

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ**

ПЕНЗЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ФАКУЛЬТЕТ ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ



РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

С1.О.11 АЛГОРИТМИЧЕСКИЕ ЯЗЫКИ И ПРОГРАММИРОВАНИЕ НА ЭВМ

Специальность 01.05.01 Фундаментальные математика и механика

Направленность (специализация) Вычислительная математика и вычислительная механика

Квалификация выпускника – Математик. Механик. Преподаватель.

Форма обучения очная

Пенза, 2019

1. Цели освоения дисциплины

Целями освоения дисциплины «Алгоритмические языки и программирование» является приобретение студентами знаний, умений, навыков по программированию на алгоритмических языках высокого уровня для решения задач вычислительной математики и вычислительной механики, овладение студентами профессиональными компетенциями, способствующими их социальной мобильности и устойчивости на рынке труда.

2. Место дисциплины в структуре ОПОП специалиста

Дисциплина «Алгоритмические языки и программирование на ЭВМ» находится в обязательной части Блока 1 «Дисциплины (модули)» ОПОП и является одной из дисциплин, формирующих профессиональные знания, умения и навыки, характерные для специалиста по специальности 01.05.01 «Фундаментальная математика и механика» направленности (специализации) «Вычислительная математика и вычислительная механика».

Изучение данной дисциплины базируется на предметах школьной программы.

Основные положения дисциплины должны быть использованы в дальнейшем при изучении следующих дисциплин:

- разработка математических пакетов программ, суперкомпьютерные вычисления, математические основы компьютерной графики;
- при выполнении и защите выпускной квалификационной работы.

3. Результаты освоения дисциплины «Алгоритмические языки и программирование на ЭВМ»

Процесс изучения дисциплины направлен на формирование элементов следующих компетенций в соответствии с ФГОС ВО по данному направлению:

Коды компетенции	Наименование компетенции	Индикатор достижения компетенции (закрепленный за дисциплиной)	В результате освоения дисциплины обучающийся должен
1	2	3	
ОПК-3	Способен самостоятельно создавать и грамотно использовать прикладные программные средства на основе современных информационных технологий и сетевых ресурсов	ОПК-3.1 Понимает методологию и обладает базовыми знаниями в области современных информационных технологий, сетевых ресурсов, необходимыми для разработки и использования новых прикладных программных средств	Знать: современные информационные технологии и языки программирования, необходимые для разработки и использования прикладных программных средств.
			Уметь: применять современные алгоритмические языки программирования и среды для разработки прикладных программных средств.
			Владеть: основными методами и средствами разработки прикладных программных средств
		ОПК-3.2 Анализирует решаемые задачи, определяет источники и осуществляет поиск информации, выбор подходящих информационных технологий и сетевых ресурсов, необходимых для разработки и использования	Знать: методы анализа решаемых задач для разработки и использования прикладных программных средств.
Уметь: анализировать решаемые задачи, строить эффективные алгоритмы для разработки прикладных программных средств			

		новых прикладных программных средств	Владеть: приемами структурного и объектно-ориентированного программирования для разработки прикладных программных средств.
		ОПК-3.3 Самостоятельно создает и грамотно использует новые прикладные программные средства, используя современные, в т.ч. инновационные информационные технологии и сетевые ресурсы	Знать: принципы создания и требования оформления программного кода.
			Уметь: разрабатывать, отлаживать и тестировать прикладные программные средства.
			Владеть: современными языками программирования и инструментальными средствами для разработки, отладки и тестирования прикладных программных средств.

4. Структура и содержание дисциплины «Алгоритмические языки и программирование на ЭВМ»

4.1. Структура дисциплины (модуля)

Общая трудоемкость дисциплины составляет 10 зачетных единиц, 360 часов.

№ п/п	Наименование разделов и тем дисциплины (модуля)	семестр	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)									Формы текущего контроля успеваемости (по неделям семестра) Форма промежуточной аттестации (по семестрам)					
			Контактная работа				Самостоятельная работа					Защита лабораторных работ	Собеседование	Проверка тестов	Проверка контрольных работ	Проверка реферата	
			Всего	Лекция	Лабораторные занятия	Иная конт. работа	Всего	Подготовка к аудиторным занятиям	Реферат, эссе и др.	Подготовка к тесту	Подготовка к зачету						Подготовка к экзамену
1	Раздел 1. Введение в программирование на языке Free Pascal	1	9	5	4		12	12					1-3				
1.1	Тема 1.1. Программирование: основные понятия и определения. Жизненный цикл и критерии качества программ	1	1	1			4	4					1				
1.2	Тема 1.2 Разработка и отладка консольных приложений в среде визуального программирования Lazarus	1	4	2	2		4	4					2				
1.3	Тема 1.3 Обзор языка программирования Free Pascal	1	4	2	2		4	4					3				
2	Раздел 2. Основы алгоритмизации и структурного программирования на языке Free Pascal	1	10	4	6		10	10					4-6				
2.1	Тема 2.1 Понятие и свойства алгоритма. Структурирование алгоритмов	1	4	2	2		4	4					4				
2.2	Тема 2.2 Реализация управляющих структур в языке Free Pascal	1	6	2	4		6	6					5-6				
3	Раздел 3. Обработка структурированных данных в языке Free Pascal	1	24	6	18		24	24					7-15				
3.1	Тема 3.1 Обработка массивов в языке Free Pascal	1	8	2	6		8	8					7-9				
3.2	Тема 3.2 Обработка строк в языке Free Pascal	1	8	2	6		8	8					10-12				
3.3	Тема 3.3 Обработка записей в языке Free Pascal	1	8	2	6		8	8					13-15				
4	Раздел 4 Подпрограммы в языке Free Pascal	1	8	2	6		8	8					16-17				
4.1	Тема 4.1 Подпрограммы-процедуры и подпрограммы-функции в языке Free Pascal	1	8	2	6		8	8					16-17				
	<i>Иная контактная работа</i>	1	3			3											
	<i>Подготовка к экзамену</i>	1					36					36					
5	Раздел 5 Введение в визуальное программирование в среде Lazarus	2	6	3	3		6	6					1-4				
5.1	Тема 5.1 Элементы графического интерфейса современных приложений	2	2	1			2	2					1				
5.2	Тема 5.2 Разработка графического интерфейса в среде Lazarus	2	2	1	1		2	2					2				
5.3	Тема 5.3 Событийное программирование в среде Lazarus	2	3	1	2		2	2					4				
6	Раздел 6. Разработка приложений с развитым графическим интерфейсом в среде Lazarus	2	12	6	6		6	6					5-9				

4.2. Содержание дисциплины (модуля)

4.2.1. Содержание лекционного курса

Раздел 1. Введение в программирование на языке Free Pascal

Тема 1.1. Программирование: основные понятия и определения. Жизненный цикл и критерии качества программ. Цели, задачи и структура курса, его место в профессиональной подготовке специалистов. Программа = алгоритм + структура данных. Жизненный цикл программ: разработка, эксплуатация, продолжающаяся разработка. Критерии качества программ: документированность, эффективность, простота, мобильность, удобство эксплуатации, стоимость. Описание и спецификация программы.

Тема 1.2 Разработка и отладка консольных приложений в среде визуального программирования Lazarus. Общие сведения среде визуального программирования Lazarus. Обзор возможностей среды визуального программирования Lazarus. Интерфейс среды визуального программирования Lazarus. Установка среды визуального программирования Lazarus под различные операционные системы. Виды приложений, поддерживаемых средой визуального программирования Lazarus. Консольное приложение в среде визуального программирования Lazarus.

Тема 1.3 Обзор языка программирования Free Pascal. Язык программирования – система обозначений для описания алгоритмов и структур данных, которая используется для программирования на компьютере. Императивные (процедурные), аппликативные (функциональные) и логические (основанные на систем правил) языки. Синтаксис и нотации его описания (форма Бэкуса-Наура, синтаксические графы). Семантические правила. Компиляция и интерпретация. История языка Pascal. Общие сведения о языке Free Pascal.

Раздел 2. Основы алгоритмизации и структурного программирования на языке Free Pascal

Тема 2.1 Понятие и свойства алгоритма. Структурирование алгоритмов. Понятие алгоритма. Свойства алгоритма (массовость, однозначность, результативность, детерминированность). Базовые управляющие структуры: последовательность, если_то_иначе, цикл с предусловием. Графическое представление алгоритмов (схемы алгоритмов, диаграммы Насси-Шнейдермана, таблицы решений). Методы структурирования алгоритмов на основе теоремы Джакопини.

Тема 2.2 Реализация управляющих структур в языке Free Pascal. Конструкции языка Free Pascal для реализации управляющих структур (условий, циклов). Оператор присваивания. Операторы условий. Оператор выбора. Операторы циклов.

Раздел 3. Обработка структурированных данных в языке Free Pascal

Тема 3.1 Обработка массивов в языке Free Pascal. Структурированный тип данных «массив» в языке Free Pascal. Статические и динамические массивы. Одномерные и многомерные массивы. Объявление массивов. Обращение к элементам массива. Ввод/вывод массивов. Типовые вычислительные задачи для одномерных и многомерных массивов.

Тема 3.2 Обработка строк в языке Free Pascal. Символьные и строковые типы данных в языке Free Pascal. Объявление строк. Обращение к символам строки как к элементам массива. Ввод/вывод строк. Типовые задачи над строками.

Тема 3.3 Обработка записей в языке Free Pascal. Структурированный тип данных «запись» в языке Free Pascal. Объявление переменных-записей и массивов записей. Обращение к полям записи. Ввод/вывод записей. Типовые задачи для записей.

Раздел 4 Подпрограммы в языке Free Pascal

Тема 4.1 Подпрограммы-процедуры и подпрограммы-функции в языке Free Pascal.

Понятие подпрограммы. Подпрограммы в языке Free Pascal. Подпрограммы-процедуры и подпрограммы функции. Глобальные и локальные данные. Фактические и формальные параметры. Способы передачи параметров (значением и по адресу). Способы передача массивов в качестве параметров. Рекурсивные подпрограммы. Примеры

Раздел 5 Введение в визуальное программирование в среде Lazarus

Тема 5.1 Элементы графического интерфейса современных приложений.

Различия между консольными и графическими приложениями. Понятие и свойства графического интерфейса. Виды пользовательских интерфейсов. Элементы графического интерфейса. Окно. Виды окон. Оконные элементы интерфейса.

Тема 5.2 Разработка графического интерфейса в среде Lazarus. Структура проекта приложения. Правила размещения проекта и именования фалов проекта. Форма-контейнер для размещения компонент и ее свойства. Библиотека компонент. Основные приемы верстки пользовательских компонент на форме.

Тема 5.3 Событийное программирование в среде Lazarus. Основные принципы событийного программирования. Программные модули проекта. Стандартные и специальные события, обрабатываемые на уровне компонент и формы. Порядок обработки событий. Правила написания обработчиков событий. Многократное использование обработчиков событий.

Раздел 6. Разработка приложений с развитым графическим интерфейсом в среде Lazarus

Тема 6.1 Компоненты ввода данных. Организация ввода данных. Однострочные редакторы TEdit, TLabelEdit. Организация контроля ввода данных. Компонент TMaskEdit. Обработка исключений при вводе данных. Специальные компоненты для ввода чисел.

Тема 6.2 Компоненты отображения и выбора данных. Компоненты отображения и выбора данных. Компонент TMemo. Компонент TStringGrid. Компоненты выбора. Компоненты отображения структурированных данных.

Тема 6.3 Многооконные приложения. Добавление в проект нескольких форм. Модальные и немодальные формы. Организация переходов между формами. Ссылка на формы из разных проектов.

Раздел 7 Эффективная разработка надежных приложений в среде Lazarus

Тема 7.1 Обработка исключительных ситуаций. Виды ошибок и исключительных ситуаций. Конструкции для обработки исключительных ситуаций. Способы реагирования на возникновения ошибок и/или исключительных ситуаций в приложении.

Тема 7.2 Разработка приложений с применением механизма действий (Action). Механизм действий. Компонент-действия TActions. Компонент список действий TActionList. Ссылка на объекты-действия из компонент интерфейса.

Тема 7.3 Отладка и тестирование приложений. Основные понятия и показатели надежности программных продуктов.

Раздел 8 Обработка и визуализация табличных данных в среде Lazarus

Тема 8.1 Табличное представление функций. Табличные данные и компонент TStringGrid. Форматированный вывод числовых данных в TStringGrid. Выделение фрагмента таблицы. Выполнение расчетов над выделенным фрагментом.

Тема 8.2 Построение графиков функций. Компоненты для отображения графической информации. Компонент TChart для отображения диаграмм и графиков. Параметры визуализации в TChart. Сохранение графической информации TChart в файл. Загрузка графической информации TChart из файла.

Раздел 9 Введение в программирование на языке C++

Тема 9.1 Обзор языка программирования C++. Семейство языков C. Среды разработки. Структура программы на языке C. Состав языка. Идентификаторы. Ключевые слова. Знаки операций. Константы. Комментарии.

Тема 9.2 Структура программы на языке C++. Программа как набор исполняемых инструкций. Структура основной программы. Виды инструкций. Стейтменты. Выражения. Функции. Библиотеки. Комментарии. Пример простой программы. Синтаксис и синтаксические ошибки.

Тема 9.3 Ввод/вывод в C++. Обзор ввода/вывода данных в C++ Потоки, буферы и файл iostream. Перенаправление. Вывод с помощью cout. Перегруженный оператор <<. Другие методы класса iostream. Очистка буфера вывода. Форматирование вывода с помощью cout. Ввод данных с помощью cin. Ввод/вывод файлов Простая работа с файлами. Текстовые и двоичные файлы. Функции работы с файлом. Режимы файлов. Произвольный доступ к файлам.

Раздел 10 Переменные и типы данных в языке C++

Тема 10.1 Переменные и основные типы данных. Простые переменные. Имена переменных. Целочисленные типы данных. Типы данных short, int и long. Типы данных без знака. Целочисленные константы. Определение типа константы. Тип данных char: символы и малые целые числа. Универсальные имена символов. Новый тип данных bool. Квалификатор const. Числа с плавающей точкой. Запись чисел с плавающей точкой. Типы данных с плавающей точкой. Константы с плавающей точкой. Преимущества и недостатки типов данных с плавающей точкой.

Тема 10.2 Область видимости и расширенные типы данных. Область видимости и пространства имен. Массивы: одномерные, многомерные. Строки. Краткий обзор структур. Свойства структур. Массивы структур. Битовые поля. Объединения Перечисления. Установка значений перечислителей. Диапазоны значений для перечислений.

Раздел 11 Операторы в языке C++

Тема 11.1 Операторы управления порядком исполнения кода в C++. Оператор if Оператор if-else. Форматирование операторов if-else. Конструкция if-else-if-else. Логические выражения. Операция логическое ИЛИ: ||. Операция логическое И: &&. Операция логическое НЕ: !. Оператор ?. Оператор switch. Операторы break и continue. Оператор цикла for. Операторы цикла с постусловием и предусловием. Операторы передачи управления. Обзор возможностей цикла for. Составные части цикла for. Изменение шага цикла. Операторы инкремента (++) и декремента (--). Комбинированные операторы присваивания Составные операторы или блоки. Цикл while. Сравнение циклов for и while. Цикл do while.

Раздел 12 Массивы, строки, указатели и ссылки в языке C++

Тема 12.1 Массивы и строки в C++. Массивы динамической памяти. Краткий обзор массивов. Правила инициализации массивов. Строки. Конкатенация строк. Использование строк в массиве. Смешанный ввод строк и чисел. Доступ к символам строки с помощью цикла for. Сравнение строк.

Тема 12.2 Указатели и ссылки в C++. Указатели и свободная память. Объявление и инициализация указателей. Указатели и числа. Выделение памяти с помощью оператора new. Освобождение памяти с помощью оператора delete. Использование оператора new для создания динамических массивов. Указатели, массивы и арифметика указателей. Динамическое распределение памяти. Функции malloc() и free(). Динамическая память и классы.

Раздел 13 Подпрограммы в языке C++

Тема 13.1 Функции в C++. Обзор функций. Определение функции. Создание прототипов и вызов функций. Аргументы функции и передача по значению. Функции с несколькими аргументами. Функции и массивы. Применение указателей в функциях, обрабатывающих массивы. Особенности использования массивов в качестве аргументов. Указатели и спецификатор const. Рекурсия. Указатели на функции. Назначение указателя на функцию.

Тема 13.2 Встроенные и перегружаемые функции в C++. Встроенные функции. Ключевое слово inline. Перегрузка функций. Типы возврата в перегрузке функций. Псевдонимы типов в перегрузке функций. Вызовы перегруженных функций. Результаты вызова перегруженных функций. Отсутствие вызываемой функции. Одна вызываемая функция. Несколько совпадений.

Тема 14.1 Основы объектно-ориентированного программирования в C++

Тема 14.1 Основы объектно-ориентированного программирования в C++. Основные понятия. Инкапсуляция. Область действия классов Управление доступом к членам класса и функциям класса. Инициализация объектов классов: конструкторы. Использование перегруженных конструкторов. Свойства. Композиция: ссылки на объекты как переменные экземпляра других классов. Использование ссылки this. Доступ к членам и функциям класса. Создание и уничтожение объектов. Операции с объектами. Свойства. Пример класса.

Тема 14.2 Наследование и полиморфизм в C++. Наследование. Базовые и производные классы. Члены protected и internal. Наследование и композиция. Взаимоотношения между базовыми и производными классами. Конструкторы и деструкторы в производных классах. Проектирование программ с наследованием. Полиморфизм. Примеры полиморфизма. Преобразование объекта производного класса в объект базового класса. Абстрактные классы и методы. Создание и использование интерфейсов. Делегаты. Перегрузка операторов Полиморфизм и виртуальные функции. Абстрактные классы.

4.2.2. Перечень и содержание лабораторных занятий

№ п/п	№ темы	Наименование лабораторных работ	Кол. ч
<i>Первый семестр</i>			
1	1.2	Разработка простого консольного приложения в среде Lazarus	2
2	1.3	Разработка консольного приложения в среде Lazarus для решения вычислительной задачи с линейным алгоритмом	2
3	2.1	Разработка консольного приложения в среде Lazarus для решения вычислительной задачи с условным алгоритмом	2
4	2.2	Разработка консольного приложения в среде Lazarus для решения вычислительной задачи с циклическим алгоритмом	4
5	3.1	Разработка консольного приложения в среде Lazarus для решения вычислительной задачи с матрицами	6
6	3.2	Разработка консольного приложения в среде Lazarus для решения задачи со строками	6
7	3.3	Разработка консольного приложения в среде Lazarus для решения задачи со структурированными данными (записями)	6
8	4.1.	Разработка модульного приложения (подпрограмм) в среде Lazarus	6
<i>Итого за первый семестр</i>			<i>34</i>
<i>Второй семестр</i>			
9	5.2-5.3	Разработка простого графического приложения в среде Lazarus	3
10	6.1	Разработка графического приложения в среде Lazarus с контролем ввода/вывода данных	2
11	6.2	Разработка графического приложения в среде Lazarus с компонентами отображения и выбора текстовых данных	2
12	6.3	Разработка многооконного графического приложения в среде Lazarus	2

№ п/п	№ темы	Наименование лабораторных работ	Кол. ч
13	7.1-7.2	Разработка приложения в среде Lazarus с развитым графическим интерфейсом и обработкой исключительных ситуаций	4
14	8.1-8.2	Разработка графического приложения в среде Lazarus для обработки табличных данных и построения графиков функций	4
<i>Итого за второй семестр</i>			<i>17</i>
<i>Третий семестр</i>			
15	9.3	Разработка простого приложения на С++	3
16	10.2	Разработка приложения на С++ для решения вычислительной задачи с линейным алгоритмом	2
17	11.1	Разработка приложения на С++ для решения вычислительной задачи с условным и циклическим алгоритмом	2
18	12.1-12.2	Разработка приложения на С++ для решения вычислительной задачи с матрицами	4
19	13.1-13.2	Разработка модульного приложения (функций) на С++	2
20	14.1-14.2	Разработка объектного приложения на С++	4
<i>Итого за третий семестр</i>			<i>17</i>
<i>Итого</i>			<i>68</i>

5. Образовательные технологии

5.1 В ходе освоения дисциплины при проведении аудиторных занятий используются следующие образовательные технологии: лекции с использованием активных и интерактивных форм проведения занятий, лабораторные работы с применением современных технологий разработки программного обеспечения.

5.2 В целях реализации компетентностного подхода следует использовать в образовательном процессе активные и интерактивные формы проведения учебных занятий (разбор конкретных ситуаций, групповую работу и др.) в сочетании с внеаудиторной работой для формирования и развития общих и профессиональных компетенций обучающихся.

5.3 Итоговая аттестация по дисциплине «Алгоритмические языки и программирование на ЭВМ» проводится в форме зачета и экзамена.

5.4 Консультации для обучающихся проводятся на протяжении всего процесса освоения дисциплины в групповой, индивидуальной, устной форме.

5.5 При организации самостоятельной работы используются следующие образовательные технологии: работа с научной, профессионально-технической и учебно-методической литературой, поиск необходимой информации в сети Интернет, организация встреч студентов с представителями российских компаний - работодателей, посвященных обсуждению перспектив развития информационных систем и технологий.

5.6 В целях реализации индивидуального подхода к обучению студентов, осуществляющих учебный процесс по собственной траектории в рамках индивидуального рабочего плана, изучение данной дисциплины базируется на следующих возможностях: обеспечение внеаудиторной работы со студентами, в том числе в электронной образовательной среде, с использованием соответствующего программного обеспечения, дистанционных форм обучения, возможностей интернет-ресурсов, индивидуальных консультаций и т.д.

5.7 Обучающиеся, из числа инвалидов и лиц с ограниченными возможностями здоровья обеспечены электронными образовательными ресурсами в формах, адаптированных к ограничениям их здоровья. Форма проведения текущей и промежуточной аттестации для студентов-инвалидов устанавливается с учетом индивидуальных психофизических особенностей (устно, письменно на компьютере, в формате тестирования и т.д.) и позволяют оценить достижения ими запланированных в основной образовательной программе результатов обучения и уровня сформированности всех заявленных компетенций. На экзамен приглашается сопровождающий, который обеспечивает техническое сопровождение студенту. При необходимости студенту-

инвалиду предоставляется дополнительное время для подготовки ответа на зачете и экзамене.

6. Учебно-методическое обеспечение самостоятельной работы студентов.

Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины.

Лабораторные работы оцениваются по балльной системе. Экзамены оцениваются по системе: неудовлетворительно, удовлетворительно, хорошо, отлично. В течение каждого семестра студенты изучают разделы, указанные преподавателем, разбирают примеры программ, повторяют основные понятия и приемы программирования, показанные на лекциях. В каждом семестре предусмотрены коллоквиумы и контрольные работы.

6.1. План самостоятельной работы студентов

№ нед.	Тема	Вид самостоятельной работы	Задание	Рекомендуемая литература	Количество часов
<i>Первый семестр</i>					
1	Программирование: основные понятия и определения. Жизненный цикл и критерии качества программ	Изучение конспекта лекций. Изучение дополнительной литературы. Подготовка к лабораторным работам.	Изучить основные понятия и определения программирования, как процесса разработки программных средств. Познакомиться с этапами разработки и критериями качества программ.	[1, 2, 5]	4
2	Разработка и отладка консольных приложений в среде визуального программирования Lazarus	Изучение конспекта лекций. Изучение дополнительной литературы. Подготовка к лабораторным работам.	Изучить приемы разработки и отладки консольных приложений в среде визуального программирования Lazarus	[1, 2, 6]	4
3	Обзор языка программирования Free Pascal	Изучение конспекта лекций. Изучение дополнительной литературы. Подготовка к лабораторным работам.	Изучить основные возможности и конструкции языка программирования Free Pascal	[1, 2, 6]	4
4	Понятие и свойства алгоритма. Структурирование алгоритмов	Изучение конспекта лекций. Изучение дополнительной литературы. Подготовка к лабораторным работам.	Изучить понятие и свойства алгоритмов, способы структурирования алгоритмов.	[1, 2, 6, 7]	4
5-6	Реализация управляющих структур в языке Free Pascal	Изучение конспекта лекций. Изучение дополнительной литературы. Подготовка к лабораторным работам.	Изучить конструкции для реализации управляющих структур в языке Free Pascal	[1, 2]	6
7-9	Обработка массивов в языке	Изучение конспекта лекций. Изучение	Изучить вопросы, связанные с	[1, 2]	8

	Free Pascal	дополнительной литературы. Подготовка к лабораторным работам.	обработкой массивов в языке Free Pascal		
10-12	Обработка строк в языке Free Pascal	Изучение конспекта лекций. Изучение дополнительной литературы. Подготовка к лабораторным работам.	Изучить вопросы, связанные с обработкой строк в языке Free Pascal	[1, 2]	8
13-15	Обработка записей в языке Free Pascal	Изучение конспекта лекций. Изучение дополнительной литературы. Подготовка к лабораторным работам.	Изучить вопросы, связанные с обработкой записей в языке Free Pascal	[1, 2]	8
16-17	Подпрограммы-процедуры и подпрограммы-функции в языке Free Pascal	Изучение конспекта лекций. Изучение дополнительной литературы. Подготовка к лабораторным работам.	Изучить основы модульного программирования и разработки подпрограмм-процедур и подпрограмм-функций в языке Free Pascal	[1, 2]	8
		Подготовка к экзамену	Изучить разделы 5-8		36
<i>Второй семестр</i>					
1	Элементы графического интерфейса современных приложений	Изучение конспекта лекций. Изучение дополнительной литературы. Подготовка к лабораторным работам.	Изучить понятие, свойства и основные элементы графического интерфейса современных приложений	[1, 2, 5]	2
2	Разработка графического интерфейса в среде Lazarus	Изучение конспекта лекций. Изучение дополнительной литературы. Подготовка к лабораторным работам.	Изучить принципы и приемы разработки графического интерфейса в среде Lazarus	[1, 2]	2
3-4	Событийное программирование в среде Lazarus	Изучение конспекта лекций. Изучение дополнительной литературы. Подготовка к лабораторным работам.	Изучить принципы и приемы событийного программирования в среде Lazarus	[1, 2]	2
5-6	Компоненты ввода данных	Изучение конспекта лекций. Изучение дополнительной литературы. Подготовка к лабораторным работам.	Изучить основные компоненты ввода данных из библиотеки компонент в среде Lazarus	[1, 2]	2
7	Компоненты отображения и выбора данных	Изучение конспекта лекций. Изучение дополнительной	Изучить основные компоненты отображения и	[1, 2]	2

		литературы. Подготовка к лабораторным работам.	выбора данных из библиотеки компонент в среде Lazarus		
8-9	Многооконные приложения	Изучение конспекта лекций. Изучение дополнительной литературы. Подготовка к лабораторным работам.	Изучить принципы и способы разработки многооконных приложений	[1, 2]	2
10-11	Обработка исключительных ситуаций	Изучение конспекта лекций. Изучение дополнительной литературы. Подготовка к лабораторным работам.	Изучить принципы и способы обработки исключительных ситуаций	[1, 2]	4
12-13	Разработка приложений с применением механизма действий (Action)	Изучение конспекта лекций. Изучение дополнительной литературы. Подготовка к лабораторным работам.	Изучить принципы и приемы разработки приложений с применением механизма действий (Action)	[1, 2]	4,05
14	Отладка и тестирование приложений	Изучение конспекта лекций. Изучение дополнительной литературы. Подготовка к лабораторным работам.	Изучить принципы и приемы отладки и тестирования приложений	[1, 2]	2
15	Табличное представление функций	Изучение конспекта лекций. Изучение дополнительной литературы. Подготовка к лабораторным работам.	Изучить принципы, практические приемы и визуальные компоненты для табличного представления функций	[1, 2]	2
16	Построение графиков функций	Изучение конспекта лекций. Изучение дополнительной литературы. Подготовка к лабораторным работам.	Изучить принципы, практические приемы и визуальные компоненты для визуального представления графиков функций	[1, 2]	2
17		Подготовка к зачету	Изучить разделы 1-6		10
<i>Третий семестр</i>					
1	Обзор языка программирования С++	Изучение конспекта лекций. Изучение дополнительной литературы. Подготовка к лабораторным работам.	Изучить основные возможности и конструкции языка программирования С++	[3, 4]	2
1	Структура программы на языке С++	Изучение конспекта лекций. Изучение дополнительной литературы. Подготовка к лабораторным работам.	Изучить структуру основной программы на языке С++	[3, 4]	2

1	Ввод/вывод в C++	Изучение конспекта лекций. Изучение дополнительной литературы. Подготовка к лабораторным работам.	Изучить организацию ввода/вывода данных в языке C++	[3, 4]	2
2	Переменные и основные типы данных	Изучение конспекта лекций. Изучение дополнительной литературы. Подготовка к лабораторным работам.	Изучить основные типы данных и способы работы с переменными	[3, 4]	2
2	Область видимости и расширенные типы данных	Изучение конспекта лекций. Изучение дополнительной литературы. Подготовка к лабораторным работам.	Изучить расширенные типы данных и область видимости переменных	[3, 4]	4
3-4	Операторы управления порядком исполнения кода в C++	Изучение конспекта лекций. Изучение дополнительной литературы. Подготовка к лабораторным работам.	Изучить языковые конструкции и приемы работы с операторами, задающими порядок исполнения кода	[3, 4]	4
5-6	Массивы и строки в C++	Изучение конспекта лекций. Изучение дополнительной литературы. Подготовка к лабораторным работам.	Изучить вопросы, связанные с обработкой массивов и строк в языке C++	[3, 4]	4
7-8	Указатели и ссылки в C++	Изучение конспекта лекций. Изучение дополнительной литературы. Подготовка к лабораторным работам.	Изучить вопросы, связанные с обработкой указателей и ссылок в языке C++	[3, 4]	4
9	Функции в C++	Изучение конспекта лекций. Изучение дополнительной литературы. Подготовка к лабораторным работам.	Изучить вопросы, связанные с разработкой функций в языке C++	[3, 4]	2
10	Встроенные и перегружаемые функции в C++	Изучение конспекта лекций. Изучение дополнительной литературы. Подготовка к лабораторным работам.	Изучить вопросы, связанные с использованием встроенных и перегружаемых функций в языке C++	[3, 4]	4
11-13	Основы объектно-ориентированного программирования в C++	Изучение конспекта лекций. Изучение дополнительной литературы. Подготовка к лабораторным работам.	Изучить основы объектно-ориентированного программирования и реализацию классов и объектов в языке C++	[3, 4]	2

		работам.			
14-16	Наследование и полиморфизм в C++	Изучение конспекта лекций. Изучение дополнительной литературы. Подготовка к лабораторным работам.	Изучить реализацию наследования классов и переопределения методов классов в языке C++	[3, 4]	3
		Подготовка к экзамену	Изучить разделы 9-14		36

6.2. Методические указания по организации самостоятельной работы студентов

Студенты получают от преподавателя задание на повторение пройденного материала и самостоятельное изучение дополнительного материала по изучаемым темам лекционного курса. Преподаватель предлагает студентам литературу для самостоятельного изучения, а также выдает лабораторные задания

Самостоятельная работа студентов организуется в компьютерном классе с развернутой ЛВС, имеющей подключение к сети Интернет и обеспечивающей доступ к ресурсам электронного обучения, современным профессиональным базам данных и информационным справочным системам.

Рекомендуется проведение следующих видов самостоятельной работы:

- работа с конспектом лекций и изучение рекомендованной литературы: изучить конспект лекций, изучить разделы рекомендованной литературы, ответить на контрольные вопросы;

- подготовка к лабораторным занятиям: изучить необходимый теоретический материал по теме лабораторного занятия, ответить на контрольные вопросы, разработать и отладить программу на алгоритмическом языке, выполнить тестирование разработанной программы;

- оформление отчётов по лабораторным работам: изучить требования к оформлению отчета, подготовить отчет по лабораторной работе и сформулировать выводы по работе;

- подготовка к зачету и экзамену: повторить материал, изученный в течение семестра.

Студентам из числа лиц с ограниченными возможностями здоровья могут быть предложены электронные образовательные ресурсы в формах, адаптированных к ограничениям их здоровья

6.3. Материалы для проведения текущего и промежуточного контроля знаний студентов

Контроль освоения компетенций

№ п/п	Вид контроля	Контролируемые разделы	Компетенции, компоненты которых контролируются
1	Текущий: собеседование при защите лабораторных работ Промежуточный: экзамен	Раздел 1	ОПК-3
2	Текущий: собеседование при защите лабораторных работ Промежуточный: экзамен	Раздел 2	ОПК-3
3	Текущий: собеседование при защите лабораторных работ Промежуточный: экзамен	Раздел 3	ОПК-3
4	Текущий: собеседование при защите лабораторных работ Промежуточный: экзамен	Раздел 4	ОПК-3
5	Текущий: собеседование при защите	Раздел 5	ОПК-3

	лабораторных работ Промежуточный: зачет		
6	Текущий: собеседование при защите лабораторных работ Промежуточный: зачет	Раздел 6	ОПК-3
7	Текущий: собеседование при защите лабораторных работ Промежуточный: зачет	Раздел 7	ОПК-3
8	Текущий: собеседование при защите лабораторных работ Промежуточный: зачет	Раздел 8	ОПК-3
9	Текущий: собеседование при защите лабораторных работ Промежуточный: экзамен	Раздел 9	ОПК-3
10	Текущий: собеседование при защите лабораторных работ Промежуточный: экзамен	Раздел 10	ОПК-3
11	Текущий: собеседование при защите лабораторных работ Промежуточный: экзамен	Раздел 11	ОПК-3
12	Текущий: собеседование при защите лабораторных работ Промежуточный: экзамен	Раздел 12	ОПК-3
13	Текущий: собеседование при защите лабораторных работ Промежуточный: экзамен	Раздел 13	ОПК-3
14	Текущий: собеседование при защите лабораторных работ Промежуточный: экзамен	Раздел 14	ОПК-3

Контроль освоения компетенции выполняется:

– для компетенции (ОПК-3) – путем оценки способности обучаемого самостоятельно создавать и грамотно использовать прикладные программные средства на основе современных информационных технологий и сетевых ресурсов.

Материалы для проведения текущего контроля знаний и промежуточной аттестации составляют отдельный документ – Фонд оценочных средств по дисциплине «Алгоритмические языки и программирование на ЭВМ».

Демонстрационные варианты оценочных средств для каждого вида контроля доступны в ЭИОС (<http://moodle.pnzgu.ru>) в разделе Оценочные средства по дисциплине в курсе «Алгоритмические языки и программирование на ЭВМ».

Вопросы к экзамену

(1 семестр)

1. Что такое программа?
2. Каковы основные этапы жизненного цикла программы?
3. Назовите критерии качества программ
4. Простая программа. Эквивалентность программ
5. Элементарная и составная программа. Управляющие структуры
6. Теорема о структурировании программ. Помеченная структурированная программа
7. Рекурсивная структурированная программа
8. Основные компоненты
9. Синтаксис. Нотация Бэкуса-Наура. Синтаксические графы.
10. Компиляция и интерпретация программ
11. Типы и значения
12. Константы и переменные
13. Простые типы
14. Структурированные типы. Массив и строка
15. Записи и файлы. Общие схемы создания и использования файлов.
16. Указатели
17. Глобальные и локальные переменные

18. Формальные и фактические параметры. Способы передачи параметров
19. Функции в языке Паскаль
20. Процедуры в языке Паскаль
21. Передача массивов в качестве параметров
22. Процедурный тип в передаче параметров
23. Процедуры и функции работы со строками в языке Паскаль
24. Процедуры и функции работы с файлами в языке Паскаль

Вопросы к зачету

(2 семестр)

1. Графический интерфейс приложений. Элементы графического интерфейса.
2. Основные принципы визуального программирования в среде Lazarus.
3. Понятие и состав проекта в среде Lazarus.
4. Визуальные компоненты в среде Lazarus. Форма и ее основные свойства.
5. Визуальные компоненты в среде Lazarus. Компоненты отображения и редактирования текста (TLabel, TEdit).
6. Визуальные компоненты в среде Lazarus. Кнопки (TButton, TBitBtn и TSpeedButton).
7. Визуальные компоненты в среде Lazarus. Многострочный редактор (TMemo).
8. Визуальные компоненты в среде Lazarus. Табличный редактор (TStringEdit).
9. Обработка событий в среде Lazarus.
10. Отладка и тестирование программ в среде Lazarus.

Практические задания. Разработка алгоритма и программы на языке программирования Pascal в соответствии со следующими темами:

1. Условные и циклические алгоритмы;
2. Обработка одномерных и двумерных массивов;
3. Обработка строк;
4. Работа с множествами;
5. Разработка и вызов подпрограмм (процедур и функций);
6. Работа с записями и файлами.

Вопросы к экзамену

(3 семестр)

- 1 Структура программы на языке C++.
- 2 Состав языка. Идентификаторы. Ключевые слова. Знаки операций. Константы. Комментарии и стили программирования.
- 3 Простые переменные. Имена переменных. Целочисленные типы данных. Типы данных short int и long . Типы данных без знака. Целочисленные константы. Определение типа константы.
- 4 Тип данных char: символы и малые целые числа.
- 5 Тип данных bool.
- 6 Числа с плавающей точкой. Запись чисел с плавающей точкой. Типы данных с плавающей точкой. Константы с плавающей точкой. Преимущества и недостатки типов данных с плавающей точкой.
- 7 Арифметические операции. Присваивание. Приоритет операций и ассоциативность. Разновидности операции деления. Операция деления по модулю.
- 8 Преобразования типов данных. Преобразования данных в выражениях.
- 9 Операции отношение и логические операции.
- 10 Побитовые операции.
- 11 Поточковые операторы ввода/вывода.
- 12 Манипуляторы.
- 13 Оператор if. Оператор if else. Форматирование операторов if else. Конструкция if else if else.
- 14 Логические выражения. Операция логическое ИЛИ: ||. Операция логическое И: &&. Операция логическое НЕ: ! .

- 15 Оператор ? . Оператор switch.
- 16 Оператор цикла for. Операторы цикла с постусловием и предусловием. Обзор возможностей цикла for. Составные части цикла for. Изменение шага цикла. Доступ к символам строки с помощью цикла for. Операторы инкремента (++) и декремента (--).
- 17 Сравнение строк Цикл while. Сравнение циклов for и while. Цикл do while.
- 18 Массивы: одномерные, многомерные. Массивы динамической памяти. Правила инициализации массивов .
- 19 Строки . Конкатенация строк. Использование строк в массиве. Смешанный ввод строк и чисел .
- 20 Краткий обзор структур. Свойства структур. Массивы структур.
- 21 Перечисления. Установка значений перечислителей. Диапазоны значений для перечислений.
- 22 Указатели и свободная память. Объявление и инициализация указателей. Указатели и числа.
- 23 Выделение памяти с помощью оператора new. Освобождение памяти с помощью оператора delete.
- 24 Обзор функций. Определение функции. Создание прототипов и вызов функций. Аргументы функции и передача по значению Функции с несколькими аргументами. Рекурсия. Указатели на функции.
- 25 Динамическое распределение памяти. Функции malloc() и free(). Динамическая память и классы
- 26 Ввода/вывода данных в C++. Текстовые и двоичные файлы. Функции работы с файлом. Потоки и буферы. Вывод с помощью cout.

7. Учебно-методическое и материально-техническое обеспечение дисциплины «Алгоритмические языки и программирование на ЭВМ»

а) учебная литература:

- 1) Гуриков С.Р. Основы алгоритмизации и программирования в среде Lazarus [Электронный ресурс]: учеб. пособие / С.Р. Гуриков. – М.: ФОРУМ : ИНФРА-М, 2019. – 336 с. – (Высшее образование: Бакалавриат). – Режим доступа: <http://znanium.com/catalog/product/1010784>.
- 2) Алексеев Е. Р. Free Pascal и Lazarus [Электронный реурс]: Учебник по программированию / Е. Р. Алексеев, О. В.Чеснокова, Т. В. Кучер. - М.: ALT Linux, 2010. - 438 с.: ил. - (Библиотека ALT Linux). - ISBN 978-5-94074-611-9. - Режим доступа: <http://znanium.com/catalog/product/408666>.
- 3) Культин Н.Б. Основы программирования в Microsoft Visual C++ 2010 [Электронный ресурс]: Самоучитель / Культин Н.Б. - СПб:БХВ-Петербург, 2010. - 384 с. ISBN 978-5-9775-0520-8 - Режим доступа: <http://znanium.com/catalog/product/350852>.
- 4) Литвиненко Н.А. Технология программирования на C++. Win32 API-приложения [Электронный ресурс]: Учебное пособие / Литвиненко Н.А. - СПб:БХВ-Петербург, 2010. - 280 с. ISBN 978-5-9775-0600-7 – Режим доступа: <http://znanium.com/catalog/product/351463>.
- 5) Л.В.Гурьянов, Л.С.Гурьянова, Б.К.Мещеряков, А.Н. Ракова. Основы конструирования программ [Электронный ресурс]: Лабораторный практикум. – Пенза, ПГУ, 2010. – 170 с. – Режим доступа: http://moodle.pnzgu.ru/pluginfile.php/772193/mod_resource/content/1/%D0%9E%D1%81%D0%BD%D0%BE%D0%B2%D1%8B%20%D0%BA%D0%BE%D0%BD%D1%81%D1%82%D1%80%D1%83%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F%20%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC.%D0%9B%D0%B0%D0%B1%D0%BE%D1%80%D0%B0%D1%82%D0%BE%D1%80%D0%BD%D1%8B%D0%B9%20%D0%BF%D1%80%D0%B0%D0%BA%D1%82%D0%B8%D0%BA%D1%83%D0%BC.pdf.

б) Колдаев В.Д. Основы алгоритмизации и программирования [Электронный ресурс]: Учебное пособие / В.Д. Колдаев; Под ред. Л.Г. Гагариной. - М.: ИД ФОРУМ: ИНФРА-М, 2012. - 416 с.: ил.; 60x90 1/16. - (Профессиональное образование). (переплет) ISBN 978-5-8199-0279-0 – Режим доступа: <http://znanium.com/catalog/product/336649>.

7) Кетков Ю.Л. Свободное программное обеспечение. FREE PASCAL для студентов и школьников [Электронный ресурс]: Практическое пособие / Кетков Ю.Л., Кетков А.Ю. - СПб:БХВ-Петербург, 2011. - 378 с. ISBN 978-5-9775-0604-5 - Режим доступа: <http://znanium.com/catalog/product/354800>

б) интернет-ресурсы:

1. Электронная библиотечная система Лань <https://e.lanbook.com>
2. Электронная библиотечная система Знаниум <http://znanium.com>

в) Программное обеспечение:

1) Лицензионное ПО:

- операционная система Microsoft Windows 7 и выше.

2) Свободно распространяемое ПО:


- браузеры Mozilla Firefox, Google Chrome;
- система для просмотра PDF-документов Adobe Acrobat Reader;
- среда визуальной разработки приложений Lazarus 2.0 и выше;
- среда разработки приложений Microsoft Visual Studio C++.

г) Другое материально-техническое обеспечение:

- компьютеры с доступом в сеть Internet для самостоятельной работы.

Рабочая программа дисциплины С1.О.11 «Алгоритмические языки и программирование на ЭВМ» составлена в соответствии с требованиями ФГОС ВО по специальности 01.05.01 «Фундаментальная математика и механика», утвержденного приказом Министерства образования и науки Российской Федерации от «6» февраля 2018 г. № 16.


Программу составили:

1. _____ Шибанов С.В., доцент каф. МО и ПЭВМ 
(Ф.И.О., должность, подпись)

2. _____
(Ф.И.О., должность, подпись)

Настоящая программа не может быть воспроизведена ни в какой форме без предварительного письменного разрешения кафедры-разработчика программы.

Программу составили:

1. _____ Шибанов С.В., доцент каф. МО и ПЭВМ 
(Ф.И.О., должность, подпись)

2. _____
(Ф.И.О., должность, подпись)

Настоящая программа не может быть воспроизведена ни в какой форме без предварительного письменного разрешения кафедры-разработчика программы.

Программа одобрена на заседании кафедры МСМ

Протокол № 11

от « 29 » 06 _____ 2018 года

Зав. кафедрой МСМ

Ю.Г. Смирнов Ю.Г.
(подпись, Ф.И.О.)

Сведения о переутверждении программы на очередной учебный год и регистрации изменений

Учебный год	Решение кафедры (№ протокола, дата)	Внесенные изменения	Подпись зав. кафедрой