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ АГРАРНЫЙ УНИВЕРСИТЕТ
имени В.Я.ГОРИНА

«УТВЕРЖДАЮ»


Декан инженерного факультета
С.В. Стребков
« 06 » 07 2018 г.

РАБОЧАЯ ПРОГРАММА

по дисциплине «Программная инженерия»

Направление 09.03.03 Прикладная информатика

Направленность (профиль) - Прикладная информатика в АПК

Квалификация - бакалавр

Майский, 2018

Рабочая программа составлена с учетом требований:

- федерального государственного образовательного стандарта высшего образования по направлению подготовки 09.03.03 «Прикладная информатика», утвержденного приказом Министерства образования и науки РФ от 12 марта 2015 г. № 207;
- порядка организации и осуществления образовательной деятельности по образовательным программам высшего образования – программам бакалавриата, программам специалитета, программам магистратуры, утвержденного приказом Министерства образования и науки РФ от 05.04.2017 г. № 301;
- основной профессиональной образовательной программы ФГБОУ ВО Белгородский ГАУ по направлению подготовки 09.03.03 «Прикладная информатика»

Составитель: канд.техн.наук, доцент Игнатенко В.А.

Рассмотрена на заседании кафедры информатики и информационных технологий от *21.06.* 2018 г., протокол № *13*

и.о. зав. кафедрой



В.А. Игнатенко

Одобрена методической комиссией инженерного факультета от *05.07.* 2018 г., протокол № *9-17/18*

Председатель методической комиссии



А.П. Слободюк

I. Цели и задачи дисциплины

Цель – формирование у студентов теоретических и практических навыков по изучению и использованию современных технологий разработки программного обеспечения в соответствии с международными стандартами обучения программной инженерии.

Задачи изучения дисциплины:

- формирование у студентов умений и навыков решения проблем оценки требований, проектирования, разработки, качества, повышения надежности и документирования программного обеспечения;
- изучение особенностей управления коллективной разработкой программного обеспечения.

II. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОСНОВНОЙ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ (ОПОП)

2.1. Цикл (раздел) ОПОП, к которому относится дисциплина

Дисциплина «Программная инженерия» относится к дисциплинам базовой части (Б1.Б.14) основной образовательной программы.

2.2. Логическая взаимосвязь с другими частями ОПОП

Наименование предшествующих дисциплин, практик, на которых базируется данная дисциплина (модуль)	1. Информатика и программирование 2. Пакеты прикладных программ 3. Информационные системы и технологии
Требования к предварительной подготовке обучающихся	<p><i>знать:</i></p> <ul style="list-style-type: none"> • основные программные конструкции; • основные способы хранения данных. <p><i>уметь:</i></p> <ul style="list-style-type: none"> • создавать программный алгоритм; • реализовывать программное пользовательское приложение. <p><i>владеть:</i></p> <ul style="list-style-type: none"> • навыками использования интегрированных сред программирования на одном из языков высокого уровня.

Знания, полученные при изучении дисциплины «Программная инженерия» используются в дальнейшем при изучении таких дисциплин как «Программирование информационных систем», «Разработка мобильных приложений», «Прикладное программирование», а так же при работе над ВКР.

III. ОБРАЗОВАТЕЛЬНЫЕ РЕЗУЛЬТАТЫ ОСВОЕНИЯ

ДИСЦИПЛИНЫ, СООТВЕТСТВУЮЩИЕ ФОРМИРУЕМЫМ КОМПЕТЕНЦИЯМ

Коды компетенций	Формулировка компетенции	Планируемые результаты обучения по дисциплине
ОПК-1	способность использовать нормативно-правовые документы, международные и отечественные стандарты в области информационных систем и технологий	<p>Знать:</p> <ul style="list-style-type: none"> нормативно-правовые документы и международные стандарты в области информационных систем и технологий. <p>Уметь:</p> <ul style="list-style-type: none"> применять международные стандарты и нормативно правовые стандарты в области информационных технологий. <p>Владеть:</p> <ul style="list-style-type: none"> методами поиска информации в нормативно-правовых документах и владеть навыками применения международных и отечественных стандартов при программировании информационных систем.
ПК-2	способность разрабатывать, внедрять и адаптировать прикладное программное обеспечение	<p>Знать:</p> <ul style="list-style-type: none"> теоретические основы по внедрению и адаптации прикладного программного обеспечения. <p>Уметь:</p> <ul style="list-style-type: none"> осуществлять внедрение и адаптацию прикладного программного обеспечения. <p>Владеть:</p> <ul style="list-style-type: none"> навыками по внедрению и адаптации прикладного программного обеспечения.
ПК-8	способность программировать приложения и создавать программные прототипы решения прикладных задач	<p>Знать:</p> <ul style="list-style-type: none"> особенности современных методологий и технологий создания программных средств; особенности современных методологий и технологий создания программных средств.

		<p>Уметь:</p> <ul style="list-style-type: none"> • использовать принципы и методы создания сложных программных средств на основе концепции и стандартов открытых систем, CASE-систем, языков 4-го поколения.
		<p>Владеть:</p> <ul style="list-style-type: none"> • современными средами разработки и отладки программных продуктов.
ПК-12	способностью проводить тестирование компонентов программного обеспечения ИС	<p>Знать:</p> <ul style="list-style-type: none"> • методы и средства тестирования компонентов ИС.
		<p>Уметь:</p> <ul style="list-style-type: none"> • составлять карты тестирования и планирование тестов.
		<p>Владеть:</p> <ul style="list-style-type: none"> • навыками проведения тестирования компонентов программного обеспечения ИС.

IV. ОБЪЕМ, СТРУКТУРА, СОДЕРЖАНИЕ ДИСЦИПЛИНЫ, ВИДЫ УЧЕБНОЙ РАБОТЫ И ФОРМЫ КОНТРОЛЯ ЗНАНИЙ

4.1. Распределение объема учебной работы по формам обучения

Вид работы	Объем учебной работы, час	
	Очная	Заочная
Формы обучения (вносятся данные по реализуемым формам)	Очная	Заочная
Семестр (курс) изучения дисциплины	4 семестр 2курс	3 курс
Общая трудоемкость, всего, час	252	252
<i>зачетные единицы</i>	7	7
Контактная работа обучающихся с преподавателем	100	28
Аудиторные занятия (всего)	100	28
В том числе:		
Лекции	40	12
Лабораторные занятия	60	16
Практические занятия	-	-
<i>Иные виды работ в соответствии с учебным планом (учебная практика)</i>	-	-
Внеаудиторная работа (всего)	20	6
В том числе:		
Контроль самостоятельной работы (на 1 подгруппу в форме компьютерного тестирования)	-*	-
Консультации согласно графику кафедры	20	6
<i>Иные виды работ в соответствии с учебным планом (курсовая работа, РГЗ и др.)</i>	-	-
Промежуточная аттестация	10	10
В том числе:		
Зачет	-	-
Экзамен (на 1 группу)	8	8
Консультация предэкзаменационная (на 1 группу)	2	2
Самостоятельная работа обучающихся	122	208
Самостоятельная работа обучающихся (всего)	122	208
в том числе:		
Самостоятельная работа по проработке лекционного материала (до 60% от объема лекций)	20	6
Самостоятельная работа по подготовке к лабораторно-практическим занятиям (до 60% от объема аудиторных занятий)	30	8
Работа над темами (вопросами), вынесенными на самостоятельное изучение	46	158
Самостоятельная работа по видам индивидуальных заданий: подготовка реферата (контрольной работы)	10	20
Подготовка к экзамену	16	16

Примечание: *осуществляется на аудиторных занятиях

4.2 Общая структура дисциплины и виды учебной работы

Наименование модулей и разделов дисциплины	Объемы видов учебной работы по формам обучения, час									
	Очная форма обучения					Заочная форма обучения				
	Всего	Лекции	Лабораторно-практ. занятия	Внеаудиторная работа и пр. агт.	Самостоятельная работа	Всего	Лекции	Лабораторно-практ. занятия	Внеаудиторная работа и пр. агт.	Самостоятельная работа
1	2	3	4	5	6	7	8	9	10	11
Модуль 1	92	18	26	10	38	96	6	8	2	80
1. Введение в дисциплину. О предмете изучения.	18	4	6	Консультации	8	23	1	2	Консультации	20
2. Процесс разработки программного обеспечения	18	4	6		8	23	1	2		20
3. Рабочий продукт, дисциплина обязательств, проект.	20	4	6		10	24	2	2		20
4. Архитектура ПО.	22	6	6		10	24	2	2		20
<i>Итоговое занятие по модулю 1</i>	4	-	2		2	-	-	-		-
Модуль 2	124	22	34	10	58	110	6	8	4	92
1. Управление требованиями».	20	4	6	Консультации	10	21	1	2	Консультации	18
2. Конфигурационное управление	28	6	8		14	22	2	2		18
3. Тестирование	20	4	6		10	19	1	-		18
4. Диаграммные техники в работе со знаниями	20	4	6		10	21	1	2		18
5. Методология MSF.	22	4	6		12	23	1	2		20
<i>Итоговое занятие по модулю 2</i>	4	-	2	2	-	-	-	-		
Подготовка реферата в форме презентации (контрольной работы)	10	-	-	-	10	20	-	-	-	20
Экзамен	26	-	-	10	16	26	-	-	10	16

4.3 Структура и содержание дисциплины по формам обучения

Наименование модулей и разделов дисциплины	Объемы видов учебной работы по формам обучения, час									
	Очная форма обучения					Заочная форма обучения				
	Всего	Лекции	Лабор.практ. зан.	Внеаудит. работа	Самост. работа	Всего	Лекции	Лабор.практ. зан.	Внеаудит. работа	Самост. работа
1	2	3	4	5	6	7	8	9	10	11
Модуль 1	92	18	26	10	38	96	6	8	2	80
1. Введение в дисциплину. О предмете изучения.	18	4	6	Консультации	8	23	1	2	Консультации	20
1.1 Место дисциплины. История развития. Основные понятия. Значимость программной инженерии.	18	4	6		8	23	1	2		20
2. Процесс разработки программного обеспечения.	18	4	6		8	23	1	2		20
2.1 Рассматривается последовательность и виды работ, выполняемых при разработке коммерческого программного продукта. Жизненный цикл программного продукта Модели жизненного цикла: каскадная модель, спиральная модель. Работы и фазы жизненного цикла.	18	4	6		8	23	1	2		20
3. Рабочий продукт, дисциплина обязательств, проект.	20	4	6		10	24	2	2		20
3.1 Понятие рабочего продукта. Виды рабочего продукта (программный код, проект, документация, процесс и др.). Понятие дисциплины обязательств. Требования к участникам команды разработчиков. Обязательства, возлагаемы на разработчиков. «Неправильная» модель поведения..	20	4	6		10	24	2	2		20
4. Архитектура ПО.	22	6	6		10	24	2	2		20
4.1 Понятие архитектуры. Наиболее часто встречающиеся архитектуры (файл-серверная, клиент-серверная, трёхуровневая, тонкий клиент). Процесс разработки архитектуры. Стандарты и методики разработки архитектуры ПО.	22	6	6	10	24	2	2	20		
<i>Итоговое занятие по модулю 1</i>	4	-	2		2	-	-	-	-	
Модуль 2	124	22	34	10	58	110	6	8	4	92
1. Управление требованиями.	20	4	6	Консультации	10	21	1	2	Консультации	18
1.1. Требования при разработке ПО. Виды требований. Стратегии управления требованиями. Статические и динамические требования.	20	4	6		10	21	1	2		18
2. Конфигурационное управление.	28	6	8		14	22	2	2		18
2.1. Понятие конфигурации, сборки, версии. Системы управления версиями. Системы автоматической сборки. Управление сборками..	28	6	8		14	22	2	2		18
3. Тестирование.	20	4	6		10	19	1	-		18
3.1 Назначение этапа тестирования программного обеспечения. Ручное тестирование. Системы автоматического тестирования. Требования к этапу тестирования. Подходы к тестированию(метод «чёрного ящика», метод «белого ящика»).	20	4	6	10	19	1	-	18		

Наименование модулей и разделов дисциплины	Объемы видов учебной работы по формам обучения, час									
	Очная форма обучения					Заочная форма обучения				
	Всего	Лекции	Лабор.практ. зан.	Внеаудит. работа	Самост. работа	Всего	Лекции	Лабор.практ. зан.	Внеаудит. работа	Самост. работа
1	2	3	4	5	6	7	8	9	10	11
4. Диаграммные техники в работе со знаниями.	20	4	6		10	21	1	2		18
4.1 Методы обобщения и структуризации знаний. Графические методы описания знаний. Язык проектирования UML. Диаграммы языка UML (на примере разработки объектно-ориентированных подходов к проектированию).	20	4	6		10	21	1	2		18
5. Методология MSF.	22	4	6		12	23	1	2		20
5.1 Особенности разработки ПО согласно методики MSF. Роль и требования к команде разработчиков. Понятие и виды ролей в команде. Циклы разработки, стадии.	22	4	6		12	23	1	2		20
<i>Итоговое занятие по модулю 2</i>	<i>4</i>	<i>-</i>	<i>2</i>		<i>2</i>	<i>-</i>	<i>-</i>	<i>-</i>		<i>-</i>
<i>Подготовка реферата в форме презентации (контрольной работы)</i>	<i>10</i>	<i>-</i>	<i>-</i>	<i>-</i>	<i>10</i>	<i>20</i>	<i>-</i>	<i>-</i>	<i>-</i>	<i>20</i>
Экзамен	26	-	-	10	16	26	-	-	10	16

**V. ОЦЕНКА ЗНАНИЙ И ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ПРОВЕДЕНИЯ
ТЕКУЩЕГО КОНТРОЛЯ ЗНАНИЙ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ
ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ**

**5.1. Формы контроля знаний, рейтинговая оценка и формируемые
компетенции (дневная форма обучения)**

№ п/п	Наименование рейтингов, модулей и блоков	Формируемые компетенции	Объем учебной работы					Форма конт- роля знаний	Количество баллов (max)
			Общая трудоемкос	Лекции	Лабор.-практ.заня	Внеаудиторн. раб. и промежут.аттест.	Самост. работа		
Всего по дисциплине		ОПК-1, ПК-2, ПК-8, ПК-12	252	40	60	30	122	Экзамен	100
<i>I. Входной рейтинг</i>								Устный опрос	5
<i>III. Рубежный рейтинг</i>								Сумма баллов за модули	60
Модуль 1		ОПК-1, ПК-2, ПК-8, ПК-12	92	18	26	10	38		30
1.	Введение в дисциплину. О предмете изучения.		18	4	6	Консультации	8	Устный опрос	
2.	Процесс разработки программного обеспечения		18	4	6		8	Устный опрос	
3.	Рабочий продукт, дисциплина обязательств, проект.		20	4	6		10	Устный опрос	
4.	Архитектура ПО		22	6	6		10	Устный опрос	
Итоговый контроль знаний по темам модуля 1.			4	-	2		2	Тестирование, ситуационные задачи	
Модуль 2		ОПК-1, ПК-2, ПК-8, ПК-12	124	22	34	10	58		30
1.	Управление требованиями		20	4	6	Консультации	10	Устный опрос, ситуационные задачи	
2.	Конфигурационное управление		28	6	8		14	Устный опрос, решение задач	
3.	Тестирование		20	4	6		10	Устный опрос, решение задач	
4.	Диаграммные техники в работе со знаниями		20	4	6		10	Устный опрос, решение задач	
5.	Методология MSF		22	4	6		12	Устный опрос, решение задач	

Итоговый контроль знаний по темам модуля 2.		4	-	2		2	Тестирование, ситуационные задачи	
<i>III. Творческий рейтинг</i>		10	-	-	-	10	<i>Реферат</i>	5
<i>IV. Выходной рейтинг</i>		26	-	-	10	16	<i>Экзамен</i>	30

5.2. Оценка знаний студента

5.2.1. Основные принципы рейтинговой оценки знаний

Уровень развития компетенций оценивается с помощью рейтинговых баллов.

Рейтинги	Характеристика рейтингов	Максимум баллов
Входной	Отражает степень подготовленности студента к изучению дисциплины. Определяется по итогам входного контроля знаний на первом практическом занятии.	5
Рубежный	Отражает работу студента на протяжении всего периода изучения дисциплины. Определяется суммой баллов, которые студент получит по результатам изучения каждого модуля.	60
Творческий	Результат выполнения студентом индивидуального творческого задания различных уровней сложности, в том числе, участие в различных конференциях и конкурсах на протяжении всего курса изучения дисциплины.	5
Выходной	Является результатом аттестации на окончательном этапе изучения дисциплины по итогам сдачи экзамена. Отражает уровень освоения информационно-теоретического компонента в целом и основ практической деятельности в частности.	30
Общий рейтинг	Определяется путём суммирования всех рейтингов	100

Итоговая оценка компетенций студента осуществляется путём автоматического перевода баллов общего рейтинга в стандартные оценки.

Неудовлетворительно	Удовлетворительно	Хорошо	Отлично
менее 51 балла	51-67 баллов	68-85 баллов	86-100 баллов

5.2.2. Критерии оценки знаний студента на экзамене

На экзамене студент отвечает в письменно-устной форме на вопросы экзаменационного билета (вопрос, тест и задача).

Количественная оценка на экзамене определяется на основании следующих критериев:

- оценку «отлично» заслуживает студент, показавший всестороннее систематическое и глубокое знание учебно-программного материала, умение свободно выполнять задания, предусмотренные программой, усвоивший основную и знакомый с дополнительной литературой, рекомендованной программой; как правило, оценка «отлично» выставляется студентам, усвоившим взаимосвязь основных понятий дисциплины и их значение для приобретаемой профессии, проявившим творческие способности в понимании, изложении и использовании учебно-программного материала;

- оценку «хорошо» заслуживает студент, обнаруживший полное знание учебно-программного материала, успешно выполняющий предусмотренные в программе задания, усвоивший основную литературу, рекомендованную в программе; как правило, оценка «хорошо» выставляется студентам, показавшим систематический характер знаний по дисциплине и способным к их самостоятельному пополнению и обновлению в ходе дальнейшей учебной работы и профессиональной деятельности;

- оценку «удовлетворительно» заслуживает студент, обнаруживший знания основного учебно-программного материала в объеме, необходимом для дальнейшей учебы и предстоящей работы по профессии, справляющийся с выполнением заданий, предусмотренных программой, знакомый с основной литературой, рекомендованной программой; как правило, оценка «удовлетворительно» выставляется студентам, допустившим погрешности в ответе на экзамене и при выполнении экзаменационных заданий, но обладающим необходимыми знаниями для их устранения под руководством преподавателя;

- оценка «неудовлетворительно» выставляется студенту, обнаружившему проблемы в знаниях основного учебно-программного материала, допустившему принципиальные ошибки в выполнении предусмотренных программой заданий; как правило, оценка «неудовлетворительно» ставится студентам, которые не могут продолжать обучение или приступить к профессиональной деятельности по окончании вуза без дополнительных занятий по соответствующей дисциплине.

5.3. Фонд оценочных средств. Типовые контрольные задания или иные материалы, необходимые для оценки формируемых компетенций по дисциплине (приложение 2)

VI. УЧЕБНО - МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

6.1 Основная учебная литература

1. Антипов, В.А. Введение в программную инженерию : Учебник [Электронный ресурс] / В.А. Антипов, А.А. Бубнов, А.Н. Пылькин, В.К. Столчнев. — М.: КУРС: ИНФРА-М, 2015. — 336 с. Режим доступа: <http://znanium.com/bookread2.php?book=850951>

6.2 Дополнительная литература

1. Антамошкин, О. А. Программная инженерия. Теория и практика [Электронный ресурс]: учебник / О. А. Антамошкин. - Красноярск: Сиб. Федер. ун-т, 2012. - 247 с. Режим доступа: <http://znanium.com/bookread2.php?book=492527>

2. Игнатенко, В. А. Методические указания и задания к выполнению лабораторно-практических и самостоятельных работ студентов по дисциплине "Программная инженерия" для студентов экономического факультета направления "Прикладная информатика" [Электронный ресурс] : методические указания / Белгородский ГАУ ; сост. В. А. Игнатенко. - Белгород : Белгородский ГАУ, 2014. - 75 с. Режим доступа: <https://clck.ru/EaGZd>

3. Методические указания и задания к выполнению самостоятельной работы по дисциплине "Программная инженерия" для студентов экономического факультета направления "Прикладная информатика" [Электронный ресурс] : методические указания / Белгородский ГАУ ; сост.: В. А. Игнатенко, Д. А. Петросов, В. Л. Михайлова. - Белгород : Белгородский ГАУ, 2015. - 34 с. Режим доступа: <https://clck.ru/FDqLT>

6.3. Учебно-методическое обеспечение самостоятельной работы обучающихся по дисциплине

Самостоятельная работа студентов заключается в инициативном поиске информации о наиболее актуальных проблемах, которые имеют большое практическое значение и являются предметом научных дискуссий в рамках изучаемой дисциплины.

Самостоятельная работа планируется в соответствии с календарными планами рабочей программы по дисциплине и в методическом единстве с тематикой учебных аудиторных занятий.

Самостоятельную работу студента поддерживает электронная информационная среда ВУЗа, доступ к которой <http://do.belgau.edu.ru> (логин, пароль студента)

6.3.1. Методические указания по освоению дисциплины

1. Методические указания и задания к выполнению самостоятельной работы по дисциплине "Программная инженерия" для студентов

экономического факультета направления "Прикладная информатика" [Электронный ресурс] : методические указания / Белгородский ГАУ ; сост.: В. А. Игнатенко, Д. А. Петросов, В. Л. Михайлова. - Белгород : Белгородский ГАУ, 2015. - 34 с. Режим доступа: <https://clck.ru/EaGZ4>

6.3.2. Видеоматериалы

1. https://www.youtube.com/watch?v=KJ65Q8rmy_0
2. <https://www.youtube.com/watch?v=h7uKK7nrBfw>
3. https://www.youtube.com/watch?v=GT9f4K_e1Kw
4. <https://www.youtube.com/watch?v=d9sBXGUtZaM>
5. <https://www.youtube.com/watch?v=WXivzQZD0gE>
6. <https://www.youtube.com/watch?v=srO7HhWJFoc>
7. https://www.youtube.com/watch?v=aNVP_WW-xlQ
8. <https://www.youtube.com/watch?v=5g7ULLqCwpw>
9. <https://www.youtube.com/watch?v=ij3E1uzreHA>
10. <https://www.youtube.com/watch?v=nrAanTkJN9M>

6.3.3 Печатные периодические издания

1. Журнал «Программная инженерия»
<http://www.novtex.ru/prin/rus/archive.html>
2. Журнал «Вестник российской сельскохозяйственной науки»
3. Журнал «Достижения науки и техники АПК»
4. Журнал «Экономика, статистика и информатика»

6.4. Ресурсы информационно-телекоммуникационной сети «Интернет», современные профессиональные базы данных, информационные справочные системы.

1. Центр Открытых Систем - Совет РАН по автоматизации научных исследований - <http://www.cplire.ru>
2. Новые информационные технологии и программы - Сайт о сво-бодном программном обеспечении и новых информационных тех-нологиях - <http://pro-spo.ru/>
3. CITForum.ru - on-line библиотека свободно доступных материалов по информационным технологиям на русском языке - <http://citforum.ru>
4. Информационная система «Единое окно доступа к образователь-ным ресурсам. Раздел. Информатика и информационные техноло-гии» - <http://window.edu.ru>
5. Профессиональная база данных и информационно справочная система по официальной технической документации для разработчиков под ОС [Microsoft Windows](https://msdn.microsoft.com/ru-ru) <https://msdn.microsoft.com/ru-ru>
6. Профессиональная база данных и информационно справочная система по официальной технической документации для разработчиков под

ОС Microsoft Windows <https://technet.microsoft.com/ru-ru>

6.5. Перечень программного обеспечения, информационных технологий.

1. Операционная система Windows;
2. Пакет программ Microsoft Office;
3. SunRav – программа для тестирования;
4. Специализированные инструментальные программные средства и системы:
 - a. Система контроля версий Git (бесплатное программное обеспечение <https://git-scm.com/downloads>)
 - b. Интегрированная среда разработки CodeBlocks. (бесплатное программное обеспечение <http://www.codeblocks.org/downloads>)

VII. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Для преподавания дисциплины используются:

1. учебная аудитория лекционного типа, оборудованная мультимедийным оборудованием для демонстрации презентаций;
2. компьютерный класс для проведения лабораторно – практических занятий.
3. помещение для самостоятельной работы обучающихся, оснащенное компьютерной техникой с подключением к сети Интернет и электронной информационно-образовательной среде ВУЗа.

VIII. ПРИЛОЖЕНИЯ

Приложение 1

**СВЕДЕНИЯ О ДОПОЛНЕНИИ И ИЗМЕНЕНИИ
РАБОЧЕЙ ПРОГРАММЫ ДИСЦИПЛИНЫ
НА 201 / 201 УЧЕБНЫЙ ГОД**

Программная инженерия

дисциплина (модуль)

09.03.03 Прикладная информатика

направление подготовки/специальность

ДОПОЛНЕНО (с указанием раздела РПД)
ИЗМЕНЕНО (с указанием раздела РПД)
УДАЛЕНО (с указанием раздела РПД)

Реквизиты протоколов заседаний кафедр, на которых пересматривалась
программа

Кафедра информатики и информационных технологий	Кафедра информатики и информационных технологий
от _____ № _____ Дата	от _____ № _____ дата

Методическая комиссия инженерного факультета

« ___ » _____ 201 года, протокол № _____

Председатель методической комиссии

Слободюк А.П.

Декан инженерного факультета

Стребков С.В.

« ___ » _____ 201 г.

Согласовано:

Директор
ООО «Матрица»



ФОНД ОЦЕНОЧНЫХ СРЕДСТВ
для проведения промежуточной аттестации обучающихся
по дисциплине Программная инженерия
Направление подготовки 09.03.03 Прикладная информатика
Профиль «Прикладная информатика в АПК»

1.Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы

Код контролируемой компетенции	Формулировка контролируемой компетенции	Этап (уровень) освоения компетенции	Планируемые результаты обучения	Наименование модулей и (или) разделов дисциплины	Наименование оценочного средства	
					Текущий контроль	Промежуточная аттестация
ОПК-1	способность использовать нормативно-правовые документы, международные и отечественные стандарты в области информационных систем и технологий	Первый этап (пороговой уровень)	Знать: нормативно-правовые документы и международные стандарты в области информационных систем и технологий.	Модуль 1	Устный опрос	Итоговое тестирование, вопросы к экзамену
					Тестирование	
					Решение ситуационных задач	
					Подготовка рефератов	
				Модуль 2	Устный опрос	Итоговое тестирование, вопросы к экзамену
					Тестирование	
					Решение ситуационных задач	
					Подготовка рефератов	

					Тестирование	
					Решение ситуационных задач	
					Подготовка рефератов	
	Второй этап (продвинутый уровень)	Уметь: применять международные стандарты и нормативно правовые стандарты в области информационных технологий	Модуль 1 «Основы системной архитектуры ИС и управления ресурсами»	Устный опрос	Итоговое тестирование, вопросы к экзамену	
Тестирование						
Решение ситуационных задач						
Подготовка рефератов						
			Модуль 2 «Архитектуры ИС: стратегии и связи»	Устный опрос	Итоговое тестирование, вопросы к экзамену	
Тестирование						
Решение ситуационных задач						
	Третий этап	Владеть: методами	Модуль 1	Устный опрос	Итоговое	

		(высокий уровень)	поиска информации в нормативно-правовых документах и владеть навыками применения международных и отечественных стандартов при программировании информационных систем	«Основы системной архитектуры ИС и управления ресурсами»	Тестирование	тестирование, вопросы к экзамену
					Решение ситуационных задач	
					Подготовка рефератов	
					Выполнение курсовой работы	
				Модуль 2	Устный опрос	
				«Архитектуры ИС: стратегии и связи»	Тестирование	
					Решение ситуационных задач	
					Подготовка рефератов	
					Выполнение курсовой работы	
ПК-2	способность разрабатывать, внедрять и	Первый этап (пороговой)	Знать: теоретические основы по внедрению и	Модуль 1	Устный опрос	Итоговое тестирование, вопросы к
					Тестирование	

	адаптировать прикладное программное обеспечение	уровень)	адаптации прикладного программного обеспечения.		Решение ситуационных задач	экзамену
					Подготовка рефератов	
				Модуль 2	Устный опрос	Итоговое тестирование, вопросы к экзамену
					Тестирование	
	Решение ситуационных задач					
	Подготовка рефератов					
	Второй этап (продвинутый уровень)	Уметь: осуществлять внедрение и адаптацию прикладного программного обеспечения.		Модуль 1	Устный опрос	Итоговое тестирование, вопросы к экзамену
					Тестирование	
					Решение ситуационных задач	
					Подготовка рефератов	
Модуль 2	Устный опрос	Итоговое тестирование, вопросы к				
	Тестирование					

					Решение ситуационных задач	экзамену	
					Подготовка рефератов		
		Третий этап (высокий уровень)	Владеть: навыками по внедрению и адаптации прикладного программного обеспечения.	Модуль 1	Устный опрос	Итоговое тестирование, вопросы к экзамену	
					Тестирование		
					Решение ситуационных задач		
					Подготовка рефератов		
				Выполнение курсовой работы			
				Модуль 2	Устный опрос		Итоговое тестирование, вопросы к экзамену
					Тестирование		
					Решение ситуационных задач		
		Подготовка					

					рефератов	
ПК-8	способность программировать приложения и создавать программные прототипы решения прикладных задач	Первый этап (пороговой уровень)	Знать: особенности современных методологий и технологий создания программных средств; особенности современных методологий и технологий создания программных средств.	Модуль 1	Устный опрос	Итоговое тестирование, вопросы к экзамену
					Тестирование	
					Решение ситуационных задач	
					Подготовка рефератов	
		Модуль 2»	Устный опрос	Итоговое тестирование, вопросы к экзамену		
			Тестирование			
			Решение ситуационных задач			
			Подготовка рефератов			
Второй этап (продвинутый уровень)	Уметь: использовать принципы и методы создания сложных программных средств на основе концепции и стандартов открытых систем,	Модуль 1	Устный опрос	Итоговое тестирование, вопросы к экзамену		
			Тестирование			
			Решение ситуационных задач			
			Подготовка			

			CASE- систем, языков 4-го поколения.		рефератов	
				Модуль 2	Устный опрос	Итоговое тестирование, вопросы к экзамену
					Тестирование	
					Решение ситуационных задач	
					Подготовка рефератов	
	Третий этап (высокий уровень)	Владеть: современными средами разработки и отладки программных продуктов.	Модуль 1	Устный опрос	Итоговое тестирование, вопросы к экзамену	
				Тестирование		
				Решение ситуационных задач		
				Подготовка рефератов		
				Выполнение курсовой работы		
			Модуль 2	Устный опрос	Итоговое тестирование, вопросы к	
				Тестирование		
				Решение		

					ситуационных задач	экзамену
					Подготовка рефератов	
ПК-12	способностью проводить тестирование компонентов программного обеспечения ИС	Первый этап (пороговой уровень)	Знать: методы и средства тестирования компонентов ИС	Модуль 1»	Устный опрос	Итоговое тестирование, вопросы к экзамену
					Тестирование	
					Решение ситуационных задач	
					Подготовка рефератов	
		Модуль 2	Устный опрос	Итоговое тестирование, вопросы к экзамену		
			Тестирование			
			Решение ситуационных задач			
			Подготовка рефератов			
Второй этап (продвинутый)	Уметь: составлять карты тестирования и	Модуль 1	Устный опрос	Итоговое тестирование, вопросы к		
			Тестирование			

		уровень)	планирование тестов.		Решение ситуационных задач	экзамену	
					Подготовка рефератов		
				Модуль 2	Устный опрос	Итоговое тестирование, вопросы к экзамену	
					Тестирование		
					Решение ситуационных задач		
					Подготовка рефератов		
		Третий этап (высокий уровень)	Владеть: навыками проведения тестирования компонентов программного обеспечения ИС.	Модуль 1	Устный опрос	Итоговое тестирование, вопросы к экзамену	
							Тестирование
							Решение ситуационных задач
							Подготовка рефератов
				Модуль 2	Устный опрос	Итоговое тестирование, вопросы к	
					Тестирование		

--	--	--	--

		Решение ситуационных задач	экзамену
		Подготовка рефератов	

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

Компетенция	Планируемые результаты обучения (показатели достижения заданного уровня компетенции)	Уровни и критерии оценивания результатов обучения, шкалы оценивания			
		Компетентность не сформирована	Пороговый уровень компетентности	Продвинутый уровень компетентности	Высокий уровень
		Не зачтено/ неудовлетворительно	Зачтено/ удовлетворительно	Зачтено/ хорошо	Зачтено/ отлично
ОПК-1	Способность использовать нормативно-правовые документы, международные и отечественные стандарты в области информационных систем и технологий	Способность использовать нормативно-правовые документы, международные и отечественные стандарты в области информационных систем и технологий не сформирована	частично владеет способностью использовать нормативно-правовые документы, международные и отечественные стандарты в области информационных систем и технологий	владеет способностью использовать нормативно-правовые документы, международные и отечественные стандарты в области информационных систем и технологий	свободно владеет способностью использовать нормативно-правовые документы, международные и отечественные стандарты в области информационных систем и технологий
	Знать: нормативно-правовые документы, международные и отечественные	Не знает теоретические основы нормативно-правовых документов, международных и	Имеет неполные знания в области теоретических основ нормативно-правовых	Имеет достаточные знания теоретических основ нормативно-правовых документов,	Имеет четкое, полностью сформированное представление знаний

	стандарты в области информационных систем и технологий	отечественных стандартов в области информационных систем и технологий	документов, международных и отечественных стандартов в области информационных систем и технологий	международных и отечественных стандартов в области информационных систем и технологий	в области теоретических основ нормативно-правовых документов, международных и отечественных стандартов в области информационных систем и технологий
	Уметь: проводить анализ нормативно-правовых документов, международных и отечественных стандартов в области информационных технологий	Не способен проводить анализ нормативно-правовых документов, международных и отечественных стандартов в области информационных технологий	Допускает ошибки при проведении анализа нормативно-правовых документов, международных и отечественных стандартов в области информационных технологий	Способен проводить анализ нормативно-правовых документов, международных и отечественных стандартов в области информационных технологий	Четко и аргументированно проводит анализ нормативно-правовых документов, международных и отечественных стандартов в области информационных технологий
	Владеть: навыками применения отечественных и международных стандартов и нормативно-правовых документов при проектировании информационных систем	Не владеет навыками применения отечественных и международных стандартов и нормативно-правовых документов при проектировании информационных систем	Частично владеет навыками применения отечественных и международных стандартов и нормативно-правовых документов при проектировании	Владеет навыками применения отечественных и международных стандартов и нормативно-правовых документов при проектировании	В совершенстве владеет навыками применения отечественных и международных стандартов и нормативно-правовых документов при

			информационных систем	информационных систем	проектировании информационных систем
ПК-2	способность разрабатывать, внедрять и адаптировать прикладное программное обеспечение	способность разрабатывать, внедрять и адаптировать прикладное программное обеспечение не сформирована	частично владеет способностью разрабатывать, внедрять и адаптировать прикладное программное обеспечение	владеет способностью разрабатывать, внедрять и адаптировать прикладное программное обеспечение	свободно владеет способностью разрабатывать, внедрять и адаптировать прикладное программное обеспечение
	Знать: теоретические основы по внедрению и адаптации прикладного программного обеспечения.	Не знает теоретические основы по внедрению и адаптации прикладного программного обеспечения.	Имеет неполные знания теоретических основ по внедрению и адаптации прикладного программного обеспечения.	Имеет достаточные знания в области теоретических основ по внедрению и адаптации прикладного программного обеспечения, допускает неточности	Имеет четкое, полностью сформированное представление в области теоретических основ по внедрению и адаптации прикладного программного обеспечения, допускает неточности
	Уметь: осуществлять внедрение и адаптацию прикладного программного	Не способен осуществлять внедрение и адаптацию прикладного программного	Допускает ошибки при внедрении и адаптации прикладного программного	Способен проводить внедрение и адаптацию прикладного программного	Четко и аргументировано осуществляет внедрение и

	обеспечения.	обеспечения.	обеспечения.	обеспечения. Допускает небольшие неточности.	адаптацию прикладного программного обеспечения.
	Владеть: навыками по внедрению и адаптации прикладного программного обеспечения	Не владеет навыками по внедрению и адаптации прикладного программного обеспечения	Частично владеет навыками по внедрению и адаптации прикладного программного обеспечения	Владеет навыками по внедрению и адаптации прикладного программного обеспечения. Допускает неточности.	В совершенстве владеет навыками по внедрению и адаптации прикладного программного обеспечения
ПК-8	способность программировать приложения и создавать программные прототипы решения прикладных задач	способность программировать приложения и создавать программные прототипы решения прикладных задач не сформирована	частично владеет способностью программировать приложения и создавать программные прототипы решения прикладных задач	владеет способностью программировать приложения и создавать программные прототипы решения прикладных задач	свободно владеет способностью программировать приложения и создавать программные прототипы решения прикладных задач
	<ul style="list-style-type: none"> Знать особенности современных методологий и технологий создания программных средств; особенности современных 	Не знает особенности современных методологий и технологий создания программных средств и современных методологий и технологий создания программных средств.	Имеет фрагментарные знания о особенностях современных методологий и технологий создания программных средств и современных методологий и технологий создания	Знает особенности современных методологий и технологий создания программных средств; особенности современных методологий и	Имеет четкие знания особенности современных методологий и технологий создания программных средств; особенности современных

	методологий и технологий создания программных средств.		программных средств.	технологий создания программных средств. Допускает небольшие неточности.	методологий и технологий создания программных средств.
	Уметь: использовать принципы и методы создания сложных программных средств на основе концепции и стандартов открытых систем, CASE- систем, языков 4-го поколения..	Не умеет использовать принципы и методы создания сложных программных средств на основе концепции и стандартов открытых систем, CASE- систем, языков 4-го поколения...	Допускает серьезные ошибки при использовании принципов и методов создания сложных программных средств на основе концепции и стандартов открытых систем, CASE- систем, языков 4-го поколения..	Умеет использовать принципы и методы создания сложных программных средств на основе концепции и стандартов открытых систем, CASE- систем, языков 4-го поколения. Допускает небольшие неточности.	Способен логично и четко использовать принципы и методы создания сложных программных средств на основе концепции и стандартов открытых систем, CASE- систем, языков 4-го поколения..
	Владеть: современными средами разработки и отладки программных продуктов.	Не владеет современными средами разработки и отладки программных продуктов.	Не полностью владеет современными средами разработки и отладки программных продуктов.	Владеет современными средами разработки и отладки программных продуктов. Допускает небольшие неточности	В совершенстве владеет современными средами разработки и отладки программных продуктов.
ПК-12	способностью проводить тестирование компонентов программного обеспечения ИС	способностью проводить тестирование компонентов программного обеспечения ИС не сформирована	частично владеет способностью проводить тестирование компонентов программного	владеет способностью проводить тестирование компонентов программного обеспечения ИС	свободно владеет способностью проводить тестирование компонентов программного

			обеспечения ИС		обеспечения ИС
	Знать: методы и средства тестирования компонентов ИС.	Не знает методы и средства тестирования компонентов ИС.	Частично знает методы и средства тестирования компонентов ИС.	Знает методы и средства тестирования компонентов ИС. Допускает небольшие неточности	Знает методы и средства тестирования компонентов ИС.
	Уметь: составлять карты тестирования и планирование тестов.	Не способен составлять карты тестирования и планирование тестов.	Способен со значительными ошибками составлять карты тестирования и планирование тестов.	Способен составлять карты тестирования и планирование тестов, с незначительными ошибками	Способен составлять карты тестирования и планирование тестов.
	Владеть: навыками проведения тестирования компонентов программного обеспечения ИС.	Не владеет навыками проведения тестирования компонентов программного обеспечения ИС.	Частично владеет навыками проведения тестирования компонентов программного обеспечения ИС.	Владеет навыками проведения тестирования компонентов программного обеспечения ИС, допускает мелкие ошибки	В совершенстве владеет навыками проведения тестирования компонентов программного обеспечения ИС.

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

3.1. Первый этап (пороговой уровень)

ЗНАТЬ (помнить и понимать): студент помнит, понимает и может продемонстрировать широкий спектр фактических, концептуальных, процедурных знаний.

3.1.1. Перечень вопросов для определения входного рейтинга

1. Какое определение стандарт дает программному средству?
2. Что такое программное изделие?
3. Что производит отрасль производства –информатика?
4. В чем особенность программного изделия, как продукта производства?
5. Что такое моральный износ?
6. Почему ПИ не подвержено физическому износу?
7. Что такое жизненный цикл программного обеспечения?
8. Чем регламентируется ЖЦ ПО?
9. Какие группы процессов входят в состав ЖЦ ПО и какие процессы входят в состав каждой группы?
10. Какие процессы, по вашему мнению, наиболее часто используются в реальных процессах и почему?
11. Что понимается под стадией ЖЦ ПО и какие стадии входят в его состав?
12. Каково соотношение между стадиями и процессами ЖЦ ПО?
13. Какие процессы ЖЦ используются для получения достоверных оценок качества ПО?
14. Что называют моделью ЖЦ ПО?
15. Как выбирается модель ЖЦ ПО?
16. Какие стадии ЖЦ чаще всего присутствуют в модели ЖЦ?
17. Каковы принципиальные особенности каскадной модели?
18. В чем заключаются преимущества и недостатки каскадной модели?
19. Каковы принципиальные особенности спиральной модели?
20. В чем заключаются преимущества и недостатки спиральной модели?
21. Каковы особенности RAD-подхода при разработке прикладного ПО?
22. Каким образом определяются метод и технология проектирования ПО?
23. Какие стандарты необходимы для выполнения конкретного проекта?
24. Какие задачи должны быть решены в процессе разработки требований к проектируемому ПО?
25. Какие этапы включает стадия формирования требований к ПО?
26. Какую информацию отражает модель TO-BE?
27. Какую информацию отражает модель AS-IS?
28. В чем заключаются основные принципы структурного подхода?
29. Какой стандарт на основе метода SADT был принят как федеральный стандарт США?
30. Чем определяются интерфейсы между функциями в модели SADT?
31. Что общего и в чем различия между методом SADT и моделированием потоков данных?
32. В чем заключаются достоинства и недостатки структурного подхода?
33. Какие модели предпроектного исследования используются на этапе проектирования и для чего?

34. Что является результатом этапа проектирования ПС?
35. Что такое модуль, какими он обладает признаками?
36. Назовите основные положения на которых основаны принципы модульности.
37. Что такое функциональная связность?
38. Какие виды связности модулей вы знаете?
39. Что такое сцепление модулей?
40. Какие виды сцепления модулей вы знаете?
41. Что понимают под стилем программирования?
42. Что такое детальное кодирование?
43. В чем заключаются основные принципы объектно-ориентированного подхода?
44. Что общего и в чем различия между структурным и объектно-ориентированным подходом?
45. В чем заключаются достоинства и недостатки объектно-ориентированного подхода?
46. Что такое UML?
47. Дайте определение объекту.
48. Что такое класс, чем он характеризуется?
49. Что такое ассоциация?
50. Что такое агрегация?
51. С какой модели начинается анализ системы?
52. Какие виды моделей используется при объектно-ориентированном подходе?
53. Какие эргономические характеристики влияют на работу пользователя с ПК?
54. Что такое интерфейс?
55. Каких правил нужно придерживаться при разработке интерфейса?
56. Какой диалог пользователя с компьютером можно назвать хорошим диалогом?
57. Изложите основные принципы при проектировании диалога типа меню.
58. Каких правил нужно придерживаться при проектировании оконной формы диалога?
59. Какие правила нужно помнить при размещении и выделении информации на экране?
60. Перечислите требования для разработки модулей помощи и справки.
61. Что такое тестирование программы?
62. Чем отличается процесс тестирования от процесса отладки?
63. Перечислите принципы тестирования.
64. Какие методы тестирования вы знаете?
65. Что понимают под процессом сборки модулей, какие методы сборки вы знаете?
66. Какие виды ошибок вы знаете?
67. Когда должна заканчиваться стадия тестирования ПО?
68. Как можно охарактеризовать процесс тестирования по стоимости и продолжительности?
69. Как связаны между собой тестирование и надежность ПС?
70. Что представляет собой процесс сопровождения ПО?
71. Какие мероприятия могут проводиться в процессе сопровождения ПО?
72. Какие мероприятия проводятся для повышения эффективности создания ПО?
73. Каковы цели управления разработкой ПО?
74. Какие характеристики качества выделяет стандарт ISO 9126:1991?
75. Перечислите основные характеристики качества ПО?
76. Как происходит оценка качества ПО?
77. Дайте определение «надежность» согласно ГОСТ 13377-75.
78. Какими факторами характеризуется надежность программного средства?
79. Назовите основные факторы, влияющие на надежность программного средства.

80. Опишите основные методы обеспечения надежности программного средства.
81. Какова классификация моделей надежности?
82. От чего зависит выбор модели надежности для расчета показателей надежности?
83. В чем заключается различие между аналитическими и эмпирическими моделями надежности ПС?
84. Объясните основные различия между статическими и динамическими аналитическими моделями.
85. Как можно охарактеризовать понятие «программная документация»?
86. Что представляет собой внешняя и внутренняя программная документация?
87. Дайте определение понятию «единая система программной документации».
88. В чем заключаются основные недостатки единой системы программной документации?
89. Дайте определение понятию «техническое задание»
90. Объясните смысл понятия «документация пользователя».
91. Какими свойствами должна обладать документация пользователя?
Дайте краткую характеристику документации пользователя.

3.1.2. Перечень вопросов к экзамену

1. Понятие программной инженерии. История развития программной инженерии как науки.
2. Понятие процесса в программной инженерии.
3. Классические модели процесса. Основные определения (модель процесса, фаза, вид деятельности)
4. Классические модели процесса. Водопадная модель.
5. Классические модели процесса. Спиральная модель.
6. Рабочий продукт (понятие, назначение).
7. Понятие архитектуры ПО.
8. Язык UML. История развития, назначение.
9. Язык UML. Виды диаграмм (не менее 3-х разных примеров).
10. Конфигурационное управление. Понятие baseline.
11. Системы контроля версий. Назначение, примеры инструментов.
12. Системы контроля версий. Понятие ветвь, операции commit, Rollback, push, pull, merge.
13. Тестирование. Виды тестирования.
14. Метод «случай использования». Назначение метода. Понятие актёра, пример.
15. Объектно-ориентированное программирование (ООП). История развития, основные принципы.
16. Объектно-ориентированное программирование (ООП). Классы и объекты.
17. Объектно-ориентированное программирование (ООП). Конструкторы и деструкторы.
18. Объектно-ориентированное программирование (ООП). Статические члены класса.
19. Объектно-ориентированное программирование (ООП). Константные члены класса.
20. Объектно-ориентированное программирование (ООП). Друзья классов.
21. Объектно-ориентированное программирование (ООП). Статический полиморфизм (перегрузка бинарных и унарных операций).
22. Объектно-ориентированное программирование (ООП). Статический полиморфизм (перегрузка функций).
23. Объектно-ориентированное программирование (ООП). Виды отношений между классами.

24. Объектно-ориентированное программирование (ООП). Наследование.
25. Объектно-ориентированное программирование (ООП). Динамический полиморфизм (виртуальные методы).

3.2. Второй этап (продвинутый уровень)

УМЕТЬ (применять, анализировать, оценивать, синтезировать): уметь использовать изученный материал в конкретных условиях и в новых ситуациях; осуществлять декомпозицию объекта на отдельные элементы и описывать то, как они соотносятся с целым, выявлять структуру объекта изучения; оценивать значение того или иного материала – научно-технической информации, исследовательских данных и т. д.; комбинировать элементы так, чтобы получить целое, обладающее новизной

3.2.1. Тестовые задания

1. Легкость применения программного обеспечения это:
 - а) **характеристики ПО, позволяющие минимизировать усилия пользователя по подготовке сходных данных, применению ПО;**
 - б) отношение уровня услуг, предоставляемых ПО пользователю при заданных условиях, к объему используемых ресурсов;
 - в) характеристики ПО, позволяющие минимизировать усилия по внесению изменений для устранения в нем ошибок и по его модификации.
2. Мобильность программного обеспечения это:
 - а) способность ПО выполнять набор функций, которые удовлетворяют потребности пользователей;
 - б) способность ПС безотказно выполнять определенные функции при заданных условиях в течение заданного периода времени;
 - в) **способность ПО быть перенесенным из одной среды (аппаратного / программного) в другое.**
3. Укажите правильную последовательность этапов при каскадной модели жизненного цикла:
 - а) Определение требований -> Тестирование -> Реализация;
 - б) **Проектирование -> Реализация -> Тестирование;**
 - в) Проектирование -> Определение требований -> Реализация.
4. Устойчивость программного обеспечения — это:
 - а) свойство, характеризующее способность ПС завершать автоматически корректное функционирование ПК, несмотря на неправильные (ошибочные) входные данные;
 - б) **свойство, характеризующее способность противостоять преднамеренным или непреднамеренным деструктивным действиям пользователя;**
 - в) свойство, характеризующее способность ПС продолжать корректное функционирование, несмотря на неправильные (ошибочные) входные данные.
5. UML — это:
 - а) язык программирования, имеющий синтаксис схож с C ++;
 - б) **унифицированный язык визуального моделирования, использует нотацию диаграмм;**
 - в) набор стандартов и спецификаций качества программного обеспечения.
6. При конструировании программного обеспечения процесс решения задачи составляет
 - а) 90 — 95%;
 - б) 50%;
 - в) **5 — 10%.**

7. При конструировании программного обеспечения на этапе разработки или выбора алгоритма решения реализуется следующее:
 - а) архитектурная обработка программы;
 - б) выбор языка программирования;**
 - в) совершенствование программы.
8. Проектирование ПО в основном рассматривается как
 - а) архитектурное проектирование;**
 - б) коммуникационные методы;
 - в) детальные методы.
9. На этапе тестирования пользователь выполняет следующее:
 - а) синтаксическую отладку;
 - б) выбор тестов и метода тестирования;**
 - в) определение формы выдачи результатов.
10. Что из приведенного не является одним из методов проектирования программного обеспечения?
 - а) структурное программирование;
 - б) объектно-ориентированное программирование;
 - в) алгебраическое программирование.**
11. Как называется процесс разбиения одной сложной задачи на несколько простых подзадач?
 - а) абстракция;
 - б) декомпозиция;**
 - в) реинжиниринг.
12. Что из приведенного является критериями оценки удобства интерфейсов?
 - а) скорость обучения;
 - б) адаптация к стилю работы пользователя;
 - в) все ответы правильные.**
13. Интерфейс пользователя — это
 - а) набор методов взаимодействия компьютерной программы и пользователя этой программы;**
 - б) набор методов для взаимодействия между программами;
 - в) способ взаимодействия между объектами.
14. Интерфейс-это
 - а) прежде всего, набор правил;
 - б) набор задач пользователя, которые он решает с помощью системы;
 - в) способ взаимодействия между объектами.**
15. Техническое задание — это
 - а) документ объяснений для заказчика;
 - б) исходный документ для сдачи ПО в эксплуатацию;
 - в) выходной документ для проектирования, разработки автоматизированной системы.**
16. Анализ требований —
 - а) отображение функций системы и ее ограничений в модели проблемы;**
 - б) показатель сопровождаемости, который определяет необходимые усилия для диагностики случаев отказов;
 - в) отображение частей программ, которые будут модифицироваться.
17. Архитектура программной системы —
 - а) а) декомпозиция решения для выделенного спектра задач домена на подсистемы или иерархию подсистем;
 - б) б) определение системы в терминах вычислительных составляющих (подсистем) и интерфейсов между ними, которое отражает правила декомпозиции проблемы на составляющие;**
 - в) в) соответствующие вариации состава выделенных компонент.
18. Агрегация —

- а) отношения, утверждает наличие связи между понятиями, не уточняя зависимости их содержания и объемов;
 - б) возможность для некоторого класса находиться одновременно в связи с одним элементом из определенного множества классов;
 - в) объединение нескольких понятий в новое понятие, существенные признаки нового понятия при этом могут быть либо суммой компонент, либо существенно новыми (отношение «доля — целое»).**
19. Ассоциация —
- а) возможность для некоторого класса находиться одновременно в связи с одним элементом из определенного множества классов;
 - б) объединение нескольких понятий в новое понятие, существенные признаки нового понятия о этом могут быть либо суммой компонент или существенно новыми (отношение «доля — целое»);
 - в) самое общее отношение, утверждает наличие связи между понятиями, не уточняя зависимости их содержания и объемов. +**
20. Валидация —
- а) обеспечение соответствия разработки требованиям ее заказчиков.**
 - б) проверка правильности трансформации проекта в код реализации;
 - в) выявление всех ошибок.
21. Верификация —
- а) обеспечение соответствия разработки требованиям ее заказчиков;
 - б) проверка правильности трансформации проекта в программу;**
 - в) действия на каждой стадии жизненного цикла с проверки и подтверждения соответствия стандартам.
22. Внешние метрики продукта:
- а) метрики надежности;**
 - б) метрики размера;
 - в) метрики сложности.
23. Внутренние метрики продукта:
- а) метрики сопровождения;
 - б) метрики годности;
 - в) метрики стиля.**
24. Продукты инженерии требований по методу С.Шлеер и С.Меллора:
- а) информационная модель системы;**
 - б) описание интерфейсов сценариев и актеров;
 - в) неформальное описание сценариев и актеров.
25. К процессу разработки ПО относятся следующие процессы:
- а) сопровождения;
 - б) проектирование;**
 - в) эксплуатация.
26. Последовательность работ по каскадной моделью:
- а) требования, проектирование, реализация;**
 - б) проектирование, сопровождение, тестирование;
 - в) требования, сопровождение, тестирование.
27. Проектирование —
- а) преобразование требований в последовательность проектных решений по системе;**
 - б) определение главных структурных особенностей системы;
 - в) определение подробностей функционирования и связей для всех компонент системы.
28. Модель жизненного цикла —
- а) определение определенных действий, которые сопровождают изменения состояний объектов;
 - б) типичная схема последовательности работ на этапах разработки программного продукта;**

- в) отражение динамики изменений состояния каждого класса объектов.
29. Понятность — это
- а) атрибут функциональности, указывающий на возможность предотвращать несанкционированный доступ;
 - б) атрибут надежности, который указывает на способность программы к перезапуску для повторного выполнения;
 - в) атрибут удобства, определяющий усилия, необходимые для распознавания логических концепций и условий их применения.**
30. Артефакт — это
- а) любой продукт деятельности специалистов по разработке программного обеспечения;**
 - б) результат ошибок разработчика во входных или проектных спецификациях;
 - в) графическое представление элементов моделирования системы.

3.2.2. Темы рефератов

1. Стандарты прикладных протоколов общего назначения.
2. Стандарты базовых сетевых протоколов и сервисов.
3. Спецификации специальных сервисных элементов прикладного уровня модели OSI и стандартов Internet.
4. Базовые спецификации POSIX, определяющие функциональность прикладных программных интерфейсов (API) операционной системы.
5. Профили, играющие роль нормативно - методических документов на государственном уровне или на уровне отдельных отраслей и организаций.
6. Многоуровневая модель системы стандартов ИТ.
7. Оценка трудоемкости и сроков разработки ПО.
8. Главные риски программных проектов и способы реагирования.
9. Планирование управления рисками программных проектов.
10. Парадигмы программирования
11. Концепция жизни в окружении цифровых устройств
12. Облачные технологии
13. Автоматизация тестирования
14. Беспроводные децентрализованные самоорганизующиеся сети мобильных устройств
15. Разработка и эксплуатация безопасных систем электронного обучения
16. Частное облако
17. Утечка исходных кодов коммерческого ПО: примеры и последствия
18. Метод верификации программного обеспечения "Model checking"
19. командные методологии управления разработкой проектов
20. Технологии Data Mining

3.3 Третий этап (высокий уровень)

ВЛАДЕТЬ навыками по применению теоретических и практических знаний и умений при решении ситуационных задач, практической направленности по дисциплине.

3.3.1. Ситуационные задачи

1. Разработать класс многоугольник. Определить методы для создания объектов, перемещения на плоскости, определения периметра, изменения/вывода параметров.
2. Создать класс Man (человек), с полями: имя, возраст, пол и вес. Определить методы задания имени, возраста и веса. Создать производный класс Student, имеющий поле года обучения. Определить методы задания и увеличения года обучения.
3. Разработать класс, представляющий строку, размер которой определяется в момент создания объекта. Определить методы: записи символов в строку; добавления символов в конец строки; вывода всей строки на экран.
4. Разработать класс для представления комплексных чисел с возможностью задания вещественной и мнимой частей как числами типов double, так и целыми числами. Обеспечить выполнение операций сложения, вычитания и умножения комплексных чисел.
5. Разработать класс объектов-векторов, задаваемых координатами концов в трехмерном пространстве. Определить операции сложения и вычитания векторов.
6. Разработать класс окружность. Определить методы для создания объектов, перемещения на плоскости, определения площади, изменения/вывода параметров. В программе создать 3 окружности и рассчитать их суммарную площадь.
7. Разработать класс, содержащий информацию о почтовом адресе организации. Предусмотреть возможность отдельного изменения/вывода составных частей адреса, создания и уничтожения объектов этого класса.
8. Создать класс "Матрица", имеющий поля: двумерный массив вещественных чисел, количество строк и столбцов в матрице. Реализовать методы: сложение с другой матрицей, умножение на число.
9. Класс "Односвязный список". Объявите класс, который реализует односвязный рекурсивный список строк в свободной памяти.
10. Класс "Прямоугольник". Объявите класс "Прямоугольник" с полями: x1, y1, x2, y2 (координаты левого верхнего и правого нижнего углов) и методами: пересечься с другим прямоугольником, проверить, попадает ли точка в данный прямоугольник.
11. Класс "Вектор". Объявите класс "Вектор", полем которого является массив чисел в свободной памяти, а методами - очистить вектор, добавить элемент в конец, вставить элемент в i-ю позицию, удалить i-й элемент, вывести на экран.
12. Класс "Комплексное число". Объявите класс "Комплексное число", полями которого являются действительная и мнимая части числа, а методами - сложение и умножение на другое комплексное число, определение модуля и вывод на экран.
13. Определите классы: "Фигура", "Прямоугольник", "Треугольник", "Круг". Фигура должна быть абстрактным классом. Этот класс хранит координаты x и y - расположения фигуры. Поведение (Методы) класса должны быть заданы интерфейсом.
14. Создать статический метод, который преобразует строку в вещественное число. Успешному преобразованию не должны мешать никакие символы, находящиеся в строке до и после записи числа.

3.4. Представления оценочного средства в фонде

3.4.1. Пример экзаменационного билета

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 1

1. Теоретический вопрос:

Понятие программной инженерии. История развития программной инженерии как науки.

2 Тестирование

1. Легкость применения программного обеспечения это:
 - а) **характеристики ПО, позволяющие минимизировать усилия пользователя по подготовке сходных данных, применению ПО;**
 - б) отношение уровня услуг, предоставляемых ПО пользователю при заданных условиях, к объему используемых ресурсов;
 - в) характеристики ПО, позволяющие минимизировать усилия по внесению изменений для устранения в нем ошибок и по его модификации.
 2. Мобильность программного обеспечения это:
 - а) способность ПО выполнять набор функций, которые удовлетворяют потребности пользователей;
 - б) способность ПС безотказно выполнять определенные функции при заданных условиях в течение заданного периода времени;
 - в) **способность ПО быть перенесенным из одной среды (аппаратного / программного) в другое.**
 3. Укажите правильную последовательность этапов при каскадной модели жизненного цикла:
 - а) Определение требований -> Тестирование -> Реализация;
 - б) **Проектирование -> Реализация -> Тестирование;**
 - в) Проектирование -> Определение требований -> Реализация.
 4. Устойчивость программного обеспечения — это:
 - а) свойство, характеризующее способность ПС завершать автоматически корректное функционирование ПК, несмотря на неправильные (ошибочные) входные данные;
 - б) **свойство, характеризующее способность противостоять преднамеренным или непреднамеренным деструктивным действиям пользователя;**
 - в) свойство, характеризующее способность ПС продолжать
- корректное функционирование, несмотря на неправильные (ошибочные) входные данные.
5. UML — это:
 - а) язык программирования, имеющий синтаксис схож с C ++;
 - б) **унифицированный язык визуального моделирования, использует нотацию диаграмм;**
 - в) набор стандартов и спецификаций качества программного обеспечения

3. Ситуационная задача

Разработать класс многоугольник. Определить методы для создания объектов, перемещения на плоскости, определения периметра, изменения/вывода параметров.

Критерии оценки:

Отлично

Хорошо

Удовлетворительно

Неудовлетворительно

3.4.2. Вопросы для устного опроса (собеседование)

Наименование раздела: Модуль 1

1. Что такое программный продукт и его основные характеристики? Составляющие стоимости ПО.
2. Программная инженерия и ее отличия от информатики и других инженерий?
3. В чем еще отличие от других инженерий?
4. Программный процесс и модель программного процесса. Методы программной инженерии. Что такое CASE системы?
5. Что такое хорошая программа и ее основные свойства? Основные трудности и проблемы программной инженерии
6. Профессиональные и этические требования ИТ-специалиста. Кодекс этики IEEE-CS/ACM.
7. Технология, стандарт и сертификация. Роль стандартов в программной инженерии.
8. Основные стандарты программной инженерии и кто их разрабатывает?
9. Жизненный цикл программного продукта. Процесс, действие, задача жизненного цикла. Фазы (этапы) жизненного цикла и их связь с процессами.
10. Основные процессы жизненного цикла ПО (ISO12207 и ISO 15504)
11. Вспомогательные процессы жизненного цикла ПО (ISO12207 и ISO 15504)
12. Организационные процессы жизненного цикла ПО (ISO12207 и ISO 15504)
13. Каскадная модель ЖЦ ПО. Преимущества, недостатки, применимость
14. Спиральная модель ЖЦ ПО. Преимущества, недостатки, применимость
15. Обзор других типов моделей ЖЦ ПО

Наименование раздела: Модуль 2

1. Особенности моделей жизненного цикла MSF, RUP, XP.
2. Что такое проект и его основные характеристики. Непроекты и их связь с проектами.
3. Управление и управление проектами. Категории управления проектами.
4. Особенности управления ИТ-проектами. Треугольник ограничений проекта.
5. РМВОК: девять областей управленческих знаний.
6. 34 компетенции менеджера IT проекта..
7. Ролевая модель команды. Роли и их ответственности.
8. Модель управления командой. Критерии выбора модели.
9. Административная модель, модель хаоса, модель открытой архитектуры. Особенности, преимущества и недостатки.

10. Роль и способы общения в команде. Преимущества и недостатки различных способов общения.
11. Чем компромисс отличается от консенсуса? Как достичь компромисса и добиться консенсуса?
12. Корпоративная политика. Типы внешних стратегий команд.
13. Что такое качество и мера качества? Какова мера качества программного продукта?
14. Основные фазы эволюции методов обеспечения качества. Роль стандартов в обеспечении качества.
15. ISO9000. 8 принципов TQM и их краткая характеристика

3.4.3. Пример ситуационной задачи (или задачи)

Задание:

Разработать класс многоугольник. Определить методы для создания объектов, перемещения на плоскости, определения периметра, изменения/вывода параметров.

3.5. Критериев оценивания контрольных заданий для использования в ФОС дисциплины

3.5.1. Критерии оценивания тестового задания:

Тестовые задания оцениваются по шкале: 1 балл за правильный ответ, 0 баллов за неправильный ответ. Итоговая оценка по тесту формируется путем суммирования набранных баллов и отнесения их к общему количеству вопросов в задании. Помножив полученное значение на 100%, можно привести итоговую оценку к традиционной следующим образом:

Процент правильных ответов Оценка

90 – 100% *От 9 до 10 баллов и/или «отлично»*

70 – 89 % *От 6 до 8 баллов и/или «хорошо»*

50 – 69 % *От 3 до 5 баллов и/или «удовлетворительно»*

менее 50 % *От 0 до 2 баллов и/или «неудовлетворительно»*

3.5.2. Критерии оценивания реферата (доклада):

От 4 до 5 баллов и/или «отлично»: глубокое и хорошо аргументированное обоснование темы; четкая формулировка и понимание изучаемой проблемы; широкое и правильное использование относящейся к теме литературы и примененных аналитических методов; содержание исследования и ход защиты указывают на наличие навыков работы студента в данной области; оформление работы хорошее с наличием расширенной библиографии; защита реферата (или выступление с докладом) показала высокий уровень профессиональной подготовленности студента;

От 2 до 3 баллов и/или «хорошо»: аргументированное обоснование темы; четкая формулировка и понимание изучаемой проблемы; использование ограниченного, но достаточного для проведения исследования количества источников; работа основана на среднем по глубине анализе изучаемой проблемы и при этом сделано незначительное число обобщений; содержание исследования и ход защиты (или выступление с докладом) указывают на наличие практических навыков работы студента в данной области; реферат (или доклад)

хорошо оформлен с наличием необходимой библиографии; ход защиты реферата (или выступления с докладом) показал достаточную профессиональную подготовку студента;

От 1 до 2 баллов и/или «удовлетворительно»: достаточное обоснование выбранной темы, но отсутствует глубокое понимание рассматриваемой проблемы; в библиографии преобладают ссылки на стандартные литературные источники; труды, необходимые для всестороннего изучения проблемы, использованы в ограниченном объеме; заметна нехватка компетентности студента в данной области знаний; оформление реферата (или доклада) содержит небрежности; защита реферата (или выступление с докладом) показала удовлетворительную профессиональную подготовку студента;

0 баллов и/или «неудовлетворительно»: тема реферата (или доклада) представлена в общем виде; ограниченное число использованных литературных источников; шаблонное изложение материала; суждения по исследуемой проблеме не всегда компетентны; неточности и неверные выводы по рассматриваемой литературе; оформление реферата (или доклада) с элементами заметных отступлений от общих требований; во время защиты (или выступления с докладом) студентом проявлена ограниченная профессиональная эрудиция.

3.5.3. Критерии оценивания на ситуационную задачу:

От 9 до 10 баллов и/или «отлично»: студент глубоко и полно владеет методами решения задачи; решение выполнено оптимальным способом; полученное решение соответствует условиям задачи; решение ситуационной задачи носит самостоятельный характер.

От 6 до 8 баллов и/или «хорошо»: решение студента соответствует указанным выше критериям, но в ход решения имеет отдельные неточности (несущественные ошибки); однако допущенные при решении ошибки исправляются самим студентом после дополнительных вопросов.

От 3 до 5 баллов и/или «удовлетворительно»: студент обнаруживает отсутствие навыков и понимание основных методик решения ситуационной задачи, но решение является неполным, имеет неточности и существенные ошибки; допущенные при решении ошибки не исправляются самим студентом после дополнительных вопросов.

От 0 до 2 баллов и/или «неудовлетворительно»: студент имеет разрозненные, бессистемные знания в области решаемой задачи; не владеет методами и подходами для решения задачи.

3.5.4. Критерии оценки для устного опроса

- оценка «зачтено» выставляется студенту, если обладает систематизированными знаниями, умениями и навыками по данному разделу дисциплины;

- оценка «не зачтено» выставляется студенту, если он не проявил систематизированных знаний, умений и навыков по данному разделу дисциплины.

3.5.5. Критерии оценивания на экзамене:

От 26 до 30 баллов и/или «отлично»: студент глубоко и полно владеет содержанием учебного материала и понятийным аппаратом; умеет связывать теорию с практикой, иллюстрировать примерами, фактами, данными научных исследований; осуществляет межпредметные связи, предложения, выводы; логично, показывает глубокие знания при ответах

на поставленные вопросы; умеет обосновывать свои суждения и профессионально-личностную позицию по излагаемому вопросу; ответ носит самостоятельный характер.

От 16 до 25 баллов и/или «хорошо»: ответ студента соответствует указанным выше критериям, но в содержании имеют место отдельные неточности (несущественные ошибки) при изложении теоретического и практического материала; ответ отличается меньшей обстоятельностью, глубиной, обоснованностью и полнотой; однако допущенные ошибки исправляются самим студентом после дополнительных вопросов экзаменатора.

От 6 до 15 баллов и/или «удовлетворительно»: студент обнаруживает знание, умения и навыки основных положений учебного материала, но излагает его неполно, непоследовательно, допускает неточности и существенные ошибки в определении понятий, формулировке положений; при аргументации ответа студент не опирается на основные положения исследовательских документов; не применяет теоретические знания, умения и навыки для объяснения эмпирических фактов и явлений, не обосновывает свои суждения; имеет место нарушение логики изложения; в целом ответ отличается низким уровнем самостоятельности, не содержит собственной профессионально-личностной позиции.

От 0 до 5 баллов и/или «неудовлетворительно»: студент имеет разрозненные, бессистемные знания, умения и навыки; не умеет выделять главное и второстепенное; в ответе допускаются ошибки в определении понятий, формулировке теоретических положений, искажающие их смысл; студент беспорядочно и неуверенно излагает материал; не умеет соединять теоретические положения с практикой; не владеет навыками и методами решения ситуационных задач.

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

Процедура оценки знаний, умений и навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций, производится преподавателем в форме текущего контроля и промежуточной аттестации.

Для повышения эффективности текущего контроля и последующей промежуточной аттестации студентов осуществляется структурирование дисциплины на модули. Каждый модуль учебной дисциплины включает в себя изучение законченного раздела, части дисциплины.

Основными видами текущего контроля знаний, умений и навыков в течение каждого модуля учебной дисциплины являются устный опрос, тестирование, решение ситуационных задач, подготовка рефератов. Студент должен выполнить все контрольные мероприятия, предусмотренные в модуле учебной дисциплины к указанному сроку, после чего преподаватель проставляет балльные оценки, набранные студентом по результатам текущего контроля модуля учебной дисциплины.

Контрольное мероприятие считается выполненным, если за него студент получил оценку в баллах, не ниже минимальной оценки, установленной программой дисциплины по данному мероприятию.

Промежуточная аттестация обучающихся проводится в форме экзамена.

Экзамен проводится в устной или письменной форме по утвержденным билетам. Каждый билет содержит по два вопроса, и третьего, вопроса или задачи, или практического задания.

Первый вопрос в экзаменационном билете - вопрос для оценки уровня обученности «знать», в котором очевиден способ решения, усвоенный студентом при изучении дисциплины.

Второй вопрос для оценки уровня обученности «знать» и «уметь», который позволяет оценить не только знания по дисциплине, но и умения ими пользоваться при решении стандартных типовых задач.

Третий вопрос (задача/задание) для оценки уровня обученности «владеть», содержание которого предполагает использование комплекса умений и навыков, для того, чтобы обучающийся мог самостоятельно сконструировать способ решения, комбинируя известные ему способы и привлекая имеющиеся знания.

По итогам сдачи экзамена выставляется оценка.

Критерии оценки знаний обучающихся на экзамене:

- оценка «отлично» выставляется, если обучающийся обладает глубокими и прочными знаниями программного материала; при ответе на все вопросы билета продемонстрировал исчерпывающее, последовательное и логически стройное изложение; правильно сформулировал понятия и закономерности по вопросам; использовал примеры из дополнительной литературы и практики; сделал вывод по излагаемому материалу;
- оценка «хорошо» выставляется, если обучающийся обладает достаточно полным знанием программного материала; его ответ представляет грамотное изложение учебного материала по существу; отсутствуют существенные неточности в формулировании понятий; правильно применены теоретические положения, подтвержденные примерами; сделан вывод; два первых вопроса билета освещены полностью, а третий доводится до логического завершения после наводящих вопросов преподавателя;
- оценка «удовлетворительно» выставляется, если обучающийся имеет общие знания основного материала без усвоения некоторых существенных положений; формулирует основные понятия с некоторой неточностью; затрудняется в приведении примеров, подтверждающих теоретические положения; все вопросы билета начаты и при помощи наводящих вопросов преподавателя доводятся до конца;
- оценка «неудовлетворительно» выставляется, если обучающийся не знает значительную часть программного материала; допустил существенные ошибки в процессе изложения; не умеет выделить главное и сделать вывод; приводит ошибочные определения; ни один вопрос билета не рассмотрен до конца, даже при помощи наводящих вопросов преподавателя.

Основным методом оценки знаний, умений и навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций является балльно-рейтинговая система, которая регламентируется положением «О балльно-рейтинговой системе оценки качества освоения образовательных программ в ФГБОУ ВО Белгородский ГАУ».

Основными видами поэтапного контроля результатов обучения студентов являются: входной контроль, текущий контроль, рубежный (промежуточный) контроль, творческий контроль, выходной контроль (экзамен или вопросы к зачету).

Уровень развития компетенций оценивается с помощью рейтинговых баллов.

Рейтинги	Характеристика рейтингов	Максимум баллов
Входной	Отражает степень подготовленности студента к изучению дисциплины. Определяется по итогам входного контроля знаний на первом практическом занятии.	5
Рубежный	Отражает работу студента на протяжении всего периода изучения дисциплины. Определяется суммой баллов, которые студент получит по результатам изучения каждого модуля.	60
Творческий	Результат выполнения студентом индивидуального творческого задания различных уровней сложности, в том числе, участие в различных конференциях и конкурсах на протяжении всего курса изучения дисциплины.	5
Выходной	Является результатом аттестации на окончательном этапе изучения дисциплины по итогам сдачи экзамена. Отражает уровень освоения информационно-теоретического компонента в целом и основ практической деятельности в частности.	30
Общий рейтинг	Определяется путём суммирования всех рейтингов	100

Общий рейтинг по дисциплине складывается из входного, рубежного, выходного (экзамена или вопросы к зачету) и творческого рейтинга.

Входной (стартовый) рейтинг – результат входного контроля, проводимого с целью проверки исходного уровня подготовленности студента и оценки его соответствия предъявляемым требованиям для изучения данной дисциплины.

Он проводится на первом занятии при переходе к изучению дисциплины (курса, раздела). Оптимальные формы и методы входного контроля: тестирование, программированный опрос, в т.ч. с применением ПЭВМ и ТСО, решение комплексных и расчетно-графических задач и др.

Рубежный рейтинг – результат рубежного (промежуточного) контроля по каждому модулю дисциплины, проводимого с целью оценки уровня знаний, умений и навыков студента по результатам изучения модуля. Оптимальные формы и методы рубежного контроля: устные собеседования, письменные контрольные опросы, в т.ч. с использованием ПЭВМ и ТСО, результаты выполнения лабораторных и практических заданий. В качестве практических

заданий могут выступать крупные части (этапы) курсовой работы или проекта, расчетно-графические задания, микропроекты и т.п.

Выходной рейтинг – результат аттестации на окончательном этапе изучения дисциплины по итогам сдачи экзамена, проводимого с целью проверки освоения информационно-теоретического компонента в целом и основ практической деятельности в частности. Оптимальные формы и методы выходного контроля: письменные экзаменационные или контрольные работы, индивидуальные собеседования.

Творческий рейтинг – составная часть общего рейтинга дисциплины, представляет собой результат выполнения студентом индивидуального творческого задания различных уровней сложности.

В рамках рейтинговой системы контроля успеваемости студентов, семестровая составляющая балльной оценки по дисциплине формируется при наборе заданной в программе дисциплины суммы баллов, получаемых студентом при текущем контроле в процессе освоения модулей учебной дисциплины в течение семестра.

Итоговая оценка /зачёта/ компетенций студента осуществляется путём автоматического перевода баллов общего рейтинга в стандартные оценки.

Максимальная сумма рейтинговых баллов по учебной дисциплине составляет 100 баллов.

По дисциплине с экзаменом необходимо использовать следующую шкалу пересчета суммарного количества набранных баллов в четырехбалльную систему:

Неудовлетворительно	Удовлетворительно	Хорошо	Отлично
менее 51 балла	51-67 баллов	68-85 баллов	86-100 баллов