


МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ  
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«РЯЗАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ С.А. ЕСЕНИНА»

Утверждаю:  
Декан  
физико-математического  
факультета  
 Н.Б. Федорова  
«30» августа 2019 г.

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ  
«АЛГОРИТМИЗАЦИЯ И ОСНОВЫ ПРОГРАММИРОВАНИЯ»**

Уровень основной профессиональной образовательной программы:  
**бакалавриат**

Направление подготовки: **38.03.05 Бизнес-информатика**

Направленность (профиль) подготовки: **Цифровая экономика**

Форма обучения: **очная**

Срок освоения ОПОП: **нормативный срок освоения 4 года**

Факультет: **физико-математический**

Кафедра: **информатики, вычислительной техники и методики преподавания  
информатики**

Рязань 2019

# ВВОДНАЯ ЧАСТЬ

## 1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Целью освоения учебной дисциплины «Алгоритмизация и основы программирования» является формирование у обучающихся общепрофессиональных компетенций в процессе изучения алгоритмизации и основ программирования для последующего применения в учебной и практической деятельности.

Задачи дисциплины:

- систематизация, формализация и расширение знаний по основам информатики, приобретенные в школе;
- привитие навыков алгоритмического мышления, культуры алгоритмизации и нисходящего структурного программирования;
- формирование теоретической базы и практических умений и навыков для решения задач на компьютере в императивных системах программирования,
- формирование основ современной культуры программирования.

## 2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП ВУЗА

**2.1.** Учебная дисциплина **Б1.Б.17. «Алгоритмизация и основы программирования»** относится к базовой части Блока 1.

**2.2.** Для изучения дисциплины необходимы знания, умения, навыки, формируемые дисциплиной предшествующего уровня образования:

- *Информатика и информационные технологии.*

**2.3.** Перечень последующих учебных дисциплин, для которых необходимы знания, умения, навыки, формируемые данной учебной дисциплиной:

- Программирование;
- Объектно-ориентированное программирование;
- Программирование на встроенном языке информационной системы;
- Кроссплатформенное программирование.

## 2.4. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения основной профессиональной образовательной программы

Изучение данной учебной дисциплины направлено на формирование у обучающихся следующих общепрофессиональных (ОПК) компетенций:

№ п/п	Номер/ индекс компетенции	Содержание компетенции (или ее части)	В результате изучения дисциплины обучающиеся должны		
			Знать	Уметь	Владеть
1.	ОПК-1	Способность решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности.	<ul style="list-style-type: none"> <li>этапы решения задачи на компьютере на основе парадигмы императивного программирования;</li> <li>понятие алгоритма и его основные свойства;</li> <li>основные понятия алгоритмической системы: представление информации в виде данных, система команд исполнителя, алгоритмический язык исполнителя;</li> <li>основные средства записи и типы алгоритмов;</li> <li>цели и составные части технологии нисходящего структурного проектирования алгоритмов и программ;</li> <li>алгоритмические структуры, их основные свойства и приемы использования;</li> <li>понятие алгоритмического модуля и его свойства;</li> <li>принципы выделения и правила записи алгоритмических модулей;</li> <li>семантику вызова алгоритмического модуля, в том числе, рекурсивного вызова, и механизмы реализации межмодульного интерфейса</li> </ul>	<ul style="list-style-type: none"> <li>при решении стандартных задач профессиональной деятельности на основе парадигмы императивного программирования конкретизировать и описать основные понятия, результаты других научных дисциплин, данные и их типы средствами записи алгоритмов и программ;</li> <li>разрабатывать и записывать алгоритмы и программы на языках Бейсик и Паскаль в соответствии с технологией нисходящего структурного проектирования;</li> <li>записывать алгоритмы на языке схем;</li> <li>анализировать структуру алгоритмов;</li> <li>описывать алфавит и основные синтаксические конструкции языков Бейсик и Паскаль с помощью нормальных форм Бэкуса-Наура</li> </ul>	<ul style="list-style-type: none"> <li>основными методами, способами и средствами переработки информации на основе парадигмы императивного программирования в соответствии с технологией нисходящего структурного проектирования.</li> </ul>
2.	ОПК-3	способность работать с компьютером как средством управления информацией, работать с информацией из различных источников, в том числе в глобальных компьютерных сетях	<ul style="list-style-type: none"> <li>составные части алгоритмического языка программирования (алфавит, синтаксис, семантика) и способы их описания;</li> <li>концепцию типов данных;</li> <li>структуры программ на языках программирования Бейсик и Паскаль;</li> <li>основные понятия алгоритмических языков программирования и их классификацию: иденти-</li> </ul>	<ul style="list-style-type: none"> <li>разрабатывать и записывать алгоритмы и программы на языках Бейсик и Паскаль в соответствии с принципом структурности;</li> <li>при решении учебной задачи конкретизировать и описать основные понятия, данные и их типы, организовать ввод и вывод,</li> </ul>	<ul style="list-style-type: none"> <li>навыками проектирования, ввода, отладки и тестирования программ в средах, по крайней мере, двух императивных систем программирования.</li> </ul>

		<p>фикаторы, данные и их типы, операции, стандартные функции, выражения, операторы;</p> <ul style="list-style-type: none"> <li>• синтаксис и семантику основных операторов языков программирования Бейсик и Паскаль (присваивания, ввода-вывода, ветвления, циклов), способы реализации алгоритмических структур средствами языков программирования;</li> <li>• основные синтаксические конструкции, используемые в языках программирования Бейсик и Паскаль для описания и обработки данных типа массив;</li> <li>• прием флаг;</li> <li>• синтаксис и семантику реализации алгоритмического модуля средствами языков Бейсик и Паскаль;</li> <li>• способы передачи параметров при реализации межмодульного интерфейса;</li> <li>• синтаксис типов запись с фиксированными полями, указатель и типизированный файл в языке Паскаль;</li> <li>• понятия статического и динамического объекта программы на языке Паскаль;</li> <li>• синтаксис и семантику основных средств управления использованием динамической памяти в языке Паскаль;</li> <li>• структуру данных файл и основные действия с файлами последовательного доступа;</li> <li>• синтаксис и семантику основных средств работы с файлами в СП Turbo-Pascal и последовательность их использования для организации ввода и вывода.</li> </ul>	<p>реализовать алгоритмические структуры средствами, по крайней мере, двух императивных языков программирования;</p> <ul style="list-style-type: none"> <li>• разрабатывать алгоритмы и программы обработки массивов;</li> <li>• разрабатывать алгоритмы и программы обработки строк;</li> <li>• решать учебные задачи в соответствии с принципом модульности;</li> <li>• разрабатывать и использовать библиотеку пользователя в СП Turbo-Pascal;</li> <li>• разрабатывать простейшие алгоритмы и программы моделирования движения графического изображения;</li> <li>• использовать тип указатель для организации односвязного списка, очереди и стека;</li> <li>• решать учебные задачи по обработке данных типа запись, файл и линейных динамических структур данных средствами системы программирования (СП) Turbo-Pascal.</li> </ul>	
--	--	---	---	--

## 2.5. Карта компетенций

### КАРТА КОМПЕТЕНЦИЙ ДИСЦИПЛИНЫ

#### НАИМЕНОВАНИЕ ДИСЦИПЛИНЫ: АЛГОРИТМИЗАЦИЯ И ОСНОВЫ ПРОГРАММИРОВАНИЯ

<b>Цель дисциплины</b>	Формирование у обучающихся общепрофессиональных компетенций в процессе изучения алгоритмизации и основ программирования для последующего применения в учебной и практической деятельности
------------------------	---

В процессе освоения данной дисциплины студент формирует и демонстрирует следующие

#### Общепрофессиональные компетенции:

КОМПЕТЕНЦИИ		Перечень компонентов	Технологии формирования	Форма оценочного средства	Уровни освоения компетенций
ИНДЕКС	ФОРМУЛИРОВКА				
ОПК-1	способность решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности	<p><b>Знать:</b></p> <ul style="list-style-type: none"> <li>этапы решения задачи на компьютере на основе парадигмы императивного программирования;</li> <li>понятие алгоритма и его основные свойства;</li> <li>основные понятия алгоритмической системы: представление информации в виде данных, система команд исполнителя, алгоритмический язык исполнителя;</li> <li>основные средства записи и типы алгоритмов;</li> <li>цели и составные части технологии нисходящего структурного проектирования алгоритмов и программ;</li> <li>алгоритмические структуры, их основные свойства и приемы использования;</li> <li>понятие алгоритмического модуля и его свойства;</li> <li>принципы выделения и правила записи алгоритмических модулей;</li> <li>семантику вызова алгоритмического модуля, в том числе, рекурсивного вызова, и механизмы реализации межмодульного интерфейса</li> </ul> <p><b>Уметь:</b></p> <ul style="list-style-type: none"> <li>при решении стандартных задач профессиональной деятельности на основе парадигмы императивного программирования конкретизировать и описать основные понятия, результаты других научных дисциплин, данные и их типы средствами записи алгоритмов и программ;</li> <li>разрабатывать и записывать алгоритмы и программы на языках Бейсик и Паскаль в соответствие с технологией нисходящего структурного проектирования;</li> </ul>	Путем проведения лекционных, лабораторных занятий, применения новых образовательных технологий, организации самостоятельной работы студентов	Защита лабораторных работ, зачет, экзамен	<p><b>Пороговый</b> Способен решать стандартные задачи получения, хранения, переработки информации в соответствии с технологией нисходящего структурного проектирования.</p> <p><b>Повышенный</b> Способен решать задачи получения, хранения, переработки информации повышенной сложности в соответствии с технологией нисходящего структурного проектирования</p>

		<ul style="list-style-type: none"> <li>• записывать алгоритмы на языке схем;</li> <li>• анализировать структуру алгоритмов;</li> <li>• описывать алфавит и основные синтаксические конструкции языков Бейсик и Паскаль с помощью нормальных форм Бэкуса-Наура.</li> </ul> <p><b>Владеть:</b></p> <ul style="list-style-type: none"> <li>• основными методами, способами и средствами переработки информации на основе парадигмы императивного программирования в соответствии с технологией нисходящего структурного проектирования</li> </ul>			
ОПК-3	<p>способность работать с компьютером как средством управления информацией, работать с информацией из различных источников, в том числе в глобальных компьютерных сетях</p>	<p><b>Знать</b></p> <ul style="list-style-type: none"> <li>• составные части алгоритмического языка программирования (алфавит, синтаксис, семантика) и способы их описания;</li> <li>• концепцию типов данных;</li> <li>• структуры программ на языках программирования Бейсик и Паскаль;</li> <li>• основные понятия алгоритмических языков программирования и их классификацию: идентификаторы, данные и их типы, операции, стандартные функции, выражения, операторы;</li> <li>• синтаксис и семантику основных операторов языков программирования Бейсик и Паскаль (присваивания, ввода-вывода, ветвления, циклов), способы реализации алгоритмических структур средствами языков программирования;</li> <li>• основные синтаксические конструкции, используемые в языках программирования Бейсик и Паскаль для описания и обработки данных типа массив;</li> <li>• прием флаг;</li> <li>• синтаксис и семантику реализации алгоритмического модуля средствами языков Бейсик и Паскаль;</li> <li>• способы передачи параметров при реализации межмодульного интерфейса;</li> <li>• синтаксис типов запись с фиксированными полями, указатель и типизированный файл в языке Паскаль;</li> <li>• понятия статического и динамического объекта программы на языке Паскаль;</li> <li>• синтаксис и семантику основных средств управления использованием динамической памяти в языке Паскаль;</li> <li>• структуру данных файл и основные действия с файлами последовательного доступа;</li> <li>• синтаксис и семантику основных средств работы с файлами</li> </ul>	<p>Путем проведения лекционных, лабораторных занятий, применения новых образовательных технологий, организации самостоятельной работы студентов</p>	<p>Защита лабораторных работ, зачет, экзамен</p>	<p><b>Пороговый</b> Способен решать стандартные задачи средствами императивных систем программирования</p> <p><b>Повышенный</b> Способен решать задачи повышенной сложности средствами императивных систем программирования</p>

	<p>в СП Turbo-Pascal и последовательность их использования для организации ввода и вывода.</p> <p><b>Уметь:</b></p> <ul style="list-style-type: none"><li>• разрабатывать и записывать алгоритмы и программы на языках Бейсик и Паскаль в соответствии с принципом структурности;</li><li>• при решении учебной задачи конкретизировать и описать основные понятия, данные и их типы, организовать ввод и вывод, реализовать алгоритмические структуры средствами, по крайней мере, двух императивных языков программирования;</li><li>• разрабатывать алгоритмы и программы обработки массивов;</li><li>• разрабатывать алгоритмы и программы обработки строк;</li><li>• решать учебные задачи в соответствии с принципом модульности;</li><li>• разрабатывать и использовать библиотеку пользователя в СП Turbo-Pascal;</li><li>• разрабатывать простейшие алгоритмы и программы моделирования движения графического изображения;</li><li>• использовать тип указатель для организации односвязного списка, очереди и стека;</li><li>• решать учебные задачи по обработке данных типа запись, файл и линейных динамических структур данных средствами системы программирования (СП) Turbo-Pascal.</li></ul> <p><b>Владеть:</b></p> <ul style="list-style-type: none"><li>• навыками проектирования, ввода, отладки и тестирования программ в средах, по крайней мере, двух императивных систем программирования.</li></ul>			
--	--	--	--	--

## ОСНОВНАЯ ЧАСТЬ

### 1. ОБЪЕМ УЧЕБНОЙ ДИСЦИПЛИНЫ И ВИДЫ УЧЕБНОЙ РАБОТЫ

Вид учебной работы	Всего часов	Семестры	
		№ 1 часов	№ 2 часов
1	2	3	4
<b>Контактная работа обучающихся с преподавателем (по видам учебных занятий) (всего)</b>	<b>192</b>	<b>84</b>	<b>108</b>
В том числе:			
Лекции (Л)	70	34	36
Лабораторные работы (ЛР)	122	50	72
<b>Самостоятельная работа студента (всего)</b>	<b>240</b>	<b>96</b>	<b>144</b>
В том числе:			
Изучение литературы и других источников	62	28	34
Подготовка к выполнению лабораторных работ	102	34	68
Подготовка к защите лабораторных работ	76	34	42
<b>Контроль</b>	<b>36</b>	<b>-</b>	<b>36</b>
<b>Вид промежуточной аттестации</b>	<b>зачет (З)</b>		+
	<b>экзамен (Э)</b>		+
<b>ИТОГО: общая трудоемкость</b>	<b>часов</b>	<b>468</b>	<b>180</b>
	<b>зач. ед.</b>	<b>13</b>	<b>5</b>

### 2. СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

#### 2.1. Содержание разделов учебной дисциплины

№ семестра	№ раздела	Наименование раздела учебной дисциплины	Содержание раздела в дидактических единицах
1	2	3	4
1	1	Алгоритмическая система и ее составные части	Алгоритмическая система как совокупность средств и понятий. Понятие алгоритма и исполнителя алгоритма. Представление информации в виде данных: понятие о вводе/выводе; входные, выходные и промежуточные данные. Система команд исполнителя. Константы, переменные, операции, функции и выражения. Понятие типа данных. Алгоритмический язык исполнителя. Принципы фон Неймана как основа структуры персонального компьютера. Представление в компьютере целых чисел. Представление в компьютере действительных чисел. Средства записи алгоритмов. Схемы алгоритмов
1	2	Основные понятия и простейшие средства алгоритмических языков программирования	Программирование как этап решения задачи на компьютере. Понятие языка программирования высокого уровня. Понятие о исполнителе программы на языке высокого уровня. Составные части языка высокого уровня: алфавит, синтаксис, семантика. Метаязыки описания конструкций языков программирования. Расширенные Бэкуса-Наура формы, примеры метаформул. Структуры программ и концепции типов данных в алгоритмических языках на примере языков Qbasic и Turbo-Pascal. Оператор как логически завершенная конструкция алгоритмического языка программирования. Различные классификации операторов: исполняемые и неисполняемые, простые и составные. Различные классификации типов данных: простые и структурированные, стандартные и определяемые пользователем. Основные простые типы данных в языках Qbasic и Pascal (кроме логического): идентификаторы стандартных типов, диапазоны, операции и функции, синтаксис типа диапазон. Особен-



			ности целочисленной и действительной арифметики в системах программирования (СП) Qbasic и Turbo-Pascal. Синтаксис разделов программы на языке Turbo-Pascal, понятие порядкового типа в языке Pascal. Синтаксис и семантика оператора присваивания, знакомство с операторами ввода/вывода в языках Qbasic и Pascal. Общая характеристика императивных СП Qbasic и Turbo-Pascal: основное меню, окна редактирования и результатов исполнения программы, диалоговые окна, встроенная система помощи. Написание, ввод, отладка и тестирование программ линейной структуры в СП Qbasic и Turbo-Pascal
1	3	Технология нисходящего структурного проектирования. Разработка разветвляющихся алгоритмов и программ в соответствии с принципом структурности	Цели и составные части технологии нисходящего структурного проектирования. Взаимосвязь принципов нисходящего проектирования, модульности и структурности. Алгоритмические структуры как основа структурного кодирования. Базовые алгоритмические структуры: следование и развилка. Вложение (суперпозиция) алгоритмических структур как основной прием принципа структурности. Присоединение алгоритмических структур. Правила пунктуации в программах на алгоритмическом языке. Составной оператор в языке Pascal. Логический тип данных в алгоритмических языках Qbasic и Pascal. Операции отношения и логические операции. Алгоритмические структуры, реализующие ветвления. Организация ветвлений средствами алгоритмических языков: синтаксис и семантика строчного и блочного условных операторов в Qbasic, условного оператора в Pascal, операторов выбора в языках Qbasic и Pascal. Понятие о приеме программирования "флаг". Примеры алгоритмов и программ разветвляющейся структуры. Разработка схем алгоритмов разветвляющейся структуры. Написание, ввод, отладка и тестирование программ разветвляющейся структуры в СП Qbasic и Turbo-Pascal
1	4	Разработка циклических алгоритмов и программ в соответствии с принципом структурности	Базовая алгоритмическая структура цикл с предусловием. Цикл с постусловием. Проблема заикливания. Цикл с параметром. Синтаксис и семантика операторов циклов в языках Qbasic и Pascal. Особенности системной организации циклов с параметром в СП Qbasic и Turbo-Pascal. Рекомендации по программированию циклов. Примеры циклических алгоритмов и программ. Разработка схем алгоритмов циклической структуры. Написание, ввод, отладка и тестирование программ циклической структуры в СП Qbasic и Turbo-Pascal
1	5	Ввод с клавиатуры и вывод на монитор средствами алгоритмических языков	Понятие о текстовом режиме монитора, текущее положение курсора. Синтаксис и семантика операторов ввода с клавиатуры в СП Qbasic и Turbo-Pascal. Особенности ввода с клавиатуры символьной и строковой информации в СП Turbo-Pascal. Общая характеристика операторов вывода на монитор в императивных алгоритмических СП. Синтаксис оператора PRINT в Qbasic. Организация слитного и зонального вывода средствами оператора PRINT, примеры. Особенности вывода значений числового и строкового типа. Организация форматированного вывода средствами оператора PRINT USING в Qbasic: синтаксис, символы форматирования, основные форматы для вывода значений строкового и числового типа, семантика. Синтаксис и семантика оператора вывода на монитор в СП Turbo-Pascal. Форматированный вывод текстовой и числовой информации на монитор в СП Turbo-Pascal. Диалоговые программы. Дружественность интерфейса с пользователем. Примеры организации форматированного вывода на монитор. Разработка схем алгоритмов, написание, ввод, отладка и тестирование программ с форматированным выводом на монитор в СП Qbasic и Turbo-Pascal
1	6	Структурированный тип данных массив	Синтаксис и семантика оператора резервирования памяти в Qbasic, примеры. Особенности распределения памяти в СП Qbasic. Синтаксис определения типа массив в языке Pascal. Рекомендации по описанию переменной типа массив и понятие полной совместимости типов в языке Pascal, примеры. Использование переменной типа массив в операторе присваивания. Элемент массива (переменная с индексами). Основные алгоритмы и программы обработки числовых массивов: поиск максимума (минимума), поиск элементов с заданными свойствами, сортировка методом пузырька, модифицированный алгоритм сортировки методом пузырька. Примеры алгоритмов и программ обработки массивов, в том числе с использованием приема "флаг". Разработка схем алгоритмов обработки массивов. Написание, ввод, отладка и тестирование программ обработки массивов в СП Qbasic и Turbo-Pascal

2	7	Символьный и строковый типы данных	Стандартный идентификатор строкового типа в языках Qbasic и Pascal. Строковый тип в языке Pascal как массив символов. Операция конкатенации в Qbasic и Turbo-Pascal. Синтаксис и семантика функции MID\$ и оператора MID\$. Функции и процедуры для работы со строками в языке Turbo-Pascal. Примеры алгоритмов и программ обработки данных строкового типа. Написание, ввод, отладка и тестирование программ обработки строк в СП Qbasic и Turbo-Pascal
2	8	Разработка алгоритмов в соответствии с принципом модульности	Понятие алгоритмического модуля, его свойства и правила выделения. Концепция "черного ящика". Правила записи алгоритмического модуля: спецификация и тело модуля. Входные и выходные формальные параметры. Блок вызова алгоритмического модуля. Фактические параметры. Семантика вызова алгоритмического модуля и механизм реализации межмодульного интерфейса. Понятие рекурсии. Семантика рекурсивного вызова. Примеры простейших алгоритмов, составленных в соответствии с принципом модульности. Разработка схем алгоритмов в соответствии с принципом модульности
2	9	Реализация принципа модульности средствами алгоритмических языков программирования	Способы записи алгоритмического модуля средствами алгоритмических языков программирования: процедуры и функции. Синтаксис процедуры, синтаксис и семантика оператора процедуры в языке Qbasic. Способы передачи параметров значением и ссылкой. Синтаксис процедуры, синтаксис и семантика оператора процедуры в языке Pascal. Параметры значения и параметры переменные. Рекомендации по выбору способа передачи параметров. Примеры простейших программ, составленных в соответствии с принципом модульности. Синтаксис функции в языках Qbasic и Pascal. Рекомендации по использованию функций в императивных СП Qbasic и Turbo-Pascal. Примеры простейших программ с использованием функций пользователя. Синтаксис процедурного типа в языке Pascal. Рекомендации по использованию процедурного типа для решения задач: параметры процедуры и параметры функции. Пример алгоритма и программы решения задачи с использованием процедурного типа. Примеры алгоритма и программ решения задачи рекурсивной природы. Понятие о библиотеке СП Turbo-Pascal. Обзор стандартных модулей. Синтаксис раздела модулей программы и особенности трансляции процедур и функций модуля СП Turbo-Pascal. Синтаксис модуля пользователя и рекомендации по организации библиотеки пользователя в СП Turbo-Pascal. Длинная целочисленная арифметика. Примеры программ сложения и вычитания натуральных чисел с числом разрядов больше десяти. Написание, ввод, отладка и тестирование программ в соответствии с принципом модульности в СП Qbasic и Turbo-Pascal
2	10	Структурированный тип данных запись	Синтаксис типа запись с фиксированными полями в языке Pascal. Рекомендации по описанию переменной типа запись, примеры. Использование переменной типа запись в операторе присваивания. Селектор записи (переменная с точкой). Синтаксис оператора присоединения. Примеры алгоритмов и программ обработки данных типа запись. Разработка схем алгоритмов обработки данных типа запись. Написание, ввод, отладка и тестирование программ обработки данных типа запись в соответствии с принципом модульности в СП Turbo-Pascal
2	11	Тип данных указатель	Понятия статических и динамических объектов программы на языке Паскаль. Синтаксис типа указатель (ссылочный тип) в языке Pascal. Рекомендации по описанию переменной типа указатель, примеры. Синтаксис и семантика оператора резервирования (захвата) памяти под динамическую переменную в языке Pascal, примеры. Использование переменной типа указатель в операторе присваивания и в выражениях отношения с операциями = и <>. Динамическая переменная (переменная с крышкой). Синтаксис и семантика освобождения области памяти динамической переменной в языке Pascal. Понятие динамических структур данных. Использование типа указатель для организации динамического односвязного списка по принципам ссылки вперед и ссылки назад. Примеры алгоритмов и программ создания динамических односвязных списков. Алгоритмы вставки (добавления), удаления и поиска элемента динамического односвязного списка. Структуры данных на основе односвязных списков: очередь и стек. Рекомендации по созданию и обработке динамических структур данных очередь и стек с примерами фрагментов алгоритмов и программ. Разработка схем алгоритмов создания и обра-

			ботки динамических структур данных. Написание, ввод, отладка и тестирование программ создания и обработки динамических структур данных в соответствии с принципом модульности в СП Turbo-Pascal
2	12	Структурированный тип данных файл	Понятие файла, файлы данных различного доступа. Основные действия с файлами данных последовательного доступа: запись (вывод в файл) и чтение (ввод из файла). Обзор файловых типов данных в языке Pascal, синтаксис типизированного файлового типа. Рекомендации по описанию переменной типа файл, примеры. Синтаксис и семантика основных операторов для работы с типизированными файлами данных в СП Turbo-Pascal: связь файловой переменной с файлом на внешнем устройстве, открытие файла для записи и чтения, вывод в файл и ввод из файла, закрытие файла. Функция конец типизированного файла. Рекомендации по корректной обработке типизированных файлов с примерами фрагментов алгоритмов и программ. Разработка схем алгоритмов создания и обработки типизированных файлов. Написание, ввод, отладка и тестирование программ создания и обработки типизированных файлов в соответствии с принципом модульности в СП Turbo-Pascal
2	13	Обзор графических возможностей систем программирования (СП) Qbasic и Turbo-Pascal	Понятие о графическом режиме монитора, текущее положение графического курсора. Операторы установки параметров окна результатов и задания текущего цвета изображения в языке Qbasic. Обзор стандартных средств языка Qbasic для изображения графических примитивов: точек, отрезков, прямоугольников, окружностей и эллипсов, дуг окружностей и эллипсов. Операторы стандартных процедур модуля Graph библиотеки СП Turbo-Pascal для определения подходящего графического драйвера, инициализации и прекращения графического режима и задания текущего цвета изображения. Обзор стандартных процедур модуля Graph для изображения графических примитивов: точек, отрезков, прямоугольников, окружностей и эллипсов, дуг окружностей и эллипсов. Простейший алгоритм моделирования движения графического изображения (анимация). Написание, ввод, отладка и тестирование программ моделирования движения графического изображения в соответствии с принципом модульности в СП Qbasic и Turbo-Pascal

## 2.2. Разделы учебной дисциплины, виды учебной деятельности и формы контроля

№ семестра	№ раздела	Наименование раздела учебной дисциплины	Виды учебной деятельности, включая самостоятельную работу студентов ( в часах)				Формы текущего контроля успеваемости (по неделям семестра)
			Л	ЛР	СРС	всего	
1	2	3	4	5	6	7	8
1	1	Алгоритмическая система и ее составные части			3	3	2 неделя: лабораторная работа №1
	2	Основные понятия и простейшие средства алгоритмических языков программирования	3	6	12	21	
	3	Технология нисходящего структурного проектирования. Разработка разветвляющихся алгоритмов и программ в соответствии с принципом структурности	8	12	22	42	3-4 неделя: лабораторная работа №2 5-6 неделя: лабораторная работа №3
	4	Разработка циклических алгоритмов и программ в соответствии с принципом структурности	9	15	24	48	7-8 неделя: лабораторная работа №4 9-11 неделя: лабораторная работа №5
	5	Ввод с клавиатуры и вывод на монитор средствами алгоритмических языков	4	8	14	26	12-14 неделя: лабораторная работа №6
	6	Структурированный тип данных массив	10	9	21	40	15-17 неделя: лабораторная работа №7
		<b>ИТОГО 1 семестр</b>	<b>34</b>	<b>50</b>	<b>96</b>	<b>180</b>	<b>Зачет</b>
2	7	Символьный и строковый типы данных	6	20	32	52	1-2 неделя: лабораторная работа №8 3-5 неделя: лабораторная работа №9
	8	Разработка алгоритмов в соответствии с принципом модульности	2	8	14	20	6-7 неделя: лабораторная работа №10
	9	Реализация принципа модульности средствами алгоритмических языков программирования	10	20	36	60	8-12 неделя: лабораторная работа №11
	10	Структурированный тип данных запись	4	4	14	16	13 неделя: лабораторная работа №12
	11	Тип данных указатель	8	8	20	32	15-16 неделя: лабораторная работа №13
	12	Структурированный тип данных файл	4	4	14	16	
	13	Обзор графических возможностей систем программирования (СИ) Qbasic и Turbo-Pascal	2	8	14	20	17-18 неделя: лабораторная работа №14
		<b>Контроль</b>				36	<b>Экзамен</b>
		<b>ИТОГО 2 семестр</b>	<b>36</b>	<b>72</b>	<b>180</b>	<b>288</b>	
		<b>ИТОГО</b>	<b>70</b>	<b>122</b>	<b>276</b>	<b>468</b>	

### 2.3. Лабораторный практикум

№ се-местра	№ раз-дела	Наименование раздела учебной дисциплины	Наименование лабораторных работ	Всего часов
1	2	3	4	5
1	1	Алгоритмическая система и ее составные части	ЛР вводная. Знакомство с императивными системами программирования QBasic и Turbo-Pascal. Ввод, отладка и тестирование простейших программ	6
	2	Основные понятия и простейшие средства алгоритмических языков программирования	ЛР №1. Написание, ввод, отладка и тестирование программ линейной структуры	
	3	Технология нисходящего структурного проектирования. Разработка разветвляющихся алгоритмов и программ в соответствии с принципом структурности	ЛР №2. Написание, ввод, отладка и тестирование алгоритмов и программ разветвляющейся структуры	6
			ЛР №3. Написание, ввод, отладка и тестирование разветвляющихся алгоритмов и программ с использованием структуры выбор	6
	4	Разработка циклических алгоритмов и программ в соответствии с принципом структурности	ЛР №4. Написание, ввод, отладка и тестирование циклических алгоритмов и программ	6
			ЛР №5. Написание, ввод, отладка и тестирование циклических алгоритмов и программ целочисленной арифметики	9
	5	Ввод с клавиатуры и вывод на монитор средствами алгоритмических языков	ЛР №6. Написание, ввод, отладка и тестирование циклических алгоритмов и программ с форматированным выводом на монитор	8
6	Структурированный тип данных массив	ЛР №7. Написание, ввод, отладка и тестирование алгоритмов и программ обработки числовых массивов	9	
		<b>ИТОГО 1 семестр</b>		<b>50</b>
2	7	Символьный и строковый типы данных	ЛР №8. Написание, ввод, отладка и тестирование алгоритмов и программ обработки строк	8
			ЛР №9. Написание, ввод, отладка и тестирование алгоритмов и программ обработки массивов строк	12
	8	Разработка алгоритмов в соответствии с принципом модульности	ЛР №10. Написание, ввод, отладка и тестирование алгоритмов и программ в соответствии с принципом модульности	8
	9	Реализация принципа модульности средствами алгоритмических языков программирования	ЛР №11. Написание, ввод, отладка и тестирование алгоритмов и программ обработки массивов в соответствии с принципом модульности	12
	10	Структурированный тип данных запись	ЛР №12. Написание, ввод, отладка и тестирование алгоритмов и программ создания и обработки односвязных списков, записей и типизированных файлов в СП Turbo-Pascal	16
	11	Тип данных указатель		
	12	Структурированный тип данных файл	ЛР №13. Написание, ввод, отладка и тестирование алгоритмов и программ длинной арифметики в соответствии с принципом модульности	8
13	Обзор графических возможностей систем программирования (СП) Qbasic и Turbo-Pascal	ЛР №14. Написание, ввод, отладка и тестирование алгоритмов и программ моделирования движения графического изображения в соответствии с принципом модульности	8	
		<b>ИТОГО 2 семестр</b>		<b>72</b>

2.4. Курсовые работы по дисциплине не предусмотрены.

### 3. САМОСТОЯТЕЛЬНАЯ РАБОТА СТУДЕНТА

#### 3.1. Виды СРС

№ семестра	№ раздела	Наименование раздела учебной дисциплины	Виды СРС	Всего часов
1	2	3	4	5
1	1	Алгоритмическая система и ее составные части	Изучение литературы и других источников	1
			Подготовка к выполнению лабораторной работы	2
1	2	Основные понятия и простейшие средства алгоритмических языков программирования	Изучение литературы и других источников	4
			Подготовка к выполнению лабораторной работы	4
			Подготовка к защите лабораторной работы	4
	3	Технология нисходящего структурного проектирования. Разработка разветвляющихся алгоритмов и программ в соответствии с принципом структурности	Изучение литературы и других источников (ЛР №2)	4
			Подготовка к выполнению лабораторной работы (ЛР №2)	4
			Подготовка к защите лабораторной работы (ЛР №2)	4
			Изучение литературы и других источников (ЛР №3)	3
			Подготовка к выполнению лабораторной работы (ЛР №3)	3
			Подготовка к защите лабораторной работы (ЛР №3)	4
	4	Разработка циклических алгоритмов и программ в соответствии с принципом структурности	Изучение литературы и других источников (ЛР №4)	4
			Подготовка к выполнению лабораторной работы (ЛР №4)	4
			Подготовка к защите лабораторной работы (ЛР №4)	4
			Изучение литературы и других источников (ЛР №5)	4
			Подготовка к выполнению лабораторной работы (ЛР №5)	4
			Подготовка к защите лабораторной работы (ЛР №5)	4
	5	Ввод с клавиатуры и вывод на монитор средствами алгоритмических языков	Изучение литературы и других источников (основная и дополнительная литература)	2
			Изучение литературы и других источников (конспекты лекций и ресурсы компьютерных сетей)	3
			Подготовка к выполнению лабораторной работы	3
			Подготовка к защите лабораторной работы (ввод и вывод в СП Qbasic)	3
			Подготовка к защите лабораторной работы (ввод и вывод в СП Turbo-Pascal)	3
Подготовка к защите лабораторной работы (массивы в СП Turbo-Pascal)			4	
6	Структурированный тип данных массив	Изучение литературы и других источников	3	
		Подготовка к выполнению лабораторной работы (разработка схемы алгоритма)	4	
		Подготовка к выполнению лабораторной работы (разработка программы в СП Qbasic)	3	
		Подготовка к выполнению лабораторной работы (разработка программы в СП Turbo-Pascal)	3	
		Подготовка к защите лабораторной работы (массивы в СП Qbasic)	4	
		Подготовка к защите лабораторной работы (массивы в СП Turbo-Pascal)	4	
		Подготовка к защите лабораторной работы (массивы в СП Turbo-Pascal)	4	
<b>ИТОГО 1 семестр</b>			<b>96</b>	

1	2	3	4	5
2	7	Символьный и строковый типы данных	Изучение литературы и других источников (ЛР №8)	4
			Подготовка к выполнению лабораторной работы (разработка схемы алгоритма ЛР №8)	4
			Подготовка к выполнению лабораторной работы (разработка программ ЛР №8)	4
			Подготовка к защите лабораторной работы (ЛР №8)	4
			Изучение литературы и других источников (ЛР №9)	4
			Подготовка к выполнению лабораторной работы (разработка схемы алгоритма ЛР №9)	4
			Подготовка к выполнению лабораторной работы (разработка программ ЛР №9)	4
			Подготовка к защите лабораторной работы (ЛР №9)	4
	8	Разработка алгоритмов в соответствии с принципом модульности	Изучение литературы и других источников	2
			Подготовка к выполнению лабораторной работы (разработка схемы алгоритма)	3
			Подготовка к выполнению лабораторной работы (разработка программ)	3
			Подготовка к защите лабораторной работы (свойства алгоритмического модуля)	3
			Подготовка к защите лабораторной работы (оформление алгоритмического модуля)	3
	9	Реализация принципа модульности средствами алгоритмических языков программирования	Изучение литературы и других источников (ЛР №10)	3
			Подготовка к выполнению лабораторной работы (разработка схемы алгоритма ЛР №10)	3
			Подготовка к выполнению лабораторной работы (разработка программ ЛР №10)	4
			Подготовка к защите лабораторной работы (реализация принципа модульности в СП Qbasic ЛР №10)	3
			Подготовка к защите лабораторной работы (реализация принципа модульности в СП Turbo-Pascal ЛР №10)	3
			Изучение литературы и других источников (основная и дополнительная литература ЛР №11)	3
			Изучение литературы и других источников (конспекты лекций и ресурсы компьютерных сетей ЛР №11)	3
			Подготовка к выполнению лабораторной работы (разработка схемы алгоритма ЛР №11)	4
			Подготовка к выполнению лабораторной работы (разработка программ ЛР №11)	4
			Подготовка к защите лабораторной работы (ввод из блока данных в СП Qbasic ЛР №11)	3
			Подготовка к защите лабораторной работы (типизированные константы в СП Turbo-Pascal ЛР №11)	3
			10	Структурированный тип данных запись
	Изучение литературы и других источников (конспекты лекций и ресурсы компьютерных сетей)	3		
	Подготовка к выполнению лабораторной работы (разработка структуры данных)	4		
	Подготовка к выполнению лабораторной работы (разработка описаний)	4		
	11	Тип данных указатель	Подготовка к выполнению лабораторной работы (разработка схем алгоритмов)	3
			Подготовка к выполнению лабораторной работы (разработка описаний)	3
			Подготовка к выполнению лабораторной работы (разработка программного модуля)	3
			Подготовка к защите лабораторной работы (тип указатель)	2
			Подготовка к защите лабораторной работы (структуры данных на основе односвязных списков)	3
			Изучение литературы и других источников (основная и дополнительная литература)	3
			Изучение литературы и других источников (конспекты лекций и ресурсы компьютерных сетей)	3

2	12	Структурированный тип данных файл	Подготовка к выполнению лабораторной работы	3
			Подготовка к защите лабораторной работы (вывод в файл)	4
			Подготовка к защите лабораторной работы (ввод из файла)	4
			Изучение литературы и других источников	3
	13	Обзор графических возможностей систем программирования (СП) Qbasic и Turbo-Pascal	Подготовка к выполнению лабораторной работы (разработка схем алгоритмов)	3
			Подготовка к выполнению лабораторной работы (разработка программы для СП Qbasic)	4
			Подготовка к выполнению лабораторной работы (разработка программы для СП Turbo-Pascal)	4
			Подготовка к защите лабораторной работы	3
<b>ИТОГО 2 семестр</b>			<b>144</b>	
Экзамен			Подготовка к экзамену	<b>36</b>
			<b>ИТОГО</b>	<b>276</b>

### 3.2. График работы студента

#### Семестр № 1

Форма оценочного средства	Усл. обозн.	НЕДЕЛЯ																
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
Защита лабораторных работ	ЗЛР		+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+

#### Семестр № 2

Форма оценочного средства	Усл. обозн.	НЕДЕЛЯ																	
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
Защита лабораторных работ	ЗЛР	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+

### 3.3. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине

Учебники, учебные пособия, ресурсы сети Интернет (см. раздел 5).

#### 3.3.1. Контрольные работы/рефераты не предусмотрены

## 4. ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ КОНТРОЛЯ УСПЕВАЕМОСТИ И РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ

(см. Фонд оценочных средств)

### 4.1. Рейтинговая система оценки знаний обучающихся по учебной дисциплине

Рейтинговая система не используется.



## 5. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

### 5.1. Основная литература

№ п/п	Автор (ы), наименование, место издания и издательство, год	Используется при изучении разделов	Семестр	Количество экземпляров	
				В библиотеке	На кафедре
1	2	3	4	5	6
1.	Информатика и программирование. Алгоритмизация и программирование [Текст]: учебник для студ. учреждений высш. проф. образования/ [Н.И.Парфилова, А.В.Пруцков, А.Н.Пылькин, Б.Г.Трусов]; под ред. Б.Г.Трусова.-М.: Издательский центр «Академия», 2012.-336 с.	1-13	1, 2	10	
2.	Новичков В.С., Пылькин А.Н. Начала программирование на языке QBasic [Текст]: Учебное пособие. М.: Горячая линия-Телеком, 2007	1-13	1, 2	10	

### 5.2. Дополнительная литература

№ п/п	Автор (ы), наименование, место издания и издательство, год	Используется при изучении разделов	Семестр	Количество экземпляров	
				В библиотеке	На кафедре
1	2	3	4	5	6
1.	Новичков, В. С. Алгоритмизация и программирование на Турбо Паскале [Текст]: учебное пособие / В. С. Новичков, Н. И. Парфилова, А. Н. Пылькин. – Москва : Горячая линия-Телеком, 2005. – 438 с.	1-13	1, 2	7	
2.	Культин, Н. Программирование в Turbo Pascal 7.0 и Delphi [Текст] : самоучитель / Н. Культин. - 2-е изд., перераб. и доп. - СПб. : БХВ-Петербург, 2001. - 416 с.	1-12	1, 2	7	

### 5.3. Базы данных, информационно-справочные и поисковые системы

1. BOOR.ru [Электронный ресурс] : электронная библиотека. – Доступ к полным текстам по паролю. – Режим доступа: <http://www.book.ru> (дата обращения: 30.08.2019).

2. East View [Электронный ресурс] : [база данных]. – Доступ к полным текстам статей научных журналов из сети РГУ имени С.А. Есенина. – Режим доступа: <http://dlib.eastview.com> (дата обращения: 30.08.2019).

3. Moodle [Электронный ресурс] : среда дистанционного обучения / Ряз. гос. ун-т. – Рязань, [Б.г.]. – Доступ, после регистрации из сети РГУ имени С.А. Есенина, из любой точки, имеющей доступ к Интернету. – Режим доступа: <http://e-learn2.rsu.edu.ru/moodle2> (дата обращения: 30.08.2019).

4. Znanium.com [Электронный ресурс] : [база данных]. – Доступ к полным текстам по паролю. – Режим доступа: <http://znanium.com> (дата обращения: 30.08.2019).

5. «Издательство «Лань» [Электронный ресурс] : электронная библиотека. – Доступ к полным текстам по паролю. – Режим доступа: <http://e-lanbook.com> (дата обращения: 30.08.2019).

6. Университетская библиотека ONLINE [Электронный ресурс] : электронная библиотека. – Доступ к полным текстам по паролю. – Режим доступа: <http://www.biblioclub.ru> (дата обращения: 30.08.2019).

7. Юрайт [Электронный ресурс] : электронная библиотека. – Доступ к полным текстам по паролю. – Режим доступа: <http://www.biblio-online.ru> (дата обращения: 30.08.2019).

8. Труды преподавателей [Электронный ресурс] : коллекция // Электронная библиотека Научной библиотеки РГУ имени С.А. Есенина. – Доступ к полным текстам по паролю. – Режим

доступа: <http://dspace.rsu.edu.ru/xmlui/handle/123456789/3> (дата обращения: 30.08.2019).

#### **5.4. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины**

1. eLIBRARY.RU [Электронный ресурс] : научная электронная библиотека. – Режим доступа: <http://elibrary.ru/defaultx.asp>, свободный (дата обращения: 15.05.2018).
2. Единая коллекция цифровых образовательных ресурсов [Электронный ресурс] : федеральный портал. – Режим доступа: <http://school-collection.edu.ru/>, свободный (дата обращения: 15.05.2018).
3. Единое окно доступа к образовательным ресурсам [Электронный ресурс] : федеральный портал. – Режим доступа: <http://window.edu.ru/>, свободный (дата обращения: 15.05.2018).
4. Интернет Университет Информационных технологий. [Электронный ресурс] : сайт. – Режим доступа: <http://www.intuit.ru/>, свободный (дата обращения 10.06.2018).
5. Петров Д.Н. Парадигмы программирования. [Электронный ресурс] : сайт. – Режим доступа: <http://dnpetrov.narod.ru/>, свободный (дата обращения 10.06.2018).
6. Портал естественных наук. [Электронный ресурс] : сайт. – Режим доступа: <http://e-science11.ru>, свободный (дата обращения 10.06.2018).
7. Портал для программистов и администраторов информационных систем. [Электронный ресурс] : сайт. – Режим доступа: <http://www.coderpost.net/>, свободный (дата обращения 10.06.2018).
8. Российский общеобразовательный портал [Электронный ресурс] : образовательный портал. – Режим доступа: <http://www.school.edu.ru/>, свободный (дата обращения: 15.05.2018).
9. Сервер Информационных Технологий [Электронный ресурс] : сайт. – Режим доступа: <http://citforum.ru/>, свободный (дата обращения 15.05.2018).
10. Сайт программирования. [Электронный ресурс] : сайт. – Режим доступа: <http://www.cyberguru.ru/>, свободный (дата обращения 10.06.2018).
11. Сайт программирования в среде Delphi. [Электронный ресурс]: сайт. – Режим доступа: <http://www.delphisources.ru/>, свободный (дата обращения 10.06.2018).
12. Федеральный центр информационно-образовательных ресурсов [Электронный ресурс] : Единое окно доступа к образовательным ресурсам. – Режим доступа: <http://fcior.edu.ru>, свободный (дата обращения: 15.05.2018).

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

**6.1. Требования к аудиториям для проведения занятий:** Специализированные лекционные аудитории, оснащенные видеопроекционным оборудованием, подключенным к компьютеру. Компьютерные классы для проведения лабораторных занятий и организации самостоятельной работы студентов, имеющие рабочие места, оснащенные компьютером с доступом к серверам кафедры ИВТ и МПИ, сети Интернет и видеопроекционному оборудованию.

**6.2. Требования к оборудованию рабочих мест преподавателя и обучающихся:** Персональный компьютер под управлением MS Windows, LibreOffice, системы программирования (СП) Qbasic и Turbo-Pascal.

**6.3. Требования к специализированному оборудованию:** отсутствует

### **7. ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ**

*(Заполняется только для стандарта ФГОС ВПО)*

## 8. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

Вид учебных занятий	Организация деятельности студента
Лекция	<p>Написание конспекта лекций: кратко, схематично, последовательно фиксировать основные положения, выводы, формулировки, обобщения; пометить важные мысли, выделять ключевые слова, термины. Проверка терминов, понятий с помощью энциклопедий, словарей, справочников с выписыванием толкований в тетрадь. Обозначить вопросы, термины, материал, который вызывает трудности, пометить и попытаться найти ответ в рекомендуемой литературе. Если самостоятельно не удастся разобраться в материале, необходимо сформулировать вопрос и задать преподавателю на консультации, на практическом занятии. Уделить внимание следующим понятиям: <i>этапы решения задачи на компьютере; основные понятия алгоритмической системы; принципы пошаговой детализации структурности и модульности; алгоритмические структуры; вложение; структурный анализ; алфавит, синтаксис и семантика языка программирования; программа для компьютера; идентификатор; переменная; данные и их типы, операция, стандартная функция, выражение; простой и структурированный оператор.</i></p>
Лабораторная работа	<p>В соответствии с запланированным на самостоятельную работу временем (раздел 3.1) изучить соответствующий теоретический материал и практические рекомендации.</p> <p>В соответствии с запланированным на самостоятельную работу временем составить схемы алгоритмов и программы решения соответствующего варианта учебной задачи.</p> <p>Согласовать заранее составленные схемы и программы с преподавателем, ведущим занятие. Тексты программ должны содержать короткие комментарии, отражающие тему и номер лабораторной работы, номер варианта, фамилию студента, связь тех или иных переменных с условием задачи, а также комментарии, отражающие этапы решения задачи. Ввод с клавиатуры и вывод на монитор числовых данных должны сопровождаться краткими текстовыми сообщениями.</p> <p>Ввести тексты программ в компьютер, осуществить отладку и тестирование программ, при наличии нескольких программ решения одной и той же задачи добиться с учетом возможных погрешностей одинаковых результатов и продемонстрировать работоспособность программ преподавателю.</p> <p>Оформить отчет о лабораторной работе с указанием фамилии студента, номера лабораторной работы и номера варианта. Оформленная работа также должна содержать полный текст задания, схему алгоритма (кроме л/р № 1), тексты отлаженных на компьютере программ с комментариями по всем переменным. После текста программы на конкретном языке программирования должны быть отражены результаты ее выполнения из окна результатов в том виде, в котором они были выведены системой программирования. В случае ветвящихся алгоритмов и программ необходимо отразить несколько вариантов выполнения программы с различными наборами исходных данных, соответствующих различным ветвям исполнения алгоритма.</p> <p>Защитить оформленную лабораторную работу, продемонстрировав теоретические и практические знания, умения и навыки по соответствующей теме, возможные варианты схем решения задачи, структуры</p>

	алгоритмов которых отличны от структур оформленных схем.
Подготовка к экзамену (зачету)	При подготовке к экзамену (зачету) необходимо ориентироваться на конспекты лекций, рекомендуемую литературу, типовые практические задания и др.

## **9. ПЕРЕЧЕНЬ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, ИСПОЛЬЗУЕМЫХ ПРИ ОСУЩЕСТВЛЕНИИ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА**

Для организации учебной и самостоятельной работы обучаемых используется технология удаленного доступа. Для каждой из учебных групп на сервере кафедры ИВТ и МПИ созданы каталоги с соответствующими правами доступа. В каталоге группы создан подкаталог для данной учебной дисциплины, в котором по мере необходимости преподавателем размещаются рабочая программа дисциплины, электронные варианты лекций, электронные обучающие ресурсы, задания к лабораторным работам, графики выполнения лабораторных работ, материалы для самостоятельной работы, контрольные материалы, оценки текущих результатов учебной деятельности обучающихся и др. материалы для организации учебного процесса по данной дисциплине. Материалы, размещенные в каталоге группы доступны любому обучающемуся соответствующей группы посредством локальной компьютерной сети университета с любого рабочего места компьютерных классов кафедры ИВТ и МПИ.

В каталоге группы также для каждого обучающегося создан личный подкаталог, к которому разрешен доступ только обучающемуся и преподавателям кафедры. В личном подкаталоге обучающийся размещает результаты своей учебной деятельности: выполненные лабораторные работы, отчеты и другие результаты.

## **10. ТРЕБОВАНИЯ К ПРОГРАММНОМУ ОБЕСПЕЧЕНИЮ УЧЕБНОГО ПРОЦЕССА**

1. Операционная система Windows Pro (договор №Тг000043844 от 22.09.15г)
2. Антивирус Kaspersky Endpoint Security (договор №02-ЗК-2019 от 15.04.2019г.)
3. Офисное приложение LibreOffice (свободно распространяемое ПО)
4. Система программирования Qbasic (свободно распространяемое ПО)
5. Система программирования Turbo-Pascal (свободно распространяемое ПО)
6. Архиватор 7-zip (свободно распространяемое ПО)
7. Браузер изображений FastStoneImageViewer (свободно распространяемое ПО)
8. PDF ридер FoxitReader (свободно распространяемое ПО)
9. Медиа проигрыватель VLC media player (свободно распространяемое ПО)
10. Запись дисков ImageBurn (свободно распространяемое ПО)
11. DJVU браузер DjVu Browser Plug-in (свободно распространяемое ПО)

**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ  
ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ  
ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ**

***Паспорт фонда оценочных средств по дисциплине  
для промежуточного контроля успеваемости***

№ п/п	Контролируемые разделы (темы) дисциплины (результаты по разделам)	Код контролируемой компетенции) или её части)	Наименование оценочного средства
1	Алгоритмическая система и ее составные части	ОПК-1, ОПК-3	Зачет, зкзамен
2	Основные понятия и простейшие средства алгоритмических языков программирования		
3	Технология нисходящего структурного проектирования. Разработка разветвляющихся алгоритмов и программ в соответствии с принципом структурности		
4	Разработка циклических алгоритмов и программ в соответствии с принципом структурности		
5	Ввод с клавиатуры и вывод на монитор средствами алгоритмических языков		
6	Структурированный тип данных массив		
7	Символьный и строковый типы данных		
8	Разработка алгоритмов в соответствии с принципом модульности		
9	Реализация принципа модульности средствами алгоритмических языков программирования		
10	Структурированный тип данных запись		
11	Тип данных указатель		
12	Структурированный тип данных файл		
13	Обзор графических возможностей систем программирования (СИ) Qbasic и Turbo-Pascal		

**ТРЕБОВАНИЯ К РЕЗУЛЬТАТАМ ОБУЧЕНИЯ ПО УЧЕБНОЙ ДИСЦИПЛИНЕ**

Индекс компетенции	Содержание компетенции	Элементы компетенции	Индекс элемента
1	2	3	4
ОПК-1	способность решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности	<b>знать</b>	
		31 этапы решения задачи на компьютере на основе парадигмы императивного программирования;	ОПК-1 31
		32 понятие алгоритма и его основные свойства;	ОПК-1 32
		33 основные понятия алгоритмической системы: представление информации в виде данных, система команд исполнителя, алгоритмический язык исполнителя;	ОПК-1 33
		34 основные средства записи и типы алгоритмов;	ОПК-1 34
		35 цели и составные части технологии нисходящего структурного проектирования алгоритмов и программ;	ОПК-1 35
		36 алгоритмические структуры, их основные свойства и приемы использования;	ОПК-1 36
		37 понятие алгоритмического модуля и его свойства;	ОПК-1 37
		38 принципы выделения и правила записи алгоритмических модулей;	ОПК-1 38
		39 семантику вызова алгоритмического модуля, в том числе, рекурсивного вызова, и механизмы реализации межмодульного интерфейса;	ОПК-1 39
		<b>уметь</b>	

		У1 при решении стандартных задач профессиональной деятельности на основе парадигмы императивного программирования конкретизировать и описать основные понятия, результаты других научных дисциплин, данные и их типы средствами записи алгоритмов и программ;	ОПК-1 У1	
		У2 разрабатывать и записывать алгоритмы и программы на языках Бейсик и Паскаль в соответствии с технологией нисходящего структурного проектирования;	ОПК-1 У2	
		У3 записывать алгоритмы с помощью схем;	ОПК-1 У3	
		У4 анализировать структуру алгоритмов;	ОПК-1 У4	
		У5 описывать алфавит и основные синтаксические конструкции языков с помощью нормальных форм Бэкуса-Наура.	ОПК-1 У5	
		<b>владеть</b>		
		В1 основными методами, способами и средствами переработки информации на основе парадигмы императивного программирования в соответствии с технологией нисходящего структурного проектирования.	ОПК-1 В1	
ОПК-3	способность работать с компьютером как средством управления информацией, работать с информацией из различных источников, в том числе в глобальных компьютерных сетях	<b>знать</b>		
		31 составные части алгоритмического языка программирования (алфавит, синтаксис, семантика) и способы их описания;	ОПК-3 31	
		32 концепцию типов данных;	ОПК-3 32	
		33 структуры программ на языках программирования Бейсик и Паскаль;	ОПК-3 33	
		34 основные понятия алгоритмических языков программирования и их классификацию: идентификаторы, данные и их типы, операции, стандартные функции, выражения, операторы;	ОПК-3 34	
		35 синтаксис и семантику основных операторов языков программирования Бейсик и Паскаль (присваивания, ввода-вывода, ветвления, циклов), способы реализации алгоритмических структур средствами языков программирования;	ОПК-3 35	
		36 основные синтаксические конструкции, используемые в языках программирования Бейсик и Паскаль для описания и обработки данных типа массив;	ОПК-3 65	
		37 прием флаг;	ОПК-3 37	
		38 синтаксис и семантику реализации алгоритмического модуля средствами языков Бейсик и Паскаль;	ОПК-3 38	
		39 способы передачи параметров при реализации межмодульного интерфейса;	ОПК-3 39	
		310 синтаксис типов запись с фиксированными полями, указатель и типизированный файл в языке Паскаль;	ОПК-3 310	
		311 понятия статического и динамического объекта программы на языках Паскаль;	ОПК-3 311	
		312 синтаксис и семантику основных средств управления использованием динамической памяти в языках Паскаль и Си;	ОПК-3 312	
		313 структуру данных файл и основные действия с файлами последовательного доступа;	ОПК-3 313	
		314 синтаксис и семантику основных средств работы с файлами в СП Turbo-Pascal и последовательность их использования для организации ввода и вывода.	ОПК-3 314	
		<b>уметь</b>		
		У1 разрабатывать и записывать алгоритмы и программы в соответствии с принципом структурности;	ОПК-3 У1	

	У2 при решении учебной задачи конкретизировать и описать основные понятия, данные и их типы, организовать ввод и вывод, реализовать алгоритмические структуры средствами, по крайней мере, двух императивных языков программирования;	ОПК-3 У2
	У3 разрабатывать алгоритмы и программы обработки массивов;	ОПК-3 У3
	У4 разрабатывать алгоритмы и программы обработки строк;	ОПК-3 У4
	У5 решать учебные задачи в соответствии с принципом модульности;	ОПК-3 У5
	У6 разрабатывать и использовать библиотеку пользователя в СП Turbo-Pascal;	ОПК-3 У6
	У7 разрабатывать простейшие алгоритмы и программы моделирования движения графического изображения;	ОПК-3 У7
	У8 использовать тип указатель для организации односвязного списка, очереди и стека;	ОПК-3 У8
	У9 решать учебные задачи по обработке данных типа запись, файл и линейных динамических структур данных средствами систем программирования (СП) Turbo-Pascal.	ОПК-3 У9
	<b>владеть</b>	
	В1 навыками проектирования, ввода, отладки и тестирования программ в средах, по крайней мере, двух императивных систем программирования.	ОПК-3 В1

### КОМПЛЕКТ ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ (ЗАЧЕТ 1 семестр)

№	Содержание оценочного средства	Индекс оцениваемой компетенции и ее элементов
1	Кратко охарактеризуйте основные этапы компьютерного решения задач на основе парадигмы императивного программирования. Раскройте значение этапов алгоритмизации и программирования.	ОПК-1 31, 32, У2
2	Охарактеризуйте основные элементы системы, являющейся исполнителем программы на алгоритмическом языке программирования. Приведите примеры систем программирования с различными типами трансляторов.	ОПК-1 33
3	Дайте определения понятиям константы, переменные, операции, функции и выражения как способам представления данных в записи алгоритма или программы. Приведите примеры представления данных в записи алгоритма или программы.	ОПК-1 33, У1 ОПК-3 34
4	Дайте определение понятию идентификатор, приведите соответствующую метаформулу. Расскажите, как используются идентификаторы в тексте программы на алгоритмическом языке, приведите примеры.	ОПК-1 У5 ОПК-3 34
5	Охарактеризуйте понятие переменная. С какими другими основными понятиями алгоритмизации и программирования связано это понятие?	ОПК-3 32, 34
6	Охарактеризуйте концепцию типов данных в языке Бейсик. Какие характеристики целых типов данных языка Бейсик Вы знаете? Проанализируйте особенности целочисленной компьютерной арифметики.	ОПК-3 32, 34

7	Охарактеризуйте концепцию типов данных в языке Бейсик. Какие характеристики действительных типов данных языка Бейсик Вы знаете? Проанализируйте особенности действительной компьютерной арифметики.	ОПК-3 32, 34
8	Опишите алгоритм определения типа переменной при исполнении программы на языке Бейсик. Приведите метаформулу, определяющую синтаксис оператора определения типа в языке Бейсик.	ОПК-1 У1, У5 ОПК-3 32, 34
9	Охарактеризуйте концепцию типов данных в языке Паскаль. Какие характеристики числовых типов данных языка Паскаль Вы знаете?	ОПК-3 32, 34
10	Проведите сравнительный анализ структур программ на языках Бейсик и Паскаль. Приведите примеры простейших программ на этих языках.	ОПК-1 У1, У5 ОПК-3 33, 34
11	Охарактеризуйте понятие идентификатор. Какие классификации операторов алгоритмических языков программирования вы знаете? Классифицируйте оператор присваивания. Опишите синтаксис и семантику этого оператора.	ОПК-1 У5 ОПК-3 34, 35
12	Охарактеризуйте цели и составные части технологии нисходящего структурного проектирования. Приведите простейший пример, иллюстрирующий эту технологию разработки алгоритмов и программ.	ОПК-1 35, У2, В1 ОПК-3 У1
13	Опишите синтаксис и семантику оператора ввода с клавиатуры в СП Qbasic. Приведите примеры организации ввода с клавиатуры данных различных типов.	ОПК-1 У5 ОПК-3 34, 35
14	Опишите синтаксис и семантику оператора ввода с клавиатуры в СП Turbo-Pascal. Приведите примеры организации ввода с клавиатуры данных различных типов.	ОПК-1 У5 ОПК-3 34, 35
15	Опишите синтаксис и семантику оператора вывода на монитор в СП Qbasic. Приведите примеры организации вывода на монитор данных различных типов.	ОПК-1 У5 ОПК-3 34, 35
16	Опишите синтаксис и семантику оператора форматированного вывода на монитор в Turbo-Pascal. Приведите примеры организации форматированного вывода на монитор данных различных типов.	ОПК-1 У5 ОПК-3 34, 35
17	Опишите синтаксис и семантику оператора вывода на монитор в СП Qbasic. Приведите примеры организации вывода на монитор данных различных типов.	ОПК-1 У5 ОПК-3 34, 35
18	Составьте схему алгоритма и программу на алгоритмическом языке ввода трех чисел и их вывода в порядке возрастания абсолютных величин, если среди них есть хотя бы одно отрицательное, и вывода в том порядке, в каком они были введены в противном случае	ОПК-1 У1, У2, У3, У4, В1 ОПК-3 34, 35, У1, У2, В1
19	Составьте схему алгоритма и программу на алгоритмическом языке определения по заданным значениям коэффициентов $a \neq 0$ и $b$ характера изменения функции $y = ax^2 + bx + c$ (возрастает, убывает, минимум, максимум) для заданного значения аргумента.	ОПК-1 У1, У2, У3, У4, В1 ОПК-3 34, 35, У1, У2, В1
20	Составьте схему алгоритма и программу на алгоритмическом языке: для трех заданных чисел $a$ , $b$ и $c$ , среди которых не более одного отрицательного, найти среднее геометрического $\sqrt[3]{abc}$ , если все они положительные, и разность между произведением двух положительных и третьим числом в	ОПК-1 У1, У2, У3, У4, В1 ОПК-3 34, 35, У1, У2, В1



	противном случае.	
21	Составьте схему алгоритма и программу на алгоритмическом языке замены для трех заданных попарно различных чисел меньшего числа на среднее арифметическое двух других, если сумма заданных чисел меньше 1.	ОПК-1 У1, У2, У3, У4, В1 ОПК-3 34, 35, У1, У2, В1
22	Составьте схему алгоритма и программу на алгоритмическом языке для выяснения возможности построения треугольника из отрезков заданной длины. В случае возможности построения определить его вид: равносторонний, равнобедренный, разносторонний, прямоугольный, непрямоугольный.	ОПК-1 У1, У2, У3, У4, В1 ОПК-3 34, 35, У1, У2, В1
23	Составьте схему алгоритма и программу на алгоритмическом языке вывода цифр заданного натурального числа в порядке слева направо (от старших разрядов к младшим).	ОПК-1 У1, У2, У3, У4, В1 ОПК-3 34, 35, У1, У2, В1
24	Составьте схему алгоритма и программу на алгоритмическом языке нахождения всех простых чисел, не превышающих заданное натуральное число.	ОПК-1 У1, У2, У3, У4, В1 ОПК-3 34, 35, 37, У1, У2, В1
25	Составьте схему алгоритма и программу на алгоритмическом языке нахождения всех чисел, не превышающих заданное натуральное число и обладающих следующим свойством: произведение цифр искомого числа равно сумме цифр заданного.	ОПК-1 У1, У2, У3, У4, В1 ОПК-3 34, 35, 37, У1, У2, В1
26	Составьте схему алгоритма и программу на алгоритмическом языке нахождения всех совершенных чисел (чисел, равных сумме всех своих делителей, включая единицу и исключая само число), не превышающих заданное натуральное число.	ОПК-1 У1, У2, У3, У4, В1 ОПК-3 34, 35, 37, У1, У2, В1
27	Составьте схему алгоритма и программу на алгоритмическом языке нахождения для двух заданных натуральных чисел наибольшего общего делителя как произведения общих простых множителей.	ОПК-1 У1, У2, У3, У4, В1 ОПК-3 34, 35, У1, У2, В1

### КОМПЛЕКТ ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ (ЭКЗАМЕН 2 СЕМЕСТР)

№	Содержание оценочного средства	Индекс оцениваемой компетенции и ее элементов
1	Охарактеризуйте принципы фон Неймана как основу структуры персонального компьютера.	ОПК-1 33
2	Охарактеризуйте способы представления в компьютере целых чисел. Как способы представления отражаются на особенностях целого типа в языках программирования высокого уровня?	ОПК-1 33 ОПК-3 34
3	Охарактеризуйте способы представления в компьютере действительных чисел. Как способы представления отражаются на особенностях действительного типа в языках программирования высокого уровня?	ОПК-1 33 ОПК-3 34
4	Дайте определения понятию алгоритм. Охарактеризуйте свойства алгоритма. Приведите примеры, иллюстрирующие свойства алгоритма.	ОПК-1 32
5	Дайте определение понятию алгоритмическая система. Охарактеризуйте составные части алгоритмической системы. Проанализируйте алгоритмическую систему на приме-	ОПК-1 33

	ре императивной системы программирования.	
6	Охарактеризуйте концепцию типов данных в алгоритмических языках программирования на примере языков Бейсик и Паскаль. Проанализируйте особенности целочисленной и действительной компьютерной арифметики. Приведите примеры, иллюстрирующие эти особенности.	ОПК-1 У1 ОПК-3 32, 34
7	Охарактеризуйте составные части алгоритмического языка программирования. Приведите метаформулы, определяющие понятие литеры алфавита языка программирования.	ОПК-1 У5 ОПК-3 31
8	Опишите метаязыки описания языков программирования. Приведите примеры метаформул.	ОПК-1 У5
9	Охарактеризуйте принцип структурности, алгоритмические структуры и приемы их использования. Приведите примеры вложения и присоединения алгоритмических структур. Какие конструкции соответствуют структуре следования в языках Бейсик и Паскаль	ОПК-1 36, У3, У4 ОПК-3 35, У1
10	Опишите синтаксис и семантику условных операторов в языках Бейсик и Паскаль. Приведите примеры организации ветвлений посредством условных операторов.	ОПК-1 У2 ОПК-3 35, У1
11	Опишите синтаксис и семантику операторов выбора в языках Бейсик и Паскаль. Приведите примеры организации ветвлений посредством операторов выбора.	ОПК-1 У2 ОПК-3 35, У1
12	Опишите синтаксис и семантику циклов с постусловием в языках Бейсик и Паскаль. Приведите соответствующие примеры организации циклов.	ОПК-1 У2 ОПК-3 35, У1
13	Опишите синтаксис и семантику циклов с предусловием в языках Бейсик и Паскаль. Приведите соответствующие примеры организации циклов.	ОПК-1 У2 ОПК-3 35, У1
14	Опишите синтаксис и семантику цикла с параметром в языке Бейсик. Приведите соответствующие пример организации цикла.	ОПК-1 У2 ОПК-3 35, У1
15	Опишите синтаксис и семантику цикла с параметром в языке Паскаль. Приведите соответствующие пример организации цикла.	ОПК-1 У2 ОПК-3 35, У1
16	Опишите синтаксис и семантику оператора резервирования памяти в СП Qbasic. Приведите пример схемы алгоритма и программы на языке Бейсик поиска элемента массива.	ОПК-1 У1, У5, В1 ОПК-3 36, У1, У2, У3, В1
17	Опишите синтаксис типа массив в языке Паскаль. Приведите пример схемы алгоритма и программы на языке Паскаль сортировки массива.	ОПК-1 У1, У5, В1 ОПК-3 36, У1, У2, У3, В1
18	Опишите синтаксис типизированной константы в языке Паскаль. Приведите примеры определения типизированных констант. Осуществите сравнение типизированной константы и переменной.	ОПК-3 У1, 34, У2
19	Охарактеризуйте строковый тип данных в языке Бейсик. Какие стандартные средства обработки строк языка Бейсик Вы знаете? Приведите примеры обработки строк средствами языка Бейсик.	ОПК-1 У1, У5, В1 ОПК-3 У2, У4
20	Охарактеризуйте строковый тип данных в языке Паскаль. Какие стандартные средства обработки строк языка Паскаль Вы знаете? Приведите примеры обработки строк средствами языка Паскаль.	ОПК-1 У1, У5, В1 ОПК-3 У2, У4
21	Охарактеризуйте принцип модульности и свойства алго-	ОПК-1 37, 38

	ритмического модуля. Осуществите сравнение принцип модульности с принципом «черного ящика».	
22	Опишите правила оформления подалгоритма, приведите пример. Охарактеризуйте вызов алгоритмического модуля.	ОПК-1 38, 39, У2, У5
23	Опишите синтаксис функции пользователя в языке Бейсик. Охарактеризуйте вызов функции пользователя в языке Бейсик и способы передачи параметров. Приведите пример описания и вызова функции пользователя в языке Бейсик.	ОПК-1 У5, В1 ОПК-3 38, 39, У5
24	Опишите синтаксис функции пользователя в языке Паскаль. Охарактеризуйте вызов функции пользователя в языке Паскаль и способы передачи параметров. Приведите пример описания и вызова функции пользователя на языке Паскаль.	ОПК-1 У5, В1 ОПК-3 38, 39, У5
25	Опишите синтаксис процедуры в языке Бейсик. Опишите синтаксис и семантику оператора процедуры в языке Бейсик и способы передачи параметров. Приведите пример описания и вызова процедуры на языке Бейсик.	ОПК-1 У5, В1 ОПК-3 38, 39, У5
26	Опишите синтаксис процедуры в языке Паскаль. Опишите синтаксис и семантику оператора процедуры в языке Паскаль и способы передачи параметров. Приведите пример описания и вызова процедуры на языке Паскаль.	ОПК-1 У5, В1 ОПК-3 38, 39, У5
27	Охарактеризуйте процедурный тип в языке Паскаль. Приведите пример использования процедурного типа для передачи параметра процедуры или параметра функции.	ОПК-1 У5, В1 ОПК-3 38, 39, У5
28	Дайте определение понятия рекурсия. Сформулируйте условия применимости приема рекурсия в алгоритмизации и программировании. Опишите на конкретном примере семантику рекурсивного вызова.	ОПК-1 39, В1 ОПК-3 38, 39, У5
29	Охарактеризуйте стандартную библиотеку СП Turbo-Pascal. Опишите синтаксис модуля пользователя в языке Паскаль, приведите пример описания. Охарактеризуйте модуль пользователя в языке Паскаль как основу библиотеки пользователя.	ОПК-1 У5, В1 ОПК-3 У6, В1
30	Опишите основы построения графических изображений средствами алгоритмических языков программирования на примере СП Qbasic и Turbo-Pascal. Опишите простейший алгоритм моделирования движения графического изображения.	ОПК-1 В1 ОПК-3 У7, В1
31	Опишите синтаксис типа запись с фиксированными полями в языке Паскаль. Приведите описания данных, имеющих тип запись.	ОПК-1 У1, У5 ОПК-3 310, У9
32	Дайте определение понятия файл. Какие классификации файлов данных Вы знаете? Опишите синтаксис типизированного файла в языке Паскаль. Приведите пример описания типизированного файла.	ОПК-1 У1, У5 ОПК-3 310, 313
33	Опишите синтаксис и семантику операторов, последовательность их исполнения для организации вывода в типизированный файл и ввода из типизированного файла в СП Turbo-Pascal. Приведите примеры организации вывода в типизированный файл и ввода из типизированного файла средствами СП Turbo-Pascal.	ОПК-1 У5 ОПК-3 310, 313, 314, У9
34	Охарактеризуйте понятия статического и динамического объекта программы на языке Паскаль. Опишите синтаксис типа указатель в языке Паскаль, синтаксис и семантику ос-	ОПК-1 У5 ОПК-3 311, 312

	новных средств управления использованием динамической памятью в языке Паскаль. Приведите примеры захвата и освобождения памяти под динамические переменные.	
35	Охарактеризуйте линейную структуру данных список. Какие способы организации списков Вы знаете? Опишите использование типа указатель для организации структуры список. Приведите пример схемы и программы алгоритмического модуля на языке Паскаль для организации односвязного списка.	ОПК-1 У1, В1 ОПК-3 У1, У2, У5, У8, У9, В1
36	Приведите пример схемы и программы алгоритмического модуля на языке Паскаль для вставки элемента в односвязный список.	ОПК-1 У1, В1 ОПК-3 У1, У2, У5, У8, У9, В1
37	Приведите пример схемы и программы алгоритмического модуля на языке Паскаль для удаления элемента односвязного списка.	ОПК-1 У1, В1 ОПК-3 У1, У2, У5, У8, У9, В1
38	Охарактеризуйте линейные структуры данных на основе односвязного списка: очередь, стек, закольцованный список. Приведите примеры фрагментов схем алгоритмов и программ на языке Паскаль для удаления и вставки элемента очереди или стека.	ОПК-1 У1, В1 ОПК-3 У1, У2, У8, У9
39	Составьте схему алгоритма и программу формирования для заданной квадратной матрицы вектора, каждая координата которого является суммой элементов, расположенных на диагонали, параллельной вспомогательной. Использовать подалгоритм вычисления суммы элементов одномерного массива.	ОПК-1 У1, У2, У3, У4, В1 ОПК-3 36, 38, У1, У2, У3, У5, В1
40	Составьте схему алгоритма и программу нахождения всех значений третьего индекса заданного трехмерного массива, каждый из которых определяет матрицу, содержащую, по крайней мере, один нулевой элемент. Использовать подалгоритм, проверяющий гипотезу о том, что матрица содержит, по крайней мере, один нулевой элемент. Если таких значений нет, то вывести соответствующее сообщение.	ОПК-1 У1, У2, У3, У4, В1 ОПК-3 36, 37, 38, У1, У2, У3, У5, В1
41	Составьте схему алгоритма и программу нахождения максимума матрицы с использованием подалгоритма поиска минимального элемента одномерного массива (подсказка: $\max\{f_k\} = -\min\{-f_k\}$ ).	ОПК-1 У1, У2, У3, У4, В1 ОПК-3 36, 38, У1, У2, У3, У5, В1
42	Составьте схему алгоритма и программу нахождения для заданной целочисленной матрицы суммы всех элементов столбцов, в которых четность элементов чередуется (для любых двух рядом стоящих элементов один – четный, другой – нечетный). Использовать подалгоритм, проверяющий гипотезу о том, что в целочисленном одномерном массиве четность элементов чередуется. Если таких столбцов нет, то вывести соответствующее сообщение.	ОПК-1 У1, У2, У3, У4, В1 ОПК-3 36, 37, 38, У1, У2, У3, У5, В1
43	Составьте схему алгоритма и программу нахождения для заданной матрицы максимального элемента из элементов строк, которые упорядочены по возрастанию. Использовать подалгоритм, проверяющий гипотезу о том, что элементы одномерного массива упорядочены по возрастанию. Если таких строк нет, то вывести соответствующее сообщение.	ОПК-1 У1, У2, У3, У4, В1 ОПК-3 36, 37, 38, У1, У2, У3, У5, В1
44	Составьте схему алгоритма и программу формирования для заданной матрицы вектора, каждая координата которого	ОПК-1 У1, У2, У3, У4, В1 ОПК-3 36, 38, У1, У2, У3,

	является средним арифметическим элементов, расположенных на диагонали, параллельной главной. Использовать подалгоритм вычисления среднего арифметического элементов одномерного массива.	У5, В1
45	Составьте схему алгоритма и программу нахождения для заданной матрицы произведения всех элементов столбцов, которые упорядочены по убыванию. Использовать подалгоритм, проверяющий гипотезу о том, что элементы одномерного массива упорядочены по убыванию. Если таких столбцов нет, то вывести соответствующее сообщение.	ОПК-1 У1, У2, У3, У4, В1 ОПК-3 36, 37, 38, У1, У2, У3, У5, В1
46	Составьте схему алгоритма и программу приближенного нахождения минимумов функций и соответствующих значений аргументов: $z=3x^2-2y^2+4xy-8x$ , $x \in [1; 3]$ , $y \in [2; 4]$ ; $z=6x^2+y^2-4x-6y+1$ , $x \in [1; 3]$ , $y \in [2; 4]$ . Использовать подалгоритм поиска максимального элемента двумерного массива (подсказка: $\min\{f_{k,L}\} = -\max\{-f_{k,L}\}$ ).	ОПК-1 У1, У2, У3, У4, В1 ОПК-3 36, 38, У1, У2, У3, У5, В1
47	Составьте схему алгоритма и программу нахождения всех значений первого индекса заданного целочисленного трехмерного массива, каждый из которых определяет матрицу, не содержащую простых чисел. Использовать подалгоритм, проверяющий гипотезу о том, что целочисленная матрица не содержит простых чисел. Если таких значений нет, то вывести соответствующее сообщение.	ОПК-1 У1, У2, У3, У4, В1 ОПК-3 36, 37, 38, У1, У2, У3, У5, В1
48	Составьте схему алгоритма и программу нахождения для заданной матрицы элемента, который является наименьшим из элементов, обладающих следующим свойством: хотя бы в одной строке матрицы все элементы не превосходят <b>М</b> . Использовать подалгоритм поиска максимального элемента одномерного массива.	ОПК-1 У1, У2, У3, У4, В1 ОПК-3 36, 38, У1, У2, У3, У5, В1
49	Составьте схему алгоритма и программу нахождения самой длинной из дуг, образованных графиками функций $f(x)=x^2+2\ln(1+x^2)$ , $f(x)=x-x\ln x^2$ и $f(x)=(x^2+2x-8)e^{-x}$ на заданном интервале $[a; b]$ с заданным числом точек $n$ . Использовать модуль вычисления длины дуги по формуле $L \approx \sum_{k=0}^n \sqrt{(f(x_{k+1}) - f(x_k))^2 + h^2}$ , где $h = \frac{b-a}{n}$ , $x_k = a + kh$ .	ОПК-1 У1, У2, У3, У4, В1 ОПК-3 36, 38, У1, У2, У3, У5, В1
50	Составьте схему алгоритма и программу нахождения нормы заданной матрицы <b>A</b> размера $3 \times 5$ и заданной матрицы <b>B</b> размера $4 \times 4$ . Использовать модуль вычисления нормы матрицы $C=(c_{km})$ размера $N \times L$ по формуле $ C  = \max_{1 \leq k \leq N} \sum_{m=1}^L  c_{km} $ .	ОПК-1 У1, У2, У3, У4, В1 ОПК-3 36, 38, У1, У2, У3, У5, В1
51	Составьте схему алгоритма и программу вычислить $A^3$ и $B^4$ , где <b>A</b> – заданная матрица размера $4 \times 4$ , <b>B</b> – заданная матрица размера $3 \times 3$ . Использовать модуль вычисления произведения двух квадратных матриц.	ОПК-1 У1, У2, У3, У4, В1 ОПК-3 36, 38, У1, У2, У3, У5, В1

## ПОКАЗАТЕЛИ И КРИТЕРИИ ОЦЕНИВАНИЯ КОМПЕТЕНЦИЙ (Шкалы оценивания)

Результаты выполнения обучающимся заданий на зачете оцениваются по шкале «зачтено» - «не зачтено», на экзамене - по пятибалльной шкале.

В основе оценивания лежат критерии порогового и повышенного уровня характеристик компетенций или их составляющих частей, формируемых на учебных занятиях по дисциплине «Алгоритмизация и основы программирования» (Таблица 2.5 рабочей программы дисциплины).

**«Отлично» (5) / «зачтено»** – оценка соответствует повышенному уровню и выставляется обучающемуся, если он глубоко и прочно усвоил программный материал, исчерпывающе, последовательно, четко и логически стройно его излагает, умеет тесно увязывать теорию с практикой, свободно справляется с задачами, вопросами и другими видами применения знаний, причем не затрудняется с ответом при видоизменении заданий, использует в ответе материал монографической литературы, правильно обосновывает принятое решение, владеет разносторонними навыками и приемами выполнения практических задач.

**«Хорошо» (4) / «зачтено»** - оценка соответствует повышенному уровню и выставляется обучающемуся, если он твердо знает материал, грамотно и по существу излагает его, не допуская существенных неточностей в ответе на вопрос или выполнении заданий, правильно применяет теоретические положения при решении практических вопросов и задач, владеет необходимыми навыками и приемами их выполнения.

**«Удовлетворительно» (3) / «зачтено»** - оценка соответствует пороговому уровню и выставляется обучающемуся, если он имеет знания только основного материала, но не усвоил его деталей, допускает неточности, демонстрирует недостаточно правильные формулировки, нарушения логической последовательности в изложении программного материала, испытывает затруднения при выполнении практических работ.

**«Неудовлетворительно» (2) / «не зачтено»** - оценка выставляется обучающемуся, который не достигает порогового уровня, демонстрирует непонимание проблемы, не знает значительной части программного материала, допускает существенные ошибки, неуверенно, с большими затруднениями выполняет практические работы.