

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ  
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«СЕВЕРО-КАВКАЗСКАЯ ГОСУДАРСТВЕННАЯ АКАДЕМИЯ»

«УТВЕРЖДАЮ»

Проректор по учебной работе

« 30 » 03

2022 г.

Г.Ю. Нагорная



**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ**

Системы программирования

Уровень образовательной программы бакалавриат

Направление подготовки 01.03.02 Прикладная математика и информатика

Направленность (профиль) Прикладная математика и информатика

Форма обучения очная

Срок освоения ОП 4 года

Институт Прикладной математики и информационных технологий

Кафедра разработчик РПД Математика

Выпускающая кафедра Математика

Начальник учебно-методического управления \_\_\_\_\_ Семенова Л.У.

Директор института ПМ и ИТ \_\_\_\_\_ Тебурев Д.Б.

Заведующий выпускающей кафедрой \_\_\_\_\_ Кочкаров А.М.

г. Черкесск, 2022 г.

## СОДЕРЖАНИЕ

<b>1. Цели освоения дисциплины</b> .....	4
<b>2. Место дисциплины в структуре образовательной программы</b> .....	5
<b>3. Планируемые результаты обучения по дисциплине</b> .....	6
<b>4. Структура и содержание дисциплины</b> .....	7
4.1. Объем дисциплины и виды работы.....	7
4.2. Содержание дисциплины .....	8
4.2.1. Разделы (темы) дисциплины, виды учебной деятельности и формы контро- ля.....	8
4.2.2. Лекционный курс.....	9
4.2.3. Лабораторный практикум .....	9
4.2.4. Практические занятия .....	11
4.3. Самостоятельная работа обучающегося.....	12
<b>5. Перечень учебно-методического обеспечения для самостоятельной работы обу- чающихся по дисциплине</b> .....	13
<b>6. Образовательные технологии</b> .....	15
<b>7. Учебно-методическое и информационное обеспечение дисциплины</b> .....	16
7.1. Перечень основной и дополнительной учебной литературы.....	16
7.2. Перечень ресурсов информационно-телекоммуникационной сети «Интернет».....	16
7.3. Информационные технологии, лицензионное программное обеспечение .....	17
<b>8. Материально-техническое обеспечение дисциплины</b> .....	18
8.1. Требования к аудиториям (помещениям, местам) для проведения занятий.....	18
8.2. Требования к оборудованию рабочих мест преподавателя и обучающихся.....	18
8.3. Требования к специализированному оборудованию.....	18
<b>9. Особенности реализации дисциплины для инвалидов и лиц с ограниченными возможностями здоровья</b> .....	19
<b>Приложение 1. Фонд оценочных средств</b> .....	20
<b>Приложение 2. Аннотация дисциплины</b> .....	36
<b>Рецензия на рабочую программу</b> .....	37
<b>Лист переутверждения рабочей программы дисциплины</b> .....	38

## 1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

**Целью** освоения дисциплины «Системы программирования» является ознакомление обучающихся с основными инструментариями, предназначенными для работы с программами на одном из языков программирования, разработка системного и прикладного программного обеспечения осуществляется с помощью систем программирования, в состав которых входят: трансляторы с языков высокого уровня; средства редактирования, компоновки и загрузки программ; макроассемблеры (машинно-ориентированные языки);

### **Задачи дисциплины:**

- освоение обучающимися основных принципов и методов системы программирования;
- изучение средств редактирования, компоновки и загрузки программ;
- изучение программно-аппаратных методов и средств системы программирования;
- изучение машинно-ориентированных языков, в частности ассемблер;
- осуществлять разработку кода программного модуля на современных языках программирования;
  - создавать программу по разработанному алгоритму как отдельный модуль;
  - выполнять отладку и тестирование программы на уровне модуля

## 2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

2.1. Дисциплина «**Системы программирования**» относится к части, формируемой участниками образовательных отношений Блока 1. Дисциплины (модули) выбираемые обучающимися, имеет тесную связь с другими дисциплинами.

2.2. Ниже приведены предшествующие и последующие дисциплины, направленные на формирование компетенций дисциплины в соответствии с матрицей компетенций ОП.

### Предшествующие и последующие дисциплины, направленные на формирование компетенций

№ п/п	Предшествующие дисциплины	Последующие дисциплины
1	<b>Основы информатики</b>	Базы данных
2	<b>Архитектура ЭВМ</b>	
3	<b>Операционные системы и сетевые технологии</b>	

### 3. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ

Планируемые результаты освоения образовательной программы (ОП) – компетенции обучающихся определяются требованиями стандарта по направлению подготовки 01.03.02 Прикладная математика и информатика и формируются в соответствии с матрицей компетенций ОП

№ п/п	Номер/индекс компетенции	Наименование компетенции (или ее части)	В результате изучения дисциплины обучающиеся должны:
1	2	3	4
1.	ПК-3	Способен осуществлять программную реализацию алгоритмов при разработке прикладных информационных систем	<p>ПК-3.1 Обладает знаниями в области программирования и современных компьютерных технологий при разработке прикладных информационных систем. Способен осуществлять оптимизацию алгоритмов при создании прикладных программ или прикладных комплексов</p> <p>ПК-3.2 Анализирует стандартные технические требования к проектированию программного обеспечения.</p> <p>ПК-3.3 Способен осуществлять разработку технических требований к программному обеспечению, при его создании</p>

## 4. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

### 4.1. Объем дисциплины и виды работы

Очная форма обучения

Вид учебной работы		Семестр	
		Всего часов	№4, часов
1		2	3
<b>Аудиторная контактная работа (всего)</b>		<b>54</b>	<b>54</b>
В том числе:			
Лекции (Л)			
Лабораторные работы (ЛР)		<b>36</b>	<b>36</b>
Практические занятия (ПЗ)		<b>18</b>	<b>18</b>
<b>Контактная внеаудиторная работа, в том числе</b>		<b>1,7</b>	<b>1,7</b>
Индивидуальные и групповые консультации			
<b>Самостоятельная работа обучающегося (СРО) (всего)</b>		<b>52</b>	<b>52</b>
<i>Подготовка к лабораторным работам (ЛР)</i>		16	16
<i>Подготовка к практическим работам (ПР)</i>		16	16
<i>Подготовка к выполнению контрольной работы</i>		8	8
<i>Работа с книжными и электронными источниками</i>		10	10
<i>Подготовка к промежуточному тестовому контролю</i>		2	2
<b>Промежуточная аттестация</b>	Зачет, в том числе	3	3
	Прием зачета, час	0,3	0,3
	СРО, час.		
<b>ИТОГО:</b> <b>Общая трудоемкость</b>	<b>часов</b>	<b>108</b>	<b>108</b>
	<b>зач. ед.</b>	<b>3</b>	<b>3</b>

## 4.2. Содержание дисциплины

### 4.2.1. Разделы (темы) дисциплины, виды учебной деятельности и формы контроля

Очная форма обучения

№ п/п	Наименование раздела (темы) дисциплины	Виды учебной деятельности, включая самостоятельную работу обучающихся (в часах)					Формы текущей и промежуточной аттестации	
		Л	ЛР	ПЗ	СРО	все го		
1	2	3	4	5	6	7	8	
<b>Семестр 4</b>								
1.	Раздел 1. Системы программирования для разработки новых программ на конкретном языке программирования.		6	4	10	20	Текущий тестовый контроль, контрольные вопросы, контрольная работа	
2.	Раздел 2. Компиляторы и интерпретаторы. Интегрированная среда разработки программ		10	4	20	34	Текущий тестовый контроль, контрольные вопросы	
3.	Раздел 3. Обширные библиотеки стандартных программ и функций. Отладочные программы, т. е. программы, помогающие находить и устранять ошибки в программе.		10	4	20	34	Текущий тестовый контроль, контрольные вопросы, контрольная работа	
4.	Раздел 4. Многооконный режим работы. Графические библиотеки и утилиты для работы с библиотеками.		10	6	2	18	Текущий тестовый контроль, контрольные вопросы, контрольная работа	
5	Контактная внеаудиторная работа, в том числе индивидуальные и групповые консультации					1,7	индивидуальные и групповые консультации	
	<b>Промежуточная аттестация</b>					<b>0,3</b>	<b>Зачет</b>	
<b>ИТОГО часов в 4 семестре:</b>				<b>36</b>	<b>18</b>	<b>52</b>	<b>108</b>	
<b>Всего:</b>				<b>36</b>	<b>18</b>	<b>52</b>	<b>108</b>	

#### 4.2.2. Лекционный курс не предусмотрен

#### 4.2.3. Лабораторный практикум

№ п/п	Наименование раздела дисциплины	Наименование лабораторной работы	Содержание лабораторной работы	Всего часов
1	2	3	4	5
<b>Семестр 4</b>				
1.	Раздел 1. Системы программирования для разработки новых программ на конкретном языке программирования.	1.1 Изучение систем программирования Основные этапы решения задачи	Постановка задачи. Задача формулируется на естественном языке. Разработка математической модели. Выбор численного метода решения задачи. Разработка алгоритма. Составление программы на выбранном языке программирования.	2
		1.2 Тестирование и отладка программы.	Отладка – это устранение ошибок. Тестирование- проверка правильности работы. Разработка тестов и контрольных примеров. Оценка полученных результатов. Сопоставление реальных и ожидаемых результатов. Если результаты неудовлетворительны, возврат к некоторым предыдущим этапам.	2
		1.3 Описание программы. Разработка инструкций пользователю	Текстовое описание программы. Разработка инструкций пользователю – лицу, применяющему разработанную программу в своей работе. Жизненным циклом программы называют весь период ее разработки и эксплуатации.	2
2.	Раздел 2. Компиляторы и интерпретаторы. Интегрированная среда разработки программ	2.1 Компиляторы и интерпретаторы	Написать программу на языке программирования Фортран и разобраться с принципом работы компилятора. Компиляторы, это программы, которые преобразуют исходные тексты программ, написанные на	10

			языке программирования высокого уровня, в программу на машинном языке, «понятную» компьютеру.	
3.	Раздел 3. Обширные библиотеки стандартных программ и функций. Отладочные программы, т. е. программы, помогающие находить и устранять ошибки в программе.	3.1 Программы на языках программирования	Написать программу на языке программирования Турбо-Паскаль и разобраться с принципом работы интерпретатора. Интерпретаторы также конвертируют высокоуровневые языки в читаемые машиной бинарные эквиваленты.	10
4.	Раздел 4. Многооконный режим работы. Графические библиотеки и утилиты для работы с библиотеками.	4.1 Работа с библиотеками программ и с утилитами	Графические библиотеки и утилиты для работы с библиотеками программ и подпрограмм	10
<b>ИТОГО часов в 4 семестре:</b>				<b>36</b>
<b>Всего:</b>				<b>36</b>

#### 4.2.3. Практические занятия

№ п/п	Наименование раздела дисциплины	Наименование практического занятия	Содержание практического занятия	Всего часов
1	2	3	4	5
<b>Семестр 4</b>				

1.	Раздел 1. Системы программирования для разработки новых программ на конкретном языке программирования.	1.1 Способы описания алгоритмов	1. Детальное описание одного короткого алгоритма программы методом блок-схемы. Алгоритм линейной, разветвляющейся и циклической структур.	2
		1.2 Разработка блок-схемы для описания алгоритмов. Разработка псевдокода для описания алгоритмов	2. Разработать блок-схему для больших алгоритмов с учетом того, что блок-схема получается громоздкой. Выделение некоторых частей и. отрегулировать переход с одной страницы на другую. Представить псевдокод как один из способов точного описания алгоритма предложениями на естественном языке.	2
2	Раздел 2. Компиляторы и интерпретаторы. Интегрированная среда разработки программ	2.1 Текстовые редакторы, служащие для создания текстов исходных программ	<p>1. На примере одной программы продемонстрировать работу компилятора, т.е. получить машинный код, зафиксировать объектные модули, что являются программами на машинном языке.</p> <p>2. Найти на жестком диске C/ файл с расширением *.exe.</p> <p>3. Компоновщики, позволяющие объединять несколько объектных модулей, порождаемых компилятором, в одну программу. В результате создается загрузочный модуль, найти его на C/. 4. Загрузчики, обеспечивающие подготовку готовой программы к выполнению.</p>	4

3	Раздел 3. Обширные библиотеки стандартных программ и функций. Отладочные программы	3.1 Работа с библиотекой стандартных программ	Программы, помогающие находить и устранять ошибки в программе применить к одной программе. Ознакомиться с Библиотеки прикладных программ, содержащие в себе наиболее часто используемые подпрограммы в виде готовых объектных модулей.	4
4	Раздел 4. Многооконный режим работы. Графические библиотеки и утилиты для работы с библиотеками.	4.1 Технология многооконного режима работы.	Ознакомиться с отладчиком, выполняющие программу в заданном режиме с целью поиска, обнаружения и локализации ошибок в графической библиотеке.	6
<b>ИТОГО часов в 4 семестре:</b>				<b>18</b>
<b>Всего:</b>				<b>18</b>

#### 4.3. Самостоятельная работа обучающегося

№ п/п	Наименование раздела (темы) дисциплины	№ п/п	Виды СРО	Всего Часов
1	3	4	5	6
<b>Семестр 4</b>				
1.	Раздел 1. Системы программирования для разработки новых программ на конкретном языке программирования.	1.1	Работа с основной и дополнительной литературой. Подготовка к лабораторному практикуму. Подготовка к выполнению контрольной работы	10
2.	Раздел 2. Компиляторы и интерпретаторы. Интегрированная среда разработки программ	2.1	Подготовка к лабораторному практикуму. Выполнения индивидуальных заданий по лабораторному практикуму	20
3.	Раздел 3. Обширные библиотеки стандартных программ и функций. Отладочные программы.	3.1	Изучение теоретического материала для выполнения практической работы Изучение теоретического материала для выполнения лабораторных работ. Подготовка к выполнению контрольной работы	20
4.	Раздел 4. Многооконный режим работы. Графические библиотеки и утилиты для работы с библиотеками.	4.1	Работа с книжными и электронными источниками. Выполнения индивидуальных заданий по лабораторному практикуму.	2
<b>ИТОГО часов:</b>				<b>52</b>
<b>Всего:</b>				<b>52</b>

## **5. ПЕРЕЧЕНЬ УЧЕБНО-МЕТОДИЧЕСКОГО ОБЕСПЕЧЕНИЯ ДЛЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ**

### **5.1. Методические указания для подготовки обучающихся к лекционным занятиям (не предусмотрено)**

### **5.2. Методические указания для подготовки обучающихся к лабораторным занятиям**

Лабораторные работы сопровождают и поддерживают лекционный курс.

Обучающийся должен подготовиться к выполнению лабораторных работ строго в соответствии с содержанием курса.

В начале каждого лабораторного занятия обучающийся должен ознакомиться с теоретическим материалом, необходимым для выполнения текущей лабораторной работы.

Подготовить ответы на контрольные вопросы, которые соответствуют теме лабораторной работы.

Каждая лабораторная работа содержит список индивидуальных заданий, на выполнение которых обучающийся должен быть готовым.

Полученные результаты выполнения лабораторной работы обучающийся должен уметь оформить и быть готовым к устной защите.

### **5.3. Методические указания для подготовки обучающихся к практическим занятиям**

В ходе подготовки к практическим занятиям, обучающийся должен:

- изучить основную и дополнительную литературу, ознакомиться с новыми публикациями в периодических и электронных изданиях по теме практического занятия учебной программы;

- подготовиться к устному опросу, для этого подготовить ответы по всем учебным вопросам, выносимым на практическое занятие;

- подготовиться к выполнению текущего практического занятия, изучая соответствующую тему лекции;

- после прохождения каждого раздела, обучающийся, для целей закрепления пройденного материала проходит тесты;

- осуществляет самоконтроль качества подготовки к практическому занятию, осуществляя проверку своих знаний, отвечая на вопросы для самопроверки по соответствующей теме.

- обучающийся, при подготовке к практическому занятию может консультироваться с преподавателем через электронную почту и получать от него наводящие разъяснения.

### **5.4 Методические указания к выполнению контрольной работы**

Контрольной работе как одной из форм самостоятельной учебно-исследовательской работы отводится особая роль при формировании компетенции будущего специалиста и бакалавра. Здесь обучающийся демонстрирует применение полученных знаний для создания приложений, решающих конкретные поставленные перед ним задачи. Обучающийся предъявляет преподавателю несколько версий программ, как правило, в электронном виде и получает от преподавателя положительное заключение о результате, либо замечания и предложения по корректировке программы. Программа должна предъявляться в виде, допускающем быстрый переход к ее компиляции, т.е. не допускается передача в виде изображения. Принимаются любые варианты программы, решающие исходную задачу. Преподаватель при приеме приводящей к верному результату программы зачитывает ее как исполненную, но может дать рекомендации по ее улучшению.

## **5.5. Методические указания по самостоятельной работе обучающихся**

Самостоятельная работа обучающихся включает в себя выполнение различного рода заданий, которые ориентированы на более глубокое усвоение материала изучаемой дисциплины. По каждой теме дисциплины обучающимся предлагается перечень заданий для самостоятельной работы. К выполнению заданий для самостоятельной работы предъявляются следующие требования: задания должны исполняться самостоятельно и представляться в установленный срок, а также соответствовать установленным требованиям по оформлению. Обучающимся следует:

- руководствоваться графиком самостоятельной работы, определенным на кафедре;
- выполнять все плановые задания, выдаваемые преподавателем для самостоятельного выполнения, и разбирать неясные вопросы на лабораторных и практических занятиях, а также получить информацию на консультациях.

### **Работа с литературными источниками и интернет ресурсами**

В процессе подготовки к практическим занятиям, студентам необходимо обратить особое внимание на самостоятельное изучение рекомендованной учебно-методической (а также научной и популярной) литературы.

Самостоятельная работа с учебниками, учебными пособиями, научной, справочной и популярной литературой, материалами периодических изданий и Интернета, статистическими данными является наиболее эффективным методом получения знаний, позволяет значительно активизировать процесс овладения информацией, способствует более глубокому усвоению изучаемого материала, формирует у обучающихся свое отношение к конкретной проблеме.

Более глубокому раскрытию вопросов способствует знакомство с дополнительной литературой, рекомендованной преподавателем по каждой теме семинарского или практического занятия, что позволяет студентам проявить свою индивидуальность в рамках выступления на данных занятиях, выявить широкий спектр мнений по изучаемой проблеме.

### **Промежуточная аттестация**

По итогам 4 семестра проводится зачет. При подготовке к сдаче зачета рекомендуется пользоваться материалами практических занятий и материалами, изученными в ходе текущей самостоятельной работы.

## 6. ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ

№ п/п	Виды учебной работы	Образовательные технологии	Всего часов
1	2	3	4
<b>Семестр 4</b>			
1	1.1 Способы описания алгоритмов	Индивидуальная работа обучающихся по заданиям лабораторного практикума с ПЭВМ. Тренинг.	4
2	1.2 Разработка блок схемы для описания алгоритмов	Самостоятельная работа обучающихся по индивидуальным заданиям практикума с ПЭВМ. Тренинг.	4
3	1.3 Разработка псевдокода для описания алгоритмов	Индивидуальная работа обучающихся по заданиям практикума с ПЭВМ. Метод мозгового штурма.	4
4	2.1 Текстовые редакторы, служащие для создания текстов исходных программ	Практическая работа обучающихся по индивидуальным заданиям практической работы с ПЭВМ. Тренинг	4
5	3.1 Работа с библиотекой стандартных программ	Практическая работа обучающихся по индивидуальным заданиям практической работы с ПЭВМ. Тренинг	4
6	4.1 Многооконный режим работы, графические библиотеки	Практическая работа с использованием компьютерных антивирусных программ. Тренинг	4
<b>Итого часов в 4 семестре:</b>			<b>24</b>
<b>Всего:</b>			<b>24</b>

## 7. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

### 7.1 Перечень основной и дополнительной учебной литературы

#### Список основной литературы

1. Кучунова, Е. В. Программирование. Процедурное программирование : учебное пособие / Е. В. Кучунова, Б. В. Олейников, О. М. Чередниченко. — Красноярск : Сибирский федеральный университет, 2016. — 92 с. — ISBN 978-5-7638-3555-7. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/84094.html>
2. Маркин, А. В. Web-программирование : учебник / А. В. Маркин. — Москва : Ай Пи Ар Медиа, 2021. — 286 с. — ISBN 978-5-4497-1002-4. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/104883.html>
3. Ачкасов, В. Ю. Введение в программирование на Delphi : учебное пособие / В. Ю. Ачкасов. — 3-е изд. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2021. — 294 с. — ISBN 978-5-4497-0882-. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/101997.html>

#### Список дополнительной литературы

1. Гунько, А. В. Программирование : учебно-методическое пособие / А. В. Гунько. — Новосибирск : Новосибирский государственный технический университет, 2019. — 74 с. — ISBN 978-5-7782-3961-6. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/98810.html>
2. Гунько, А. В. Программирование (в среде Windows) : учебное пособие / А. В. Гунько. — Новосибирск : Новосибирский государственный технический университет, 2019. — 155 с. — ISBN 978-5-7782-3890-9. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/99209.html>

### 7.2. Перечень ресурсов информационно-телекоммуникационной сети «Интернет»

<http://window.edu.ru>- Единое окно доступа к образовательным ресурсам;

<http://fcior.edu.ru> - Федеральный центр информационно-образовательных ресурсов;

<http://elibrary.ru> - Научная электронная библиотека.

### 7.3. Информационные технологии, лицензионное программное обеспечение

Лицензионное программное обеспечение	Реквизиты лицензий/ договоров
Microsoft Azure Dev Tools for Teaching 1. Windows 7, 8, 8.1, 10	Идентификатор подписчика: 1203743421 Срок действия: 30.06.2022

	(продление подписки)
MS Office 2003, 2007, 2010, 2013	Сведения об Open Office: 63143487, 63321452, 64026734, 6416302, 64344172, 64394739, 64468661, 64489816, 64537893, 64563149, 64990070, 65615073 Лицензия бессрочная
Антивирус Dr.Web Desktop Security Suite	Лицензионный сертификат Серийный № 8DVG-V96F-H8S7-NRBC Срок действия: с 20.10.2022 до 22.10.2023
Цифровой образовательный ресурс IPRs-mart	Лицензионный договор № 9368/22П от 11.06.2021 г. Срок действия: с 01.07.2022 до 01.07.2023

**Бесплатное ПО:** Lazarus, Firebird, IBE Expert, Pascal ABC, Python, VBA, Virtual box, Sumatra PDF, 7-Zip, 1С: Предприятие 8.3 Учебная версия.

## **8. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ**

### **8.1. Требования к аудиториям (помещениям, местам) для проведения занятий Учебная аудитория для проведения занятий лекционного типа.**

#### **1. Учебная аудитория для проведения занятий лекционного типа**

Ауд.243 Специализированная мебель:

Кафедра настольная - 1 шт., доска меловая – 1 шт., стулья – 65 шт., парты – 34 шт.

Технические средства обучения, служащие для предоставления учебной информации большой аудитории:

Экран на штативе – 1 шт.

Проектор – 1 шт.

Ноутбук – 1 шт.

**2. Учебная аудитория для проведения занятий семинарского типа, курсового проектирования (выполнение курсовых работ), групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации**

Ауд.251 Специализированная мебель:

Стол преподавательский – 1 шт., компьютерные столы – 10 шт., парты -7 шт., стулья – 24 шт., доска меловая – 1 шт.

Лабораторное оборудование, технические средства обучения, служащие для предоставления учебной информации большой аудитории:

Персональный компьютер – 10 шт.

Экран настенный рулонный – 1 шт.

#### **3. Лаборатория компьютерной графики**

Ауд.251 Специализированная мебель:

Стол преподавательский - 1 шт., компьютерные столы – 10 шт., парты -7 шт., стулья – 24 шт., доска меловая – 1 шт.

Лабораторное оборудование, технические средства обучения, служащие для предоставления учебной информации большой аудитории:

Персональный компьютер – 10 шт.

Экран настенный рулонный – 1 шт.

### **8.2. Требования к оборудованию рабочих мест преподавателя и обучающихся**

1. Рабочее место преподавателя, оснащенное компьютером.

2. Рабочие места обучающихся оснащенные компьютером.

### **8.3. Требования к специализированному оборудованию - нет**

## **9. ОСОБЕННОСТИ РЕАЛИЗАЦИИ ДИСЦИПЛИНЫ ДЛЯ ИНВАЛИДОВ И ЛИЦ С ОГРАНИЧЕННЫМИ ВОЗМОЖНОСТЯМИ ЗДОРОВЬЯ**

Для обеспечения образования инвалидов и обучающихся с ограниченными возможностями здоровья разрабатывается (в случае необходимости) адаптированная образовательная программа, индивидуальный учебный план с учетом особенностей их психофизического развития и состояния здоровья, в частности применяется индивидуальный подход к освоению дисциплины, индивидуальные задания: рефераты, письменные работы и, наоборот, только устные ответы и диалоги, индивидуальные консультации, использование диктофона и других записывающих средств для воспроизведения лекционного и семинарского материала.

В целях обеспечения обучающихся инвалидов и лиц с ограниченными возможностями здоровья комплектуется фонд основной учебной литературой, адаптированной к ограничению электронных образовательных ресурсов, доступ к которым организован в БИЦ Академии. В библиотеке проводятся индивидуальные консультации для данной категории пользователей, оказывается помощь в регистрации и использовании сетевых и локальных электронных образовательных ресурсов, предоставляются места в читальном зале.

**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ**

**ПО ДИСЦИПЛИНЕ Системы программирования**

# 1. ПАСПОРТ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ ПО ДИСЦИПЛИНЕ

## Системы программирования

### Компетенции, формируемые в процессе изучения дисциплины

Индекс	Формулировка компетенции
ПК-3	Способен осуществлять программную реализацию алгоритмов при разработке прикладных информационных систем

### 2. Этапы формирования компетенции в процессе освоения дисциплины

Основными этапами формирования указанных компетенций при изучении обучающимися дисциплины являются последовательное изучение содержательно связанных между собой разделов (тем) учебных занятий. Изучение каждого раздела (темы) предполагает овладение обучающимися необходимыми компетенциями. Результат аттестации обучающихся на различных этапах формирования компетенций показывает уровень освоения компетенций обучающимися.

Этапность формирования компетенций прямо связана с местом дисциплины в образовательной программе.

Разделы (темы) дисциплины	Формируемые компетенции (коды)
	ПК-3
Раздел 1. Системы программирования для разработки новых программ на конкретном языке программирования.	+
Раздел 2. Компиляторы и интерпретаторы. Интегрированная среда разработки программ	+
Раздел 3. Обширные библиотеки стандартных программ и функций. Отладочные программы.	+
Раздел 4. Многооконный режим работы. Мощные графические библиотеки и утилиты для работы с библиотеками.	+

### 3. Показатели, критерии и средства оценивания компетенций, формируемых в процессе изучения дисциплины

#### ПК- 3 Способен осуществлять программную реализацию алгоритмов при разработке прикладных информационных систем

Индикаторы достижения компетенций	Критерии оценивания результатов обучения				Средства оценивания результатов обучения	
	неудовлетв	удовлетв	хорошо	отлично	Текущий контроль	Промежуточная аттестация
ПК-3.1 Обладает знаниями в области программирования и современных компьютерных технологий при разработке прикладных информационных систем. Способен осуществлять оптимизацию алгоритмов при создании прикладных программ или прикладных комплексов	Допускает существенные ошибки в знаниях стандартных пакетов прикладных программ для решения практических задач защиты информации, отлаживать и тестировать программы по алгоритмам шифрования; пробелы в знаниях элементов теории чисел.	Демонстрирует некоторые способности в знаниях стандартных пакетов прикладных программ для решения практических задач защиты информации, отлаживать и тестировать программы по алгоритмам шифрования, а также криптографическим методам и средств обеспечения информационной безопасности.	Демонстрирует способность в знаниях стандартных пакетов прикладных программ для решения практических задач защиты информации, отлаживать и тестировать программы по алгоритмам шифрования, а также криптографическим методам и средств обеспечения информационной безопасности	Демонстрирует профессиональную способность к языкам программирования, для целей автоматизации шифрования информации; методов кодирования; знания о государственной политике в области безопасности компьютерных систем; знания элементов теории чисел, составляющие математическую основу криптографических методов.	Текущий тестовый контроль, контрольные вопросы, контрольная работа	Зачет
ПК-3.2 Анализирует стандартные технические требования к проектированию программного обеспечения.	Допускает частично освоенное умение пользоваться стандартными программами для решения практических задач защиты информации; отлаживать и тестировать программы по алгоритмам шифрования; применять криптографические методы и средства.	Демонстрирует в целом удовлетворительные, но не систематизированные умения пользоваться стандартными программами для решения практических задач защиты информации; отлаживать и тестировать программы по алгоритмам шифрования; применять криптографические методы чисел для моделирования задач защиты ин-	Демонстрирует в целом хорошие, но содержащие отдельные пробелы в использовании стандартных программ для решения практических задач защиты информации; отлаживать и тестировать программы по алгоритмам шифрования; применять элементы теории чисел для моделирования задач криптографии.	Демонстрирует умения в использовании стандартных программ для решения практических задач защиты информации; отлаживать и тестировать программы по алгоритмам шифрования; применять элементы теории чисел для задач криптографических методов и средств обеспечения информационной безопасности.	Текущий тестовый контроль, контрольные вопросы, контрольная работа	Зачет

		формации.				
ПК-3.3 Способен осуществлять разработку технических требований к программному обеспечению, при его создании	Фрагментарно владеет некоторыми пакетами стандартных программ для решения задач защиты информации; не владеет интернет технологиями для организации защиты информации, не владеет возможностями математической основы криптографии теории чисел.	Владеет отдельными навыками работы со стандартными программами для решения задач защиты информации; технологиями отладки и тестирования программ по алгоритмам шифрования; элементов теории чисел для моделирования задач защиты информации.	Демонстрирует в целом успешные знания по стандартным программам для решения практических задач защиты информации; технологиями отладки и тестирования программ по алгоритмам шифрования; по криптографическим методам и средствам обеспечения информационной безопасности.	Демонстрирует профессиональные навыки работы в специализированных стандартных пакетах программ для решения практических задач защиты информации; владеет технологиями отладки и тестирования программ по алгоритмам шифрования; владеет способностью применять элементы теории чисел для решения задач криптографии.	Текущий текущий контроль, контрольные вопросы, контрольная работа	Зачет

## 4. Комплект контрольно-оценочных средств по дисциплине

### Контрольные вопросы

#### по дисциплине Системы программирования

##### Вопросы к разделу 1.

Понятие алгоритма. Свойства алгоритмов.

Формы представления алгоритмов.

Эффективность алгоритмов.

Можно ли считать алгоритмом `while true do x:=2;`

Единицы измерения информации.

Этапы решения задач на ЭВМ.

Можно ли считать программу правильной (неправильной) при успешном(неуспешном) выполнении тестов?

Формы представления числовой информации.

Позиционные системы счисления.

Булевские операции: конъюнкция, дизъюнкция, инверсия.

Таблицы истинности.

Выполнение логических операций над целыми операндами

Эволюция языков программирования.

Разница между компиляторами и интерпретаторами.

Исходный, объектный и загрузочный модули.

##### Вопросы к разделу 2.

Консольные приложения (CLI) и приложения с графическим интерфейсом (GUI).

Интегрированная среда программирования.

Создание приложений CLI в Lazarus.

Создание приложений GUI в Lazarus.

Форма и размещение на ней управляющих элементов. Палитра компонент.

Инспектор объектов.

Компоненты Edit, Label и их основные свойства.

Компонент Button и его основные свойства и события.

Разновидности операции деления в Pascal.

Различия между операторами цикла repeat и while.

Выразить while через repeat.

Выразить repeat через while.

Выразить for через while.

Какие операторы цикла являются универсальными, т. е. позволяющими обойтись без других операторов цикла?

Как округлить результат вычисления (вещественное число) до 3 значащих десятичных цифр?

##### Вопросы к разделу 3.

Какой тип удобен для представления 4-х сторон света (север, восток, юг, запад)?

Как можно представить условие — в будний(выходной) день?

Разновидности файлов в Pascal.

Как дописать строку к текстовому файлу?

Как записать информацию в текстовый файл, очистив его предыдущее содержимое.

Что может привести к неполной записи информации в файл?

Попробуйте упростить выражение  $(abc > 3) \text{ and } (bc > abc) \text{ and } (3 \geq abc)$

Определить значение выражения  $(A \text{ div } 10) * 10$  при целом  $a=25$

Определить значение выражения  $(A / 10) * 10$  при целом  $a=25$

Записать в элементы массива  $a[1..100]$  с четными индексами их номера, увеличенные на 1.

Основные свойства и методы для Tstrings.

Вопросы к разделу 3.

Сравните максимальное количество сравнений при двоичном и последовательном поиске.

Сравните минимальное количество сравнений при двоичном и последовательном поиске.

В чем схожесть и различие алгоритмов пузырька и quick-sort.

Какие минусы имеет использование отсортированных файлов.

#### **Вопросы к разделу 4.**

Какие основные типы БД поддерживаются при работе с Lazarus?

Назовите и кратко охарактеризуйте методы для работы с таблицами dbf.

Как получить значение поля таблицы dbf?

Назовите основные компоненты для работы с БД, поддерживающими SQL/

Каково основное назначение компонент типа Tconnection.

Каково основное назначение компонент типа TSQL.

Что такое транзакция?

Как стартовать транзакцию?

Как подтвердить транзакцию?

Как откатить транзакцию?

Как выполнить запрос на удаление записей?

Как выполнить запрос на чтение записей?

Как перейти к следующей записи?

**Вопросы к зачету**  
**по дисциплине «Системы программирования»**

1. Понятие системы программирования
2. Стадии разработки программ
3. Технология программирования.
4. Отличие программного способа записи алгоритмов от других
1. Уровни языка программирования
2. Достоинства и недостатки у машинных языков
3. Понятие файла. Виртуальная файловая система.
4. Разграничение доступа к файлам.
5. Язык ассемблер, краткая характеристика
6. Стандартные функции
7. Запись арифметических выражений
8. Как записываются логические выражения
9. Для каких целей используются подпрограммы
10. В каких случаях следует применять процедуры
11. Отличие функции от процедуры
12. Способы задания символьной константы
13. Отличие компиляции от интерпретации
14. Типы данных в языках программирования
15. Требования предъявляемые к программам
16. Классификации языков программирования
17. Что такое массив данных в программировании
18. Какие вы знаете виды массивов
19. Алгоритмы управления очередностью обмена информацией с внешними ЗУ
20. Принципы построения систем ввода-вывода
21. Периферийные устройства
22. Организация интерфейсов ввода-вывода
23. Программное управление вводом-выводом в ЭВМ
24. Передача данных (ввод-вывод) с прямым доступом к памяти
25. Архитектура классических ЭВМ
26. Основные типы микропроцессоров
27. Дайте определения понятиям алфавит, синтаксис, семантика.
28. Языки и системы программирования их особенности
29. Объектно-ориентированное программирование
30. Парадигма защитного программирования.
31. Понятие операционной системы, ее назначение и функции.
32. Утилиты операционной системы
33. Доступ к устройствам.
34. Системы счисления, используемые в ЭВМ.
35. Способы хранения целых беззнаковых чисел, чисел со знаком, вещественных чисел.

36. Организация элементов памяти на триггерных схемах
37. Структура АЛУ: сумматоры, регистры, коммутаторы.
38. Статическая память ЭВМ.
39. Динамическая память ЭВМ и процесс регенерации.
40. ПЗУ – память с разными способами записи постоянных данных.
41. Особенности системы прерывания ПЭВМ
42. Защищенный режим в ПЭВМ. Слово состояния программы
43. Классификация систем памяти. Защита памяти
44. Страничная адресация памяти.
45. Сегментная адресация памяти (на примере ПЭВМ).
46. Алгоритмы замещения информации в основной памяти
47. Основные характеристики процессоров.
48. Обзор современных процессоров ведущих мировых производителей.
49. Процессоры нетрадиционной архитектуры. Клеточные и ДНК-процессоры.
50. Нейронные процессоры.
51. Назначение и характеристики вычислительных систем.

## Список лабораторных работ

### по дисциплине «Системы программирования»

#### **Лабораторная работа 1.1 Изучение систем программирования**

Основные этапы решения задачи

Постановка задачи. Задача формулируется на естественном языке. Разработка математической модели. Выбор численного метода решения задачи. Разработка алгоритма. Составление программы на выбранном языке программирования.

#### **Лабораторная работа 2.1**

Тестирование и отладка программы.

Отладка – это устранение ошибок. Тестирование- проверка правильности работы. Разработка тестов и контрольных примеров. Оценка полученных результатов. Сопоставление реальных и ожидаемых результатов. Если результаты неудовлетворительны, возврат к некоторым предыдущим этапам.

#### **Лабораторная работа 2.2**

Описание программы. Разработка инструкций пользователю Текстовое описание программы. Разработка инструкций пользователю – лицу, применяющему разработанную программу в своей работе. Жизненным циклом программы называют весь период ее разработки и эксплуатации.

#### **Лабораторная работа 3.1 Компиляторы и интерпретаторы**

Написать программу на языке программирования Фортран и разобраться с принципом работы компилятора. Компиляторы, это программы, которые преобразуют исходные тексты программ, написанные на языке программирования высокого уровня, в программу на машинном языке, «понятную» компьютеру.

**Лабораторная работа 3.2 Программы на языках программирования** Написать программу на языке программирования Турбо-Паскаль и разобраться с принципом работы интерпретатора.

Интерпретаторы также конвертируют высокоуровневые языки в читаемые машиной бинарные эквиваленты.

**Лабораторная работа 4.1 Работа с библиотеками программ и с утилитами** Графические библиотеки и утилиты для работы с библиотеками программ и подпрограмм

## Перечень практических работ

### по дисциплине «Системы программирования»

**Раздел 1. Системы программирования для разработки новых программ на конкретном языке программирования.**

#### **Практическая работа 1.1. Способы описания алгоритмов**

1. Детальное описание одного короткого алгоритма программы методом блок-схемы. Алгоритм линейной, разветвляющейся и циклической структур.
2. Разработать блок-схему для больших алгоритмов с учетом того, что блок-схема получается громоздкой. Выделение некоторых частей.
- 3.Отрегулировать переход с одной страницы на другую.

4. Представить псевдокод как один из способов точного описания алгоритма предложениями на естественном языке.

**Практическая работа 1.2.** Разработка блок-схемы для описания алгоритмов. Разработка псевдокода для описания алгоритмов

1. Разработать блок-схему для больших алгоритмов с учетом того, что блок-схема получается громоздкой.

2. Выделение некоторых частей и. отрегулировать переход с одной страницы на другую.

3. Представить псевдокод как один из способов точного описания алгоритма предложениями на естественном языке.

**Раздел 2. Компиляторы и интерпретаторы.**

**Практическая работа 2.1** Интегрированная среда разработки программ

Текстовые редакторы, служащие для создания текстов исходных программ. Работа с текстовым редактором

**Раздел 3. Обширные библиотеки стандартных программ и функций.**

**Отладочные программы**

**Практическая работа 3.1** Работа с библиотекой стандартных программ

Программы, помогающие находить и устранять ошибки в программе применить к одной программе. Ознакомиться с Библиотеки прикладных программ, содержащие в себе наиболее часто используемые подпрограммы в виде готовых объектных модулей.

**Раздел 4. Многооконный режим работы.**

**Практическая работа 4.1** Графические библиотеки и утилиты для работы с библиотеками. Технология многооконного режима работы.

Ознакомиться с отладчиком, выполняющие программу в заданном режиме с целью поиска, обнаружения и локализации ошибок в графической библиотеке.

## **Комплект заданий для контрольной работы**

### **по дисциплине «Системы программирования»**

Задание 1. Выполнить разработку программы и теста для решения одного из вариантов предложенных заданий:

1. Вычислить периметр и площадь прямоугольного треугольника, если известны  $a$ ,  $b$ ,  $c$ .

2. Вычислить площадь круга заданного радиуса  $R$ .

3. Даны два числа. Найти среднее арифметическое кубов этих чисел.

4. Даны два действительных числа  $x$  и  $y$ . Вычислить их сумму, разность, произведение и частное.

5. Дана сторона равностороннего треугольника. Найти площадь этого треугольника.

6. Дана сторона равностороннего треугольника. Найти радиус вписанной окружности.

7. Известна длина окружности. Найти площадь круга, ограниченного этой окружностью.

8. Найти площадь шара, радиуса  $R$ .
9. Вычислить площадь треугольника по формуле Герона, если заданы его стороны.

Задание 2. Выполнить компиляцию, отладку и тестирование разработанного приложения.

### **Комплект тестовых вопросов**

#### **по дисциплине «Системы программирования»**

1. Системы для разработки новых программ на конкретном языке программирования:
  - а) системы программирования
  - б) программированные системы
  - в) системы ретуширования
  
2. К какому уровню языков относятся языки ассемблера \_\_\_\_\_
  
  
3. Какой язык программирования был создан в 1979 году и назван в честь первого в мире программиста:
  - а) Паскаль
  - б) Фортран
  - в) Ада
  
4. Для чего используют программы-ассемблеры \_\_\_\_\_
  
  
5. Какой язык называется машинно-ориентированным:
  - а) язык, в основу которого заложены принципы объектно-ориентированного программирования
  - б) любой универсальный язык программирования
  - в) язык, определяющийся набором команд конкретного процессора
  
6. Какой язык программирования, созданный в 1957 году, является одним из первых алгоритмических языков и до сих пор применяется для научных вычислений:
  - а) Паскаль
  - б) Фортран
  - в) Ада
  
7. Что такое системы программирования?
  
  
8. Выберите верное утверждение о языке ассемблер:
  - а) программа, написанная на Ассемблере для одного процессора не будет работать на другом

- б) программы, написанные на языке Ассемблер создаются только в среде Linux
- в) программа, написанная на Ассемблере для одного процессора будет работать на любом другом

9. Как называются формальные языки, созданные для разработки программ:

- а) языки высокого уровня
- б) популярные языки
- в) алгоритмические языки

10. Как называют программы, предназначенные для перевода в машинные коды программы, написанной на языке высокого уровня \_\_\_\_\_

11. Из слов какого языка строятся команды языков программирования высокого уровня:

- а) естественного языка
- б) логического языка
- в) алгоритмического языка

12. Одно из самых удобных средств разработки программ современных систем программирования:

- а) ретранслятор
- б) компилятор или интерпретатор
- в) интерстеллер

13. Одно из самых удобных средств разработки программ современных систем программирования:

- а) интегрированная среда разработки
- б) интригованная среда разработки
- в) интегрирующая среда разработки

14. Одно из самых удобных средств разработки программ современных систем программирования:

- а) тихий режим работы
- б) однооконный режим работы
- в) многооконный режим работы

15. Одно из самых удобных средств разработки программ современных систем программирования:

- а) встроенный ассемблер
- б) встроенный кассемблер
- в) встроенный ассемблятор

16. Одна из популярных систем программирования:

- а) Basic C
- б) Turbo Basic
- в) Basic S

17. Одна из популярных систем программирования:

- а) Slow Basic
- б) Fast Basic
- в) Quick Basic

18. Одна из популярных языков программирования \_\_\_\_\_

20. В последнее время получили распространение системы программирования, ориентированные на создание:

- а) Yandex-приложений
- б) Windows-приложений
- в) Google-приложений

21. Язык Паскаль был разработан в этом году \_\_\_\_\_

22. Язык Си разработан Деннисом Ритчи в этом году:

- а) 1982
- б) 1972
- в) 1985

23. Один из программных компонентов системы программирования:

- а) галерея подпрограмм
- б) регулятор текста
- в) редактор текста

24. Один из программных компонентов системы программирования:

- а) ретранслятор с соответствующего языка
- б) транслятор с соответствующего языка
- в) регулятор соответствующего языка

25. Один из программных компонентов системы программирования:

- а) наладчик
- б) постановщик
- в) компоновщик

26. Один из программных компонентов системы программирования:

- а) доводчик
- б) отладчик
- в) наладчик

27. Один из программных компонентов системы программирования:

- а) библиотеки подпрограмм
- б) библиотеки программ
- в) галерея подпрограмм

28. Программа для ввода и модификации текста:

- а) компоновщик
- б) транслятор
- в) редактор текста

29. Трансляторы делятся на некоторое число классов \_\_\_\_\_

30. Программа, которая позволяет управлять процессом исполнения программы, для поиска и исправления ошибок в программе \_\_\_\_\_

31. В какой аппаратной части современного ПК хранится информация о конфигурации ПК \_\_\_\_\_

## **5. Методические материалы, определяющие процедуры оценивания компетенции**

### **5.1 Критерии оценивания качества выполнения лабораторного практикума**

Оценка **«зачтено»** выставляется обучающемуся, если лабораторная работа выполнена правильно и обучающийся ответил на все вопросы, поставленные преподавателем на защите.

Оценка **«не зачтено»** выставляется обучающемуся, если лабораторная работа выполнена не правильно или обучающийся не проявил глубоких теоретических знаний при защите работы

### **5.2 Критерии оценивания качества выполнения практических работ**

Оценка **«зачтено»** выставляется обучающемуся, если практическая работа выполнена правильно и обучающий ответил на все вопросы, поставленные преподавателем на защите.

Оценка **«не зачтено»** выставляется обучающемуся, если практическая работа выполнена не правильно или обучающий не проявил глубоких теоретических знаний при защите работы .

### **5.3 Критерии оценивания качества устного ответа**

Оценка **«отлично»** выставляется за глубокое знание предусмотренного программой материала, за умение четко, лаконично и логически последовательно отвечать на поставленные вопросы.

Оценка **«хорошо»** – за твердое знание основного (программного) материала, за грамотные, без существенных неточностей ответы на поставленные вопросы.

Оценка **«удовлетворительно»** – за общее знание только основного материала, за ответы, содержащие неточности или слабо аргументированные, с нарушением последовательности изложения материала.

Оценка **«неудовлетворительно»** – за незнание значительной части программного материала, за существенные ошибки в ответах на вопросы, за неумение ориентироваться в материале, за незнание основных понятий дисциплины.

### **5.4 Критерии оценивания тестирования**

При тестировании все верные ответы берутся за 100%.

90%-100% отлично

75%-90% хорошо

60%-75% удовлетворительно

менее 60% неудовлетворительно

### **5.5 Критерии оценивания качества контрольной работы**

Оценка **«отлично»** выставляется за глубокое знание предусмотренного программой материала, за умение четко, лаконично и логически последовательно отвечать на поставленные вопросы.

Оценка **«хорошо»** – за твердое знание основного (программного) материала, за грамотные, без существенных неточностей ответы на поставленные вопросы.

Оценка **«удовлетворительно»** – за общее знание только основного материала, за ответы, содержащие неточности или слабо аргументированные, с нарушением последовательности изложения материала.

Оценка **«неудовлетворительно»** – за незнание значительной части программного материала, за существенные ошибки в ответах на вопросы, за неумение ориентироваться в

материале, за незнание основных понятий дисциплины.

### **5.6 Критерии оценивания результатов освоения дисциплины**

Оценка «**зачтено**» выставляется обучающемуся, если обучающий почти ответил на все вопросы, поставленные преподавателем на защите.

Оценка «**не зачтено**» выставляется обучающемуся, если обучающий не проявил глубоких теоретических знаний при ответе на вопросы.