


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
«ИНФОРМАТИКА И ПРОГРАММИРОВАНИЕ»**

Уровень основной профессиональной образовательной программы:
бакалавриат

Направление подготовки: **02.03.03 Математическое обеспечение и администрирование информационных систем**

Направленность (профиль) подготовки: **Администрирование информационных систем**

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 дисциплины «Информатика и программирование» является формирование у обучающихся общекультурных и общепрофессиональных компетенций в процессе изучения информатики и 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. Дисциплина Б1.Б.9.«Информатика и программирование» относится к базовой части Блока 1.

2.2. Для изучения дисциплины «Информатика и программирование» необходимы знания, умения, навыки, формируемые дисциплиной предшествующего уровня образования:
- «Информатика и информационные технологии».

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

- «Объектно-ориентированные языки и системы»;
- «Архитектура вычислительных систем и компьютерных сетей»;
- «Проектирование реляционных БД»;
- «Параллельное программирование»;
- «Кроссплатформенное программирование»;
- «Функциональное программирование»;
- «Компьютерная графика».

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

Изучение данной дисциплины направлено на формирование у обучающихся следующих общепрофессиональных (ОПК) и профессиональных (ПК) компетенций:

№ п/п	Номер/индекс компетенции	Содержание компетенции (или ее части)	Перечень планируемых результатов обучения по дисциплине В результате изучения дисциплины обучающиеся должны:		
			Знать:	Уметь:	Владеть (навыками):
1	2	3	4	5	6
1	ОПК-3	готовность анализировать проблему и направления развития технологий программирования	<ul style="list-style-type: none"> • предмет и задачи исследования информатики как науки; • сущность и значение информации в развитии современного информационного общества; • этапы решения задачи на компьютере на основе парадигмы императивного программирования; • понятие алгоритма и его основные свойства; • основные понятия алгоритмической системы: представление информации в виде данных, система команд исполнителя, алгоритмический язык исполнителя; • основные средства записи и типы алгоритмов; • цели и составные части технологии нисходящего структурного проектирования алгоритмов и программ; • алгоритмические структуры, их основные свойства и приемы использования; • понятие алгоритмического модуля и его свойства; • принципы выделения и правила записи алгоритмических модулей; • семантику вызова алгоритмического модуля, в том числе, рекурсивного вызова, и механизмы реализации межмодульного интерфейса; • основные парадигмы программирования; • номенклатуру языков программирования различного уровня и их возможности. 	<ul style="list-style-type: none"> • при решении задач практической информатики на основе парадигмы императивного программирования конкретизировать и описать основные понятия, результаты других научных дисциплин, данные и их типы средствами записи алгоритмов; • разрабатывать и записывать алгоритмы и программы в соответствии с технологией нисходящего структурного проектирования; • записывать алгоритмы с помощью схем; • анализировать структуру алгоритмов; • описывать алфавит и основные синтаксические конструкции языков с помощью нормальных форм Бэкуса-Наура. 	<ul style="list-style-type: none"> • основными методами, способами и средствами переработки информации на основе парадигмы императивного программирования в соответствии с технологией нисходящего структурного проектирования.
2	ОПК-8	способность использовать знания методов проектирования и производства программного продукта, принципов построения, структуры и приемов работы с инструментальными средствами,	<ul style="list-style-type: none"> • составные части алгоритмического языка программирования (алфавит, синтаксис, семантика) и способы их описания; • концепцию типов данных; • структуры программ на языках программирования Бейсик, Паскаль и Си; • основные понятия алгоритмических языков программирования и их классификацию: идентификаторы, данные и их типы, операции, стандартные функции, выражения, операторы; • синтаксис и семантику основных операторов языков программирования Бейсик, Паскаль и Си (присваивания, ввода-вывода, ветвления, циклов), 	<ul style="list-style-type: none"> • разрабатывать и записывать алгоритмы и программы в соответствии с принципом структурности; • при решении учебной задачи конкретизировать и описать основные понятия, данные и их типы, организовать ввод и вывод, реализовать алгоритмические структуры средствами, по крайней мере, трех императивных языков программирования; • разрабатывать алгоритмы и программы обработки массивов; 	<ul style="list-style-type: none"> • навыками проектирования, ввода, отладки и тестирования программ в средах, по крайней мере, трех императивных систем программирования.

[Введите текст]

	<p>поддерживающими созданием программного обеспечения</p>	<p>способы реализации алгоритмических структур средствами языков программирования;</p> <ul style="list-style-type: none"> • основные синтаксические конструкции, используемые в языках программирования Бейсик, Паскаль и Си для описания и обработки данных типа массив; • прием флаг; • синтаксис и семантику реализации алгоритмического модуля средствами языков Бейсик, Паскаль и Си; • способы передачи параметров при реализации межмодульного интерфейса; • синтаксис типов запись с фиксированными полями, указатель и типизированный файл в языке Паскаль; • синтаксис типов структура, указатель и файл в языке Си; • понятия статического и динамического объекта программы на языках Паскаль и Си; • синтаксис и семантику основных средств управления использованием динамической памятью в языках Паскаль и Си; • структуру данных файл и основные действия с файлами последовательного доступа; • синтаксис и семантику основных средств обработки файлов в СП Turbo-Pascal и Turbo-C++ и последовательность их использования для организации ввода и вывода. 	<ul style="list-style-type: none"> • разрабатывать алгоритмы и программы обработки строк; • решать учебные задачи в соответствии с принципом модульности; • разрабатывать и использовать библиотеку пользователя в СП Turbo-Pascal и Turbo-C++; • разрабатывать простейшие алгоритмы и программы моделирования движения графического изображения; • использовать тип указатель для организации односвязного списка, очереди и стека; • решать учебные задачи по обработке данных типа запись, структура, файл и линейных динамических структур данных средствами систем программирования (СП) Turbo-Pascal и Turbo-C++. 	
--	---	---	---	--

2.5. Карта компетенций

КАРТА КОМПЕТЕНЦИЙ ДИСЦИПЛИНЫ					
НАИМЕНОВАНИЕ ДИСЦИПЛИНЫ: ИНФОРМАТИКА И ПРОГРАММИРОВАНИЕ					
Цель дисциплины		Целью освоения дисциплины «Информатика и программирование» является формирование у обучающихся общекультурных и общепрофессиональных компетенций в процессе изучения информатики и 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				
Общепрофессиональные компетенции:					
ОПК-3	готовность анализировать проблемы и направления развития технологий программирования	<p>Знать:</p> <ul style="list-style-type: none"> предмет и задачи исследования информатики как науки; сущность и значение информации в развитии современного информационного общества; этапы решения задачи на компьютере на основе парадигмы императивного программирования; понятие алгоритма и его основные свойства; основные понятия алгоритмической системы: представление информации в виде данных, система команд исполнителя, алгоритмический язык исполнителя; основные средства записи и типы алгоритмов; цели и составные части технологии нисходящего структурного проектирования алгоритмов и программ; алгоритмические структуры, их основные свойства и приемы использования; понятие алгоритмического модуля и его свойства; принципы выделения и правила записи алгоритмических модулей; семантику вызова алгоритмического модуля, в том числе, рекурсивного вызова, и механизмы реализации межмодульного интерфейса; основные парадигмы программирования; номенклатуру языков программирования различного уровня и их возможности. <p>Уметь:</p> <ul style="list-style-type: none"> при решении задач практической информатики на основе парадигмы императивного программирования конкретизировать и описать основные понятия, результаты других научных дисциплин, данные и их типы средствами записи алгоритмов; 	Путем проведения лекционных, лабораторных занятий, применения новых образовательных технологий, организации самостоятельной работы студентов	Лабораторные работы, зачет, экзамен	<p>Пороговый Способен решать стандартные задачи получения, хранения, переработки информации в соответствии с технологией нисходящего структурного проектирования.</p> <p>Повышенный Способен решать задачи получения, хранения, переработки информации повышенной сложности в соответствии с технологией нисходящего структурного проектирования.</p>

[Введите текст]

		<ul style="list-style-type: none"> • разрабатывать и записывать алгоритмы и программы в соответствии с технологией нисходящего структурного проектирования; • записывать алгоритмы с помощью схем; • анализировать структуру алгоритмов; • описывать алфавит и основные синтаксические конструкции языков с помощью нормальных форм Бэкуса-Наура. <p>Владеть:</p> <ul style="list-style-type: none"> • основными методами, способами и средствами переработки информации на основе парадигмы императивного программирования в соответствии с технологией нисходящего структурного проектирования. 			
ОПК-8	<p>способность использовать знания методов проектирования и производства программного продукта, принципов построения, структуры и приемов работы с инструментальными средствами, поддерживающими создание программного обеспечения</p>	<p>Знать:</p> <ul style="list-style-type: none"> • составные части алгоритмического языка программирования (алфавит, синтаксис, семантика) и способы их описания; • концепцию типов данных; • структуры программ на языках программирования Бейсик, Паскаль и Си; • основные понятия алгоритмических языков программирования и их классификацию: идентификаторы, данные и их типы, операции, стандартные функции, выражения, операторы; • синтаксис и семантику основных операторов языков программирования Бейсик, Паскаль и Си (присваивания, ввода-вывода, ветвления, циклов), способы реализации алгоритмических структур средствами языков программирования; • основные синтаксические конструкции, используемые в языках программирования Бейсик, Паскаль и Си для описания и обработки данных типа массив; • прием флаг; • синтаксис и семантику реализации алгоритмического модуля средствами языков Бейсик, Паскаль и Си; • способы передачи параметров при реализации межмодульного интерфейса; • синтаксис типов запись с фиксированными полями, указатель и типизированный файл в языке Паскаль; • синтаксис типов структура, указатель и файл в языке Си; • понятия статического и динамического объекта программы на языках Паскаль и Си; • синтаксис и семантику основных средств управления использованием динамической памятью в языках Паскаль и Си; • структуру данных файл и основные действия с файлами последовательного доступа; • синтаксис и семантику основных средств обработки файлов в СП Turbo- 	<p>Путем проведения лекционных, лабораторных занятий, применения новых образовательных технологий, организации самостоятельной работы студентов</p>	<p>Лабораторные работы, зачет, экзамен</p>	<p>Пороговый Способен решать стандартные задачи средствами императивных систем программирования</p> <p>Повышенный Способен решать задачи повышенной сложности средствами императивных систем программирования</p>

[Введите текст]

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

[Введите текст]

ОСНОВНАЯ ЧАСТЬ

1. Объем дисциплины и виды учебной работы

Вид учебной работы	Всего часов	Семестры		
		№ 1	№ 2	№ 3
		часов	часов	часов
1	2	3	4	5
1. Контактная работа обучающихся с преподавателем (по видам учебных занятий) (всего)	288	90	108	90
В том числе:				
Лекции (Л)	90	36	36	18
Лабораторные работы (ЛР)	198	54	72	72
2. Самостоятельная работа студента (всего)	288	90	108	90
В том числе:				
Изучение литературы и других источников	77	28	26	23
Подготовка к выполнению лабораторных работ	125	32	51	42
Подготовка к защите лабораторных работ	86	30	31	25
Контроль	72	-	36	36
Вид промежуточной аттестации	зачет (З)		+	
	экзамен (Э)			+
ИТОГО: общая трудоемкость	часов	648	180	252
	зач. ед.	18	5	7

2. Содержание дисциплины

2.1. Содержание разделов дисциплины

№ семестра	№ раздела	Наименование раздела дисциплины	Содержание раздела в дидактических единицах
1	2	3	4
1	1	Информатика как наука и как вид практической деятельности	История информатики, место информатики в системе наук. Теоретическая и прикладная информатика. Принципы фон Неймана как основа структуры персонального компьютера. Позиционные системы счисления. Представление в компьютере целых чисел. Представление в компьютере действительных чисел. Алгоритмизация и программирование как составные части информатики. Основные этапы компьютерного решения задач на основе парадигмы императивного программирования
1	2	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. Средства записи алгоритмов. Схемы алгоритмов

1	2	3	4
1	3	Основные понятия и простейшие средства алгоритмических языков программирования	Программирование как этап решения задачи на компьютере. Понятие языка программирования 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. Особенности 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	4	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	5	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	6	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.

[Введите текст]

1	2	3	4
			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	7	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	8	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	9	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	10	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

1	2	3	4
2	11	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	12	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 обработки 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	13	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	14	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

1	2	3	4
3	15	Парадигмы программирования	Основные парадигмы программирования: императивная, событийно-управляемая (объектно-ориентированная и параллельная), декларативная (функциональная и логическая). Архитектура и возможности семейства языков высокого уровня
3	16	Общая характеристика языков программирования семейства Си	Особенности языков программирования семейства Си. Структура простейшей программы на языке Си: раздел препроцессора и раздел функций. Синтаксис разделов программы, составной оператор. Понятие о библиотеке языка Си. Трансляция программ в СП (СП) семейства Си. Основные простые типы данных в языке Си: идентификаторы типов, модификаторы типов, диапазоны значений, синтаксис и семантика оператора резервирования памяти, инициализация переменных, операции и функции, преобразование типов. Некоторые особенности операций языка Си: перегрузка операции деления, операция присваивания и возможные варианты записи выражений с использованием операции присваивания, операции инкремента и декремента в префиксной и постфиксной форме, операции взятия адреса и содержимое по указанному адресу. Знакомство с операторами потокового ввода/вывода. Общая характеристика императивной СП Turbo-C++: основное меню, окна редактирования и результатов исполнения программы, диалоговые окна, встроенная система помощи
3	17	Разработка разветвляющихся алгоритмов и программ в СП Turbo-C++	Интерпретация данных целого типа как данных логического типа в языке Си. Операции отношения и логические операции. Организация развилки средствами языка Си: синтаксис и семантика условных оператора и операции. Операция запятая. Примеры организации развилки средствами Си. Синтаксис и семантика оператора переключателя. Организация посредством переключателя структуры выбор, пример. Разработка схем алгоритмов разветвляющейся структуры. Написание, ввод, отладка и тестирование программ разветвляющейся структуры в СП Turbo-C++
3	18	Разработка циклических алгоритмов и программ в СП Turbo-C++	Синтаксис и семантика операторов циклов в языке Си: циклы с предусловием и постусловием, цикл for. Особенности системной организации цикла for в СП Turbo-C++. Рекомендации по программированию циклов. Примеры циклических алгоритмов и программ. Разработка схем алгоритмов циклической структуры. Написание, ввод, отладка и тестирование программ циклической структуры в СП Turbo-C++
3	19	Ввод с клавиатуры и вывод на монитор средствами C++	Синтаксис и семантика операторов потокового ввода/вывода. Манипуляторы. Примеры организации потокового ввода/вывода. Синтаксис и семантика функций форматированного ввода/вывода. Синтаксис формата ввода, символы преобразования и модификаторы. Примеры организации форматированного ввода. Синтаксис формата вывода, управляющие символы. Примеры организации форматированного вывода. Написание, ввод, отладка и тестирование программ с форматированным выводом на монитор в СП Turbo-C++
3	20	Массивы в языке Си	Синтаксис и семантика оператора резервирования памяти под массив, примеры. Идентификатор массива как указатель на область памяти, захваченной под массив. Использование переменной типа массив в выражении с операцией присваивания. Особенности распределения памяти под элементы массива и инициализация массивов в СП семейства Си. Традиционная индексная и альтернативная, с помощью операции "*", адресация элементов массива, примеры. Примеры обработки массивов средствами языка Си. Разработка схем алгоритмов обработки массивов. Написание, ввод, отладка и тестирование программ обработки массивов в СП Turbo-C++
3	21	Строки в языке Си	Строки как массивы символов. Особенности инициализации и организации ввода строк в языке Си. Стандартные функции обработки строк в языке Си. Алгоритмы копирования, удаления и замены части строки средствами языка Си, примеры. Разработка схем алгоритмов обработки строк. Написание, ввод, отладка и тестирование программ обработки строковых данных в СП Turbo-C++

1	2	3	4
3	22	Реализация принципа модульности средствами языка Си	Синтаксис функции в языке Си. Формальные и фактические параметры. Способы передачи параметра простого типа. Рекомендации по использованию функций в программах на С++, примеры. Параметры массивы и параметры функции, примеры. Функции с аргументами по умолчанию, примеры. Перегрузка функций, пример. Классы памяти в языке Си. Рекомендации по организации простейшей библиотеки пользователя средствами СП Turbo-C++. Разработка схем алгоритмов в соответствии с принципом модульности. Написание, ввод, отладка и тестирование программ в соответствии с принципом модульности в СП Turbo-C++
3	23	Тип данных структура	Синтаксис типа структура в языках семейства Си. Оператор определения типа пользователя, пример. Определение типа структура с помощью тэга. Рекомендации по описанию переменной типа структура, примеры. Использование переменной типа структура в выражении с операцией присваивания. Член структуры (идентификатор с точкой). Разработка схем алгоритмов обработки данных типа структура. Написание, ввод, отладка и тестирование программ обработки данных типа структура в соответствии с принципом модульности в СП Turbo-C++
3	24	Динамические переменные в языках семейства Си	Синтаксис типа указатель (ссылочный тип) в языках семейства Си. Рекомендации по описанию переменной типа указатель. Синтаксис и семантика оператора захвата памяти под динамическую переменную. Идентификатор динамической переменной в языках семейства Си (идентификатор со звездочкой). Синтаксис и семантика освобождения области памяти динамической переменной. Синтаксис и семантика операторов захвата и освобождения области памяти для динамического массива. Использование типа указатель для организации односвязного списка, определение типа элемента односвязного списка с помощью тэга. Адресация члена динамической переменной типа структура с помощью литеры алфавита “->” (идентификатор со стрелкой). Рекомендации по организации вставки (добавления), удаления и поиска элементов линейных динамических структур данных. Примеры алгоритмов создания и обработки линейных динамических структур данных и соответствующих программ на языке С++. Разработка схем алгоритмов создания и обработки линейных динамических структур данных. Написание, ввод, отладка и тестирование программ создания и обработки линейных динамических структур данных в соответствии с принципом модульности в СП Turbo-C++
3	25	Тип файл в СП Turbo-C++	Организация файлов данных в языках семейства Си. Синтаксис и семантика захвата памяти под указатель на буфер файла. Синтаксис и семантика функции открытия файла. Атрибуты режима по умолчанию для записи (вывода), чтения (ввода) и добавления данных. Атрибуты режимов текстового и бинарного файла данных. Синтаксис и семантика функций форматированной записи данных в файл и форматированного чтения данных из файла. Функция конец файла в СП Turbo-C++. Рекомендации для СП Turbo-C++ по корректному открытию файла и чтению данных из файла, примеры. Синтаксис и семантика функции закрытия файла. Примеры алгоритма и программы обработки файла данных, в которых для хранения промежуточных результатов используется линейная динамическая структура. Разработка схем алгоритмов обработки файлов данных. Написание, ввод, отладка и тестирование программ создания и текстовых файлов в соответствии с принципом модульности в СП Turbo-C++

2.2. Разделы дисциплины, виды учебной деятельности и формы контроля

№ семестра	№ раздела	Наименование раздела 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	Информатика как наука и как вид практической деятельности	2		4	6	1 неделя: лабораторная работа №1
	2	Алгоритмическая система и ее составные части			2	2	
	3	Основные понятия и простейшие средства алгоритмических языков программирования	3	3	6	12	
	4	Технология нисходящего структурного проектирования. Разработка разветвляющихся алгоритмов и программ в соответствии с принципом структурности	8	12	20	40	2-5 неделя: лабораторная работа №2
	5	Разработка циклических алгоритмов и программ в соответствии с принципом структурности	9	18	22	47	6-7 неделя: лабораторная работа №3 8-9 неделя: лабораторная работа №4 10-11 неделя: лабораторная работа №5
	6	Ввод с клавиатуры и вывод на монитор средствами алгоритмических языков	4	9	12	25	12-14 неделя: лабораторная работа №6
	7	Структурированный тип данных массив	10	12	24	36	15-18 неделя: лабораторная работа №7
ИТОГО 1 семестр			36	54	90	180	Зачет
2	8	Символьный и строковый типы данных	6	20	26	52	1-2 неделя: лабораторная работа №8 3-5 неделя: лабораторная работа №9
	9	Разработка алгоритмов в соответствии с принципом модульности	2	8	10	20	6-7 неделя: лабораторная работа №10
	10	Реализация принципа модульности средствами алгоритмических языков программирования	10	20	30	60	8-12 неделя: лабораторная работа №11
	11	Структурированный тип данных запись	4	4	8	16	13 неделя: лабораторная работа №12
	12	Тип данных указатель	8	8	16	32	15-16 неделя: лабораторная работа №13
	13	Структурированный тип данных файл	4	4	8	16	

[Введите текст]

1	2	3	4	5	6	7	8
	14	Обзор графических возможностей систем программирования (СП) Qbasic и Turbo-Pascal	2	8	10	20	17-18 неделя: лабораторная работа №14
		Контроль				36	Экзамен
		ИТОГО 2 семестр	36	72	108	252	
3	15	Парадигмы программирования	1		1	2	1 неделя: лабораторная работа №15
	16	Общая характеристика языков программирования семейства Си	1	2	3	6	
	17	Разработка разветвляющихся алгоритмов и программ в СП Turbo-C++	2	10	12	24	2-3 неделя: лабораторная работа №16
	18	Разработка циклических алгоритмов и программ в СП Turbo-C++	2	8	10	20	4 неделя: лабораторная работа №17 5 неделя: лабораторная работа №18
	19	Ввод с клавиатуры и вывод на монитор средствами C++	1	8	9	18	6-7 неделя: лабораторная работа №19
	20	Массивы в языке Си	2	8	10	20	8-9 неделя: лабораторная работа №20
	21	Строки в языке Си	2	8	10	20	10-11 неделя: лабораторная работа №21
	22	Реализация принципа модульности средствами языка Си	3	16	19	38	12-16 неделя: лабораторная работа №22
	23	Тип данных структура	1	4	5	10	
	24	Динамические переменные в языках семейства Си	2	6	8	16	17 неделя: лабораторная работа №23 18 неделя: лабораторная работа №24
	25	Тип файл в СП Turbo-C++	1	2	3	6	
		Контроль				36	Экзамен
		ИТОГО 3 семестр	18	72	90	216	
		ИТОГО	90	198	288	648	

2.3. Лабораторный практикум

№ семестра	№ раздела	Наименование раздела дисциплины	Наименование лабораторных работ	Всего часов
1	2	3	4	5
1	1	Информатика как наука и как вид практической деятельности	ЛР вводная. Знакомство с императивными системами программирования QBasic и Turbo-Pascal. Ввод, отладка и тестирование простейших программ	1

[Введите текст]

1	2	3	4	5
1	2	Алгоритмическая система и ее составные части	ЛР №1. <i>Написание, ввод, отладка и тестирование программ линейной структуры</i>	3
	3	Основные понятия и простейшие средства алгоритмических языков программирования		
1	4	Технология нисходящего структурного проектирования. Разработка разветвляющихся алгоритмов и программ в соответствии с принципом структурности	ЛР №2. <i>Написание, ввод, отладка и тестирование алгоритмов и программ разветвляющейся структуры</i>	6
			ЛР №3. <i>Написание, ввод, отладка и тестирование разветвляющихся алгоритмов и программ с использованием структуры выбор</i>	6
1	5	Разработка циклических алгоритмов и программ в соответствии с принципом структурности	ЛР №4. <i>Написание, ввод, отладка и тестирование циклических алгоритмов и программ</i>	9
			ЛР №5. <i>Написание, ввод, отладка и тестирование циклических алгоритмов и программ целочисленной арифметики</i>	9
1	6	Ввод с клавиатуры и вывод на монитор средствами алгоритмических языков	ЛР №6. <i>Написание, ввод, отладка и тестирование циклических алгоритмов и программ с форматированным выводом на монитор</i>	9
1	7	Структурированный тип данных массив	ЛР №7. <i>Написание, ввод, отладка и тестирование алгоритмов и программ обработки числовых массивов</i>	12
ИТОГО 1 семестр				54
2	8	Символьный и строковый типы данных	ЛР №8. <i>Написание, ввод, отладка и тестирование алгоритмов и программ обработки строк</i>	8
			ЛР №9. <i>Написание, ввод, отладка и тестирование алгоритмов и программ обработки массивов строк</i>	12
2	9	Разработка алгоритмов в соответствии с принципом модульности	ЛР №10. <i>Написание, ввод, отладка и тестирование алгоритмов и программ в соответствии с принципом модульности</i>	8
2	10	Реализация принципа модульности средствами алгоритмических языков программирования	ЛР №11. <i>Написание, ввод, отладка и тестирование алгоритмов и программ обработки массивов в соответствии с принципом модульности</i>	12
2	11	Структурированный тип данных запись	ЛР №12. <i>Написание, ввод, отладка и тестирование алгоритмов и программ создания и обработки односвязных списков, записей и типизированных файлов в СП Turbo-Pascal</i>	16
	12	Тип данных указатель		
2	13	Структурированный тип данных файл	ЛР №13. <i>Написание, ввод, отладка и тестирование алгоритмов и программ длинной арифметики в соответствии с принципом модульности</i>	8
2	14	Обзор графических возможностей систем программирования (СП) Qbasic и Turbo-Pascal	ЛР №14. <i>Написание, ввод, отладка и тестирование алгоритмов и программ моделирования движения графического изображения в соответствии с принципом модульности</i>	8
ИТОГО 2 семестр				72
3	16	Общая характеристика языков программирования семейства Си	ЛР №15. <i>Знакомство с императивной системой программирования Turbo-C++. Написание, ввод, отладка и тестирование алгоритмов и программ разветвляющейся структуры в СП Turbo-C++</i>	4

1	2	3	4	5
3	17	Разработка разветвляющихся алгоритмов и программ в СП Turbo-C++	ЛР №16. Написание, ввод, отладка и тестирование разветвляющихся алгоритмов и программ с использованием структуры выбор в СП Turbo-C++	8
3	18	Разработка циклических алгоритмов и программ в СП Turbo-C++	ЛР №17. Написание, ввод, отладка и тестирование циклических алгоритмов и программ в СП Turbo-C++	4
			ЛР №18. Написание, ввод, отладка и тестирование циклических алгоритмов и программ целочисленной арифметики в СП Turbo-C++	4
3	19	Ввод с клавиатуры и вывод на монитор средствами C++	ЛР №19. Написание, ввод, отладка и тестирование циклических алгоритмов и программ с форматированным выводом на монитор в СП Turbo-C++	8
3	20	Массивы в языке Си	ЛР №20. Написание, ввод, отладка и тестирование алгоритмов и программ обработки массивов в СП Turbo-C++	8
3	21	Строки в языке Си	ЛР №21. Написание, ввод, отладка и тестирование алгоритмов и программ обработки строковых данных в СП Turbo-C++	8
3	22	Реализация принципа модульности средствами языка Си	ЛР №22. Написание, ввод, отладка и тестирование алгоритмов и программ в соответствии с принципом модульности в СП Turbo-C++	8
3	23	Тип данных структура	ЛР №23. Написание, ввод, отладка и тестирование алгоритмов и программ создания и обработки односвязных списков, структур и файлов в СП Turbo-C++	12
	24	Динамические переменные в языках семейства Си		
	25	Тип файл в СП Turbo-C++		
		ИТОГО 3 семестр		72
		ИТОГО		198

3. Самостоятельная работа студента

3.1. Виды СРС

№ семестра	№ раздела	Наименование раздела дисциплины	Виды СРС	Всего часов
1	2	3	4	5
1	1	Информатика как наука и как вид практической деятельности	Изучение литературы и других источников	2
			Подготовка к выполнению лабораторной работы	2
1	2	Алгоритмическая система и ее составные части	Изучение литературы и других источников	1
			Подготовка к выполнению лабораторной работы	1
1	3	Основные понятия и простейшие средства алгоритмических языков программирования	Изучение литературы и других источников	2
			Подготовка к выполнению лабораторной работы	2
			Подготовка к защите лабораторной работы	2
1	4	Технология нисходящего структурного проектирования. Разработка разветвляющихся алгоритмов и программ в соответствии с принципом структурности	Изучение литературы и других источников (ЛР №2)	3
			Подготовка к выполнению лабораторной работы (ЛР №2)	3
			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

[Введите текст]

1	5	Разработка циклических алгоритмов и программ в соответствии с принципом структурности	Изучение литературы и других источников (ЛР №4)	3
			Подготовка к выполнению лабораторной работы (ЛР №4)	4
			Подготовка к защите лабораторной работы (ЛР №4)	4
			Изучение литературы и других источников (ЛР №5)	3
			Подготовка к выполнению лабораторной работы (ЛР №5)	4
			Подготовка к защите лабораторной работы (ЛР №5)	4
1	6	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)	2
			Подготовка к защите лабораторной работы (ввод и вывод в СП Turbo-Pascal)	2
1	7	Структурированный тип данных массив	Изучение литературы и других источников (основная и дополнительная литература)	3
			Изучение литературы и других источников (конспекты лекций и ресурсы компьютерных сетей)	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
			ИТОГО 1 семестр	90
2	8	Символьный и строковый типы данных	Изучение литературы и других источников (ЛР №8)	3
			Подготовка к выполнению лабораторной работы (разработка схемы алгоритма ЛР №8)	3
			Подготовка к выполнению лабораторной работы (разработка программ ЛР №8)	3
			Подготовка к защите лабораторной работы (ЛР №8)	4
			Изучение литературы и других источников (ЛР №9)	3
			Подготовка к выполнению лабораторной работы (разработка схемы алгоритма ЛР №9)	4
			Подготовка к выполнению лабораторной работы (разработка программ ЛР №9)	3
			Подготовка к защите лабораторной работы (ЛР №9)	3
2	9	Разработка алгоритмов в соответствии с принципом модульности	Изучение литературы и других источников	2
			Подготовка к выполнению лабораторной работы (разработка схемы алгоритм)	2
			Подготовка к выполнению лабораторной работы (разработка программ)	2
			Подготовка к защите лабораторной работы (свойства алгоритмического модуля)	2
			Подготовка к защите лабораторной работы (оформление алгоритмического модуля)	2
2	10	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)	3
			Подготовка к защите лабораторной работы (реализация принципа модульности в СП Qbasic ЛР №10)	2
			Подготовка к защите лабораторной работы (реализация принципа модульности в СП Turbo-Pascal ЛР №10)	3

[Введите текст]

			Изучение литературы и других источников (основная и дополнительная литература ЛР №11)	2
			Изучение литературы и других источников (конспекты лекций и ресурсы компьютерных сетей ЛР №11)	3
			Подготовка к выполнению лабораторной работы (разработка схемы алгоритма ЛР №11)	3
			Подготовка к выполнению лабораторной работы (разработка программ ЛР №11)	3
			Подготовка к защите лабораторной работы (ввод из блока данных в СП Qbasic ЛР №11)	2
			Подготовка к защите лабораторной работы (типизированные константы в СП Turbo-Pascal ЛР №11)	3
2	11	Структурированный тип данных запись	Изучение литературы и других источников (основная и дополнительная литература)	2
			Изучение литературы и других источников (конспекты лекций и ресурсы компьютерных сетей)	2
			Подготовка к выполнению лабораторной работы (разработка структуры данных)	2
			Подготовка к выполнению лабораторной работы (разработка описаний)	2
2	12	Тип данных указатель	Подготовка к выполнению лабораторной работы (разработка схем алгоритмов)	2
			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)	2
			Изучение литературы и других источников (основная и дополнительная литература)	2
			Изучение литературы и других источников (конспекты лекций и ресурсы компьютерных сетей)	2
2	13	Структурированный тип данных файл	Подготовка к выполнению лабораторной работы	2
			Подготовка к защите лабораторной работы (вывод в файл)	2
			Подготовка к защите лабораторной работы (ввод из файла)	2
			Изучение литературы и других источников	2
2	14	Обзор графических возможностей систем программирования (СП) Qbasic и Turbo-Pascal	Подготовка к выполнению лабораторной работы (разработка схем алгоритмов)	2
			Подготовка к выполнению лабораторной работы (разработка программы для СП Qbasic)	3
			Подготовка к выполнению лабораторной работы (разработка программы для СП Turbo-Pascal)	3
			Подготовка к защите лабораторной работы	2
			Изучение литературы и других источников по теме «Структурированный тип данных запись»	3
			Изучение литературы и других источников по теме «Тип данных указатель»	3
			Изучение литературы и других источников по теме «Структурированный тип данных файл»	3
			Разбор практических заданий по теме «Принцип модульности»	3
		ИТОГО 2 семестр		108
3	15	Парадигмы программирования	Изучение литературы и других источников	1
3	16	Общая характеристика языков программирования семейства Си	Подготовка к выполнению лабораторной работы	1
			Подготовка к защите лабораторной работы	2
3	17	Разработка разветв-	Изучение литературы и других источников (основная и дополнительная литература)	2

[Введите текст]

		ляющихся алгоритмов и программ в СП Turbo-C++	Изучение литературы и других источников (конспекты лекций и ресурсы компьютерных сетей)	2
			Подготовка к выполнению лабораторной работы (разработка схем алгоритмов)	3
			Подготовка к выполнению лабораторной работы (разработка програм)	3
			Подготовка к защите лабораторной работы	2
3	18	Разработка циклических алгоритмов и программ в СП Turbo-C++	Изучение литературы и других источников	2
			Подготовка к выполнению лабораторной работы (разработка схем алгоритмов)	2
			Подготовка к выполнению лабораторной работы (разработка програм)	2
			Подготовка к защите лабораторной работы (циклы с предусловием и постусловием)	2
			Подготовка к защите лабораторной работы (цикл for)	2
3	19	Ввод с клавиатуры и вывод на монитор средствами C++	Изучение литературы и других источников	3
			Подготовка к выполнению лабораторной работы	3
			Подготовка к защите лабораторной работы	3
3	20	Массивы в языке Си	Изучение литературы и других источников	2
			Подготовка к выполнению лабораторной работы (разработка схемы алгоритма)	3
			Подготовка к выполнению лабораторной работы (разработка програм)	3
			Подготовка к защите лабораторной работы	2
3	21	Строки в языке Си	Изучение литературы и других источников	2
			Подготовка к выполнению лабораторной работы (разработка схем алгоритмов)	3
			Подготовка к выполнению лабораторной работы (разработка програм)	3
			Подготовка к защите лабораторной работы	2
3	22	Реализация принципа модульности средствами языка Си	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 (разработка схем алгоритмов)	3
			Подготовка к выполнению лабораторной работы (разработка программных модулей)	3
			Подготовка к выполнению лабораторной работы (разработка программы)	3
			Подготовка к защите лабораторной работы (способы передачи параметров)	2
			Подготовка к защите лабораторной работы (передача параметров массивов и функций)	3
3	23	Тип данных структура	Изучение литературы и других источников	2
			Подготовка к выполнению лабораторной работы	3
3	24	Динамические переменные в языках семейства Си	Подготовка к защите лабораторной работы	2
			Изучение литературы и других источников	2
			Подготовка к выполнению лабораторной работы (разработка схемы алгоритма)	2
			Подготовка к выполнению лабораторной работы (разработка програм)	2
3	25	Тип файл в СП Turbo-C++	Подготовка к защите лабораторной работы	3
		ИТОГО 3 семестр		90
		ИТОГО		288

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	18
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	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+

Семестр № 3

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

3.3.1. Контрольные работы/рефераты *не предусмотрены*

4. Оценочные средства для контроля успеваемости и результатов освоения учебной дисциплины (см. Фонд оценочных средств)

4.2. 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. Основы информатики [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, 2, 15	1, 3	14	
2.	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.	1-14	1, 2	14	

[Введите текст]

3.	Новичков В.С., Пылькин А.Н. Начала программирования на языке QBasic [Текст]: Учебное пособие. М.: Горячая линия-Телеком, 2007	1-14	1, 2	14	
4.	Каширин И.Ю., Новичков В.С. От С к С++ [Текст]: Учебное пособие для вузов.- М.: Горячая линия – Телеком, 2005	16-25	3	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.	1-14	1, 2	7	
2.	Новожилов, О. П. Информатика [Электронный ресурс] : учебник для прикладного бакалавриата / О. П. Новожилов. – 3-е изд., перераб. и доп. – Москва : Юрайт, 2017. – 619 с. Режим доступа: https://www.biblio-online.ru/book/FEE705BC-11CB-46EB-810E-2634A4DE5E46 (дата обращения: 30.08.2019).	1, 2, 15	1, 3	ЭБС	
3.	Савич, У. Программирование на С++ [Текст] / У. Савич. – 4-е изд. – Санкт-Петербург : Питер, 2004. – 781 с.	16-25	3	5	

5.3. Базы данных, информационно-справочные и поисковые системы

1. VOOR.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

[Введите текст]

1. eLIBRARY.RU [Электронный ресурс] : научная электронная библиотека. – Режим доступа: <http://elibrary.ru/defaultx.asp>, свободный (дата обращения: 30.08.2019).
2. Единая коллекция цифровых образовательных ресурсов [Электронный ресурс] : федеральный портал. – Режим доступа: <http://school-collection.edu.ru/>, свободный (дата обращения: 30.08.2019).
3. Единое окно доступа к образовательным ресурсам [Электронный ресурс] : федеральный портал. – Режим доступа: <http://window.edu.ru/>, свободный (дата обращения: 30.08.2019).
4. Интернет Университет Информационных технологий. [Электронный ресурс] : сайт. – Режим доступа: <http://www.intuit.ru/>, свободный (дата обращения 30.08.2019).
5. Петров Д.Н. Парадигмы программирования. [Электронный ресурс] : сайт. – Режим доступа: <http://dnppetov.narod.ru/>, свободный (дата обращения 30.08.2019).
6. Портал естественных наук. [Электронный ресурс] : сайт. – Режим доступа: <http://e-science11.ru>, свободный (дата обращения 30.08.2019).
7. Портал для программистов и администраторов информационных систем. [Электронный ресурс] : сайт. – Режим доступа: <http://www.coderpost.net/>, свободный (дата обращения 30.08.2019).
8. Российский общеобразовательный портал [Электронный ресурс] : образовательный портал. – Режим доступа: <http://www.school.edu.ru/>, свободный (дата обращения: 30.08.2019).
9. Сервер Информационных Технологий [Электронный ресурс] : сайт. – Режим доступа: <http://citforum.ru/>, свободный (дата обращения 30.08.2019).
10. Сайт программирования. [Электронный ресурс] : сайт. – Режим доступа: <http://www.cyberguru.ru/>, свободный (дата обращения 30.08.2019).
11. Сайт программирования в среде Delphi. [Электронный ресурс]: сайт. – Режим доступа: <http://www.delphisources.ru/>, свободный (дата обращения 30.08.2019).
12. Федеральный центр информационно-образовательных ресурсов [Электронный ресурс] : Единое окно доступа к образовательным ресурсам. – Режим доступа: <http://fcior.edu.ru>, свободный (дата обращения: 30.08.2019).

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, Microsoft Office или аналог, системы программирования (СП) Qbasic, Turbo-Pascal и Turbo-C++.

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 алгоритмов которых отличны от структур оформленных схем. Пере-</p>

	чень примерных контрольных заданий и вопросов для защиты лабораторных работ приведен в разделе 11.2.
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 по дисциплине

Для организации учебной и самостоятельной работы обучающихся 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 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. Система программирования Turbo-C++ (свободно распространяемое ПО)
7. Архиватор 7-zip (свободно распространяемое ПО)
8. Браузер изображений FastStoneImageViewer (свободно распространяемое ПО)
9. PDF ридер FoxitReader (свободно распространяемое ПО)
10. Медиа проигрыватель VLC media player (свободно распространяемое ПО)
11. Запись дисков ImageBurn (свободно распространяемое ПО)
12. DJVU браузер DjVu Browser Plug-in (свободно распространяемое ПО)

**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ
АТТЕСТАЦИИ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ
«Информатика и программирование»**

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	Информатика как наука и как вид практической деятельности	ОПК-3 ОПК-8	Зачет, экзамен
2	Алгоритмическая система и ее составные части		
3	Основные понятия и простейшие средства алгоритмических языков программирования		
4	Технология нисходящего структурного проектирования. Разработка разветвляющихся алгоритмов и программ в соответствии с принципом структурности		
5	Разработка циклических алгоритмов и программ в соответствии с принципом структурности		
6	Ввод с клавиатуры и вывод на монитор средствами алгоритмических языков		
7	Структурированный тип данных массив		
8	Символьный и строковый типы данных		
9	Разработка алгоритмов в соответствии с принципом модульности		
10	Реализация принципа модульности средствами алгоритмических языков программирования		
11	Структурированный тип данных запись		
12	Тип данных указатель		
13	Структурированный тип данных файл		
14	Обзор графических возможностей систем программирования (СП) Qbasic и Turbo-Pascal		
15	Парадигмы программирования		
16	Общая характеристика языков программирования семейства Си		
17	Разработка разветвляющихся алгоритмов и программ в СП Turbo-C++		
18	Разработка циклических алгоритмов и программ в СП Turbo-C++		
19	Ввод с клавиатуры и вывод на монитор средствами C++		
20	Массивы в языке Си		
21	Строки в языке Си		
22	Реализация принципа модульности средствами языка Си		
23	Тип данных структура		
24	Динамические переменные в языках семейства Си		
25	Тип файл в СП Turbo-C++		

ТРЕБОВАНИЯ К РЕЗУЛЬТАТАМ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ

Индекс компетенции	Содержание компетенции	Элементы компетенции	Индекс элемента
1	2	3	4
ОПК-3	готовность анализировать проблемы и направления развития технологий программирования	знать	
		31 предмет и задачи исследования информатики как науки;	ОПК-3 31
		32 сущность и значение информации в развитии современного информационного общества;	ОПК-3 32
		33 этапы решения задачи на компьютере на основе парадигмы императивного программирования;	ОПК-3 33
		34 понятие алгоритма и его основные свойства;	ОПК-3 34
		35 основные понятия алгоритмической системы: представление информации в виде данных, система команд исполнителя, алгоритмический язык исполнителя;	ОПК-3 35
		36 основные средства записи и типы алгоритмов;	ОПК-3 36
		37 цели и составные части технологии нисходящего структурного проектирования алгоритмов и программ;	ОПК-3 37
		38 алгоритмические структуры, их основные свойства и приемы использования;	ОПК-3 38
		39 понятие алгоритмического модуля и его свойства;	ОПК-3 39
		310 принципы выделения и правила записи алгоритмических модулей;	ОПК-3 310
		311 семантику вызова алгоритмического модуля, в том числе, рекурсивного вызова, и механизмы реализации межмодульного интерфейса;	ОПК-3 311
		312 основные парадигмы программирования;	ОПК-3 312
		313 номенклатуру языков программирования различного уровня и их возможности.	ОПК-3 313
		уметь	
У1 при решении задач практической информатики на основе парадигмы императивного программирования конкретизировать и описать основные понятия, результаты других научных дисциплин, данные и их типы средствами записи алгоритмов;	ОПК-3 У1		
У2 разрабатывать и записывать алгоритмы и программы в соответствие с технологией нисходящего структурного проектирования;	ОПК-3 У2		
У3 записывать алгоритмы с помощью схем;	ОПК-3 У3		
У4 анализировать структуру алгоритмов;	ОПК-3 У4		
У5 описывать алфавит и основные синтаксические конструкции языков с помощью нормальных форм Бэкуса-Наура.	ОПК-3 У5		
владеть			
В1 основными методами, способами и средствами переработки информации на основе парадигмы императивного программирования в соответствии с технологией нисходящего структурного проектирования.	ОПК-3 В1		

1	2	3	4
ОПК-8	способность использовать знания методов проектирования и производства программного продукта, принципов построения, структуры и приемов работы с инструментальными средствами, поддерживающими создание программного обеспечения	знать	
		31 составные части алгоритмического языка программирования (алфавит, синтаксис, семантика) и способы их описания;	ОПК-8 31
		32 концепцию типов данных;	ОПК-8 32
		33 структуры программ на языках программирования Бейсик, Паскаль и Си;	ОПК-8 33
		34 основные понятия алгоритмических языков программирования и их классификацию: идентификаторы, данные и их типы, операции, стандартные функции, выражения, операторы;	ОПК-8 34
		35 синтаксис и семантику основных операторов языков программирования Бейсик, Паскаль и Си (присваивания, ввода-вывода, ветвления, циклов), способы реализации алгоритмических структур средствами языков программирования;	ОПК-8 35
		36 основные синтаксические конструкции, используемые в языках программирования Бейсик, Паскаль и Си для описания и обработки данных типа массив;	ОПК-8 36
		37 прием флаг;	ОПК-8 37
		38 синтаксис и семантику реализации алгоритмического модуля средствами языков Бейсик, Паскаль и Си;	ОПК-8 38
		39 способы передачи параметров при реализации межмодульного интерфейса;	ОПК-8 39
		310 синтаксис типов запись с фиксированными полями, указатель и типизированный файл в языке Паскаль;	ОПК-8 310
		311 синтаксис типов структура, указатель и файл в языке Си;	ОПК-8 311
		312 понятия статического и динамического объекта программы на языках Паскаль и Си;	ОПК-8 312
		313 синтаксис и семантику основных средств управления использованием динамической памятью в языках Паскаль и Си;	ОПК-8 313
		314 структуру данных файл и основные действия с файлами последовательного доступа;	ОПК-8 314
315 синтаксис и семантику основных средств обработки файлов в СП Turbo-Pascal и Turbo-C++ и последовательность их использования для организации ввода и вывода.	ОПК-8 315		
уметь			
У1 разрабатывать и записывать алгоритмы и программы в соответствие с принципом структурности;	ОПК-8 У1		

1	2	3	4
		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, трех императивных языков программирования;	ОПК-8 У2
		У3 разрабатывать алгоритмы и программы обработки массивов;	ОПК-8 У3
		У4 разрабатывать алгоритмы и программы обработки строк;	ОПК-8 У4
		У5 решать учебные задачи в соответствии с принципом модульности;	ОПК-8 У5
		У6 разрабатывать и использовать библиотеку пользователя в СП Turbo-Pascal и Turbo-C++;	ОПК-8 У6
		У7 разрабатывать простейшие алгоритмы и программы моделирования движения графического изображения;	ОПК-8 У7
		У8 использовать тип указатель для организации односвязного списка, очереди и стека;	ОПК-8 У8
		У9 решать учебные задачи по обработке данных типа запись, структура, файл и линейных динамических структур данных средствами систем программирования (СП) Turbo-Pascal и Turbo-C++.	ОПК-8 У9
		владеть	
		В1 навыками проектирования, ввода, отладки и тестирования программ в средах, по крайней мере, трех императивных систем программирования.	ОПК8 В1

КОМПЛЕКТ ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ (ЗАЧЕТ 1 семестр)

№	Содержание оценочного средства	Индекс оцениваемой компетенции и ее элементов
1	Проанализируйте темпы изменения содержания информатики как науки и как вида человеческой деятельности по сравнению с другими науками. Сделайте выводы о необходимости самоорганизации и самообразования для успешной деятельности в сфере информационных технологий.	ОПК-3 31, 32
2	Охарактеризуйте алгоритмизацию и программирование как составные части информатики. Приведите примеры простейших алгоритмов и программ.	ОПК-3 36, У2
3	Кратко охарактеризуйте основные этапы компьютерного решения задач на основе парадигмы императивного программирования. Раскройте значение этапов алгоритмизации и программирования.	ОПК-3 33, 34, У2
4	Проанализируйте необходимость представления информации в виде данных при использовании технических средств в информационных процессах. Приведите примеры ввода/вывода данных.	ОПК-3 32, 33, 312
5	Охарактеризуйте основные элементы системы, являющейся исполнителем программы на алгоритмическом языке программирования. Приведите примеры систем программирования с различными типами трансляторов.	ОПК-3 35, 312, 313

[Введите текст]

6	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-3 36 ОПК-8 34
7	Дайте определение понятию идентификатор, приведите соответствующую метаформулу. Расскажите, как используются идентификаторы в тексте программы на алгоритмическом языке, приведите примеры.	ОПК-3 У5 ОПК-8 34
8	Охарактеризуйте понятие переменная. С какими другими основными понятиями алгоритмизации и программирования связано это понятие?	ОПК-8 32, 34
9	Охарактеризуйте концепцию типов данных в языке Бейсик. Какие характеристики целых типов данных языка Бейсик Вы знаете? Проанализируйте особенности целочисленной компьютерной арифметики.	ОПК-8 32, 34
10	Охарактеризуйте концепцию типов данных в языке Бейсик. Какие характеристики действительных типов данных языка Бейсик Вы знаете? Проанализируйте особенности действительной компьютерной арифметики.	ОПК-8 32, 34
11	Опишите алгоритм определения типа переменной при исполнении программы на языке Бейсик. Приведите метаформулу, определяющую синтаксис оператора определения типа в языке Бейсик.	ОПК-3 У5 ОПК-8 32, 34
12	Охарактеризуйте концепцию типов данных в языке Паскаль. Какие характеристики числовых типов данных языка Паскаль Вы знаете?	ОПК-8 32, 34
13	Проведите сравнительный анализ структур программ на языках Бейсик и Паскаль. Приведите примеры простейших программ на этих языках.	ОПК-3 У5 ОПК-8 33, 34
14	Охарактеризуйте понятие идентификатор. Какие классификации операторов алгоритмических языков программирования вы знаете? Классифицируйте оператор присваивания. Опишите синтаксис и семантику этого оператора.	ОПК-3 У5 ОПК-8 34, 35
15	Охарактеризуйте цели и составные части технологии нисходящего структурного проектирования. Приведите простейший пример, иллюстрирующий эту технологию разработки алгоритмов и программ.	ОПК-3 37, У2, В1 ОПК-8 У1
16	Опишите синтаксис и семантику оператора ввода с клавиатуры в СП Qbasic. Приведите примеры организации ввода с клавиатуры данных различных типов.	ОПК-3 У5 ОПК-8 34, 35
17	Опишите синтаксис и семантику оператора ввода с клавиатуры в СП Turbo-Pascal. Приведите примеры организации ввода с клавиатуры данных различных типов.	ОПК-3 У5 ОПК-8 34, 35

[Введите текст]

18	Опишите синтаксис и семантику оператора вывода на монитор в СП Qbasic. Приведите примеры организации вывода на монитор данных различных типов.	ОПК-3 У5 ОПК-8 34, 35
19	Опишите синтаксис и семантику оператора форматированного вывода на монитор в Turbo-Pascal. Приведите примеры организации форматированного вывода на монитор данных различных типов.	ОПК-3 У5 ОПК-8 34, 35
20	Опишите синтаксис и семантику оператора вывода на монитор в СП Qbasic. Приведите примеры организации вывода на монитор данных различных типов.	ОПК-3 У5 ОПК-8 34, 35
21	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-3 У1, У2, У3, У4, В1 ОПК-8 34, 35, У1, У2, В1
22	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-3 У1, У2, У3, У4, В1 ОПК-8 34, 35, У1, У2, В1
23	Составьте схему алгоритма и программу на алгоритмическом языке: для трех заданных чисел a , b и c , среди которых не более одного отрицательного, найти среднее геометрическое $\sqrt[3]{abc}$, если все они положительные, и разность между произведением двух положительных и третьим числом в противном случае.	ОПК-3 У1, У2, У3, У4, В1 ОПК-8 34, 35, У1, У2, В1
24	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-3 У1, У2, У3, У4, В1 ОПК-8 34, 35, У1, У2, В1
25	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-3 У1, У2, У3, У4, В1 ОПК-8 34, 35, У1, У2, В1
26	Составьте схему алгоритма и программу на алгоритмическом языке вывода цифр заданного натурального числа в порядке слева направо (от старших разрядов к младшим).	ОПК-3 У1, У2, У3, У4, В1 ОПК-8 34, 35, У1, У2, В1
27	Составьте схему алгоритма и программу на алгоритмическом языке нахождения всех простых чисел, не превышающих заданное натуральное число.	ОПК-3 У1, У2, У3, У4, В1 ОПК-8 34, 35, 37, У1, У2, В1
28	Составьте схему алгоритма и программу на алго-	ОПК-3 У1, У2, У3, У4, В1

[Введите текст]

	ритмическом языке нахождения всех чисел, не превышающих заданное натуральное число и обладающих следующим свойством: произведение цифр искомого числа равно сумме цифр заданного.	ОПК-8 34, 35, У1, У2, В1
29	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-3 У1, У2, У3, У4, В1 ОПК-8 34, 35, У1, У2, В1
30	Составьте схему алгоритма и программу на алгоритмическом языке нахождения для двух заданных натуральных чисел наибольшего общего делителя как произведения общих простых множителей.	ОПК-3 У1, У2, У3, У4, В1 ОПК-8 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 информатику как науку и как вид человеческой деятельности. Приведите примеры использования информационных технологий в различных предметных областях.	ОПК-3 31, 32
2	Охарактеризуйте принципы фон Неймана как основу структуры персонального компьютера.	ОПК-3 31
3	Дайте определение позиционной системы счисления. Опишите алгоритмы перевода чисел из десятичной системы счисления в двоичную и обратно, из двоичной в восьмеричную и шестнадцатеричную и обратно. Приведите примеры перевода чисел из одной позиционной системы счисления в другую.	ОПК-3 31
4	Охарактеризуйте способы представления в компьютере целых чисел. Как способы представления отражаются на особенностях целого типа в языках программирования высокого уровня?	ОПК-3 31, 34
5	Охарактеризуйте способы представления в компьютере действительных чисел. Как способы представления отражаются на особенностях действительного типа в языках программирования высокого уровня?	ОПК-3 31, 34
6	Дайте определения понятию алгоритм. Охарактеризуйте свойства алгоритма. Приведите примеры, иллюстрирующие свойства алгоритма.	ОПК-3 34
7	Дайте определение понятию алгоритмическая система. Охарактеризуйте составные части алгоритмической системы. Проанализируйте алгоритмическую систему на примере императивной системы программирования.	ОПК-3 35
8	Охарактеризуйте концепцию типов данных в алгоритмических языках программирования на примере языков Бейсик и Паскаль. Проанализируйте особенности целочисленной и действительной компь-	ОПК-8 32, 34

[Введите текст]

	ютерной арифметики. Приведите примеры, иллюстрирующие эти особенности.	
--	--	--

[Введите текст]

9	Охарактеризуйте составные части алгоритмического языка программирования. Приведите метафоры, определяющие понятие литеры алфавита языка программирования.	ОПК-3 У5 ОПК-8 31
10	Опишите метаязык описания языков программирования. Приведите примеры метаформул.	ОПК-3 У5
11	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-3 38, У3, У4 ОПК-8 35, У1
12	Опишите синтаксис и семантику условных операторов в языках Бейсик и Паскаль. Приведите примеры организации ветвлений посредством условных операторов.	ОПК-3 У2 ОПК-8 35, У1
13	Опишите синтаксис и семантику операторов выбора в языках Бейсик и Паскаль. Приведите примеры организации ветвлений посредством операторов выбора.	ОПК-3 У2 ОПК-8 35, У1
14	Опишите синтаксис и семантику циклов с постусловием в языках Бейсик и Паскаль. Приведите соответствующие примеры организации циклов.	ОПК-3 У2 ОПК-8 35, У1
15	Опишите синтаксис и семантику циклов с предусловием в языках Бейсик и Паскаль. Приведите соответствующие примеры организации циклов.	ОПК-3 У2 ОПК-8 35, У1
16	Опишите синтаксис и семантику цикла с параметром в языке Бейсик. Приведите соответствующие пример организации цикла.	ОПК-3 У2 ОПК-8 35, У1
17	Опишите синтаксис и семантику цикла с параметром в языке Паскаль. Приведите соответствующие пример организации цикла.	ОПК-3 У2 ОПК-8 35, У1
18	Опишите синтаксис и семантику оператора резервирования памяти в СП Qbasic. Приведите пример схемы алгоритма и программы на языке Бейсик поиска элемента массива.	ОПК-3 У5, В1 ОПК-8 36, У1, У2, У3, В1
19	Опишите синтаксис типа массив в языке Паскаль. Приведите пример схемы алгоритма и программы на языке Паскаль сортировки массива.	ОПК-3 У5, В1 ОПК-8 36, У1, У2, У3, В1
20	Опишите синтаксис типизированной константы в языке Паскаль. Приведите примеры определения типизированных констант. Осуществите сравнение типизированной константы и переменной.	ОПК-8 34, У2
21	Охарактеризуйте строковый тип данных в языке Бейсик. Какие стандартные средства обработки строк языка Бейсик Вы знаете? Приведите примеры обработки строк средствами языка Бейсик.	ОПК-3 У5, В1 ОПК-8 У2, У4

22	Охарактеризуйте строковый тип данных в языке Паскаль. Какие стандартные средства обработки строк языка Паскаль Вы знаете? Приведите примеры обработки строк средствами языка Паскаль.	ОПК-3 У5, В1 ОПК-8 У2, У4
23	Охарактеризуйте принцип модульности и свойства алгоритмического модуля. Осуществите сравнение принцип модульности с принципом «черного ящика».	ОПК-3 39, 310
24	Опишите правила оформления подалгоритма, приведите пример. Охарактеризуйте вызов алгоритмического модуля.	ОПК-3 310, 311, У2, У5
25	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-3 У5, В1 ОПК-8 38, 39, У5
26	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-3 У5, В1 ОПК-8 38, 39, У5
27	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-3 У5, В1 ОПК-8 38, 39, У5
28	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-3 У5, В1 ОПК-8 38, 39, У5
29	Охарактеризуйте процедурный тип в языке Паскаль. Приведите пример использования процедурного типа для передачи параметра процедуры или параметра функции.	ОПК-3 У5, В1 ОПК-8 38, 39, У5
30	Дайте определение понятия рекурсия. Сформулируйте условия применимости приема рекурсия в алгоритмизации и программировании. Опишите на конкретном примере семантику рекурсивного вызова.	ОПК-3 311, В1 ОПК-8 38, 39, У5
31	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-3 У5, В1 ОПК-8 У6, В1
32	Опишите основы построения графических изображений средствами алгоритмических языков программирования на примере СП Qbasic и Turbo-Pascal. Опишите простейший алгоритм моделирования движения графического изображения.	ОПК-3 В1 ОПК-8 У7, В1

33	Опишите синтаксис типа запись с фиксированными полями в языке Паскаль. Приведите описания данных, имеющих тип запись.	ОПК-3 У5 ОПК-8 310, У9
34	Дайте определение понятия файл. Какие классификации файлов данных Вы знаете? Опишите синтаксис типизированного файла в языке Паскаль. Приведите пример описания типизированного файла.	ОПК-3 У5 ОПК-8 310, 314
35	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-3 У5 ОПК-8 310, 314, 315, У9
36	Охарактеризуйте понятия статического и динамического объекта программы на языке Паскаль. Опишите синтаксис типа указатель в языке Паскаль, синтаксис и семантику основных средств управления использованием динамической памяти в языке Паскаль. Приведите примеры захвата и освобождения памяти под динамические переменные.	ОПК-3 У5 ОПК-8 312, 313
37	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-3 В1 ОПК-8 У1, У2, У5, У8, У9, В1
38	Приведите пример схемы и программы алгоритмического модуля на языке Паскаль для вставки элемента в односвязный список.	ОПК-3 В1 ОПК-8 У1, У2, У5, У8, У9, В1
39	Приведите пример схемы и программы алгоритмического модуля на языке Паскаль для удаления элемента односвязного списка.	ОПК-3 В1 ОПК-8 У1, У2, У5, У8, У9, В1
40	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-3 В1 ОПК-8 У1, У2, У8, У9
41	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-3 У1, У2, У3, У4, В1 ОПК-8 36, 38, У1, У2, У3, У5, В1

42	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-3 У1, У2, У3, У4, В1 ОПК-8 36, 37, 38, У1, У2, У3, У5, В1
43	Составьте схему алгоритма и программу нахождения максимума матрицы с использованием подалгоритма поиска минимального элемента одномерного массива (подсказка: $\max\{f_k\} = -\min\{-f_k\}$).	ОПК-3 У1, У2, У3, У4, В1 ОПК-8 36, 38, У1, У2, У3, У5, В1
44	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-3 У1, У2, У3, У4, В1 ОПК-8 36, 37, 38, У1, У2, У3, У5, В1
45	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-3 У1, У2, У3, У4, В1 ОПК-8 36, 37, 38, У1, У2, У3, У5, В1
46	Составьте схему алгоритма и программу формирования для заданной матрицы вектора, каждая координата которого является средним арифметическим элементов, расположенных на диагонали, параллельной главной. Использовать подалгоритм вычисления среднего арифметического элементов одномерного массива.	ОПК-3 У1, У2, У3, У4, В1 ОПК-8 36, 38, У1, У2, У3, У5, В1
47	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-3 У1, У2, У3, У4, В1 ОПК-8 36, 37, 38, У1, У2, У3, У5, В1
48	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-3 У1, У2, У3, У4, В1 ОПК-8 36, 38, У1, У2, У3, У5, В1

49	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-3 У1, У2, У3, У4, В1 ОПК-8 36, 37, 38, У1, У2, У3, У5, В1
50	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 M . Использовать подалгоритм поиска максимального элемента одномерного массива.	ОПК-3 У1, У2, У3, У4, В1 ОПК-8 36, 38, У1, У2, У3, У5, В1
51	Составьте схему алгоритма и программу нахождения самой длинной из дуг, образованных графиками функций $f(x)=x^2+2\text{Ln}(1+x^2)$, $f(x)=x-x\text{Ln}x^2$ 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-3 У1, У2, У3, У4, В1 ОПК-8 36, 38, У1, У2, У3, У5, В1
52	Составьте схему алгоритма и программу нахождения нормы заданной матрицы A размера 3×5 и заданной матрицы B размера 4×4 . 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-3 У1, У2, У3, У4, В1 ОПК-8 36, 38, У1, У2, У3, У5, В1
53	Составьте схему алгоритма и программу вычислить A^3 и B^4 , где A – заданная матрица размера 4×4 , B – заданная матрица размера 3×3 . Использовать модуль вычисления произведения двух квадратных матриц.	ОПК-3 У1, У2, У3, У4, В1 ОПК-8 36, 38, У1, У2, У3, У5, В1

**КОМПЛЕКТ ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ПРОМЕЖУТОЧНОЙ
АТТЕСТАЦИИ (ЭКЗАМЕН 3 СЕМЕСТР)**

№	Содержание оценочного средства	Индекс оцениваемой компетенции и ее элементов
1	Охарактеризуйте императивную парадигму программирования. Приведите примеры соответствующих языков программирования.	ОПК-3 312, 313
2	Охарактеризуйте событийно-управляемую парадигму программирования. Приведите примеры соответствующих языков программирования.	ОПК-3 312, 313
3	Охарактеризуйте декларативную парадигму программирования. Приведите примеры соответствующих языков программирования.	ОПК-3 312, 313

[Введите текст]

4	Дайте общую характеристику языков программирования семейства Си. Каким парадигмам программирования могут соответствовать программы на языке Си?	ОПК-3 312, 313
5	Охарактеризуйте структуру простейшей программы и особенности стандартной библиотеки языка Си. Какой транслятор используется в СП семейства языков Си?	ОПК-8 33
6	Опишите простейший синтаксис функции main() программы на языке Си. Приведите пример простейшей программы на языке Си.	ОПК-3 У5 ОПК-8 33
7	Опишите основные стандартные типы языка Си и их совместимость, синтаксис и семантику резервирования памяти под автоматические переменные. Приведите примеры резервирования памяти.	ОПК-3 У5 ОПК-8 34
8	Охарактеризуйте основные операции языка Си и их совместное использование, особенности операций деления, присваиваний, инкремента и декремента. Приведите примеры выражений языка Си.	ОПК-8 34
9	Опишите операции отношения и логические операции языка Си. Приведите примеры логических выражений на языке Си.	ОПК-8 34
10	Опишите синтаксис и семантику условного оператора в языке Си. Приведите примеры организации ветвлений средствами условного оператора на языке Си.	ОПК-3 У5 ОПК-8 35
11	Опишите синтаксис и семантику условной операции в языке Си. Приведите примеры организации ветвлений средствами условной операции на языке Си.	ОПК-3 У5 ОПК-8 34
12	Опишите синтаксис и семантику селектора в языке Си. Приведите примеры организации структуры выбор средствами селектора на языке Си.	ОПК-3 У5 ОПК-8 35
13	Опишите синтаксис и семантику операторов циклов с предусловием и постусловием в языке Си. Приведите соответствующие примеры организации циклов	ОПК-3 У5 ОПК-8 35
14	Опишите синтаксис и семантику оператора цикла for в языке Си. Приведите соответствующие примеры организации циклов.	ОПК-3 У5 ОПК-8 35
15	Опишите синтаксис и семантику потокового ввода и вывода в языке Си. Какие манипуляторы Вы знаете? Приведите примеры потокового ввода и вывода на языке Си.	ОПК-3 У5 ОПК-8 34
16	Опишите синтаксис и семантику форматированного ввода в языке Си. Приведите примеры форматированного ввода на языке Си.	ОПК-3 У5 ОПК-8 34
17	Опишите синтаксис и семантику форматированного вывода в языке Си. Приведите примеры форматированного вывода на языке Си.	ОПК-3 У5 ОПК-8 34

[Введите текст]

18	Охарактеризуйте тип данных массив в языке Си. Опишите синтаксис и семантику оператора резервирования памяти в языке Си. Приведите примеры резервирования памяти под массив на языке Си.	ОПК-3 У5 ОПК-8 36
19	Охарактеризуйте особенности хранения значений элементов массивов в памяти. Опишите синтаксис и семантику инициализации массивов в языке Си. Приведите примеры инициализации массивов на языке Си.	ОПК-3 У5 ОПК-8 36
20	Опишите альтернативный доступ к элементам массива с помощью операции значения по указанному адресу в языке Си. Приведите примеры альтернативный доступ к элементам массива в языке Си.	ОПК-3 У5 ОПК-8 36
21	Охарактеризуйте способ представления строк в языке Си в виде массива символов. Какие стандартные средства обработки строк языка Си Вы знаете? Приведите примеры обработки строк стандартными средствами языка Си.	ОПК-3 У5 ОПК-8 У4
22	Охарактеризуйте особенности обработки строк как массивов символов средствами языка Си: копирование части строки, удаление части строки. Приведите соответствующие примеры обработки строк средствами языка Си.	ОПК-3 В1 ОПК-8 У4
23	Охарактеризуйте особенности обработки строк как массивов символов средствами языка Си: вставка строки. Приведите соответствующий пример обработки строк средствами языка Си.	ОПК-3 В1 ОПК-8 У4
24	Охарактеризуйте особенности реализация принципа модульности средствами языка Си. Опишите синтаксис функции пользователя в языке Си, способы передачи параметра простого типа. Приведите простейшие примеры описания и вызова функций пользователя с разными способами передачи параметров на языке Си.	ОПК-3 У5 ОПК-8 38, 39
25	Опишите, как осуществить средствами языка Си передачу параметра массива и параметра функции. Опишите простейший способ организации библиотеки пользователя средствами языка Си. Приведите примеры описания и вызова функций пользователя с параметром массивом и параметром функцией на языке Си.	ОПК-3 У2, У5, В1 ОПК-8 38, 39, У5, У6
26	Опишите перегрузку функции в языке Си. Приведите пример описания и вызова функций пользователя, иллюстрирующий перегрузку функции на языке Си.	ОПК-3 У2, В1 ОПК-8 38, 39, У5, У6
27	Опишите синтаксис функции с аргументами по умолчанию в языке Си. Приведите пример описания и вызова функции пользователя с аргументами по умолчанию на языке Си.	ОПК-3 У2, У5, В1 ОПК-8 38, 39, У5, У6
28	Опишите синтаксис типа структура в языке Си. Приведите описания данных, имеющих тип струк-	ОПК-3 У5 ОПК-8 311, У9

[Введите текст]

	тура.	
29	Охарактеризуйте понятия статического и динамического объекта программы на языке Си. Опишите синтаксис типа указатель в языке Си. Приведите примеры резервирования памяти под переменную типа указатель на языке Си.	ОПК-3 У5 ОПК-8 311, 312, 313
30	Опишите синтаксис и семантику средств управления использованием динамической памятью под переменные простого типа в языке Си. Приведите примеры захвата и освобождения памяти под динамические переменные простого типа на языке Си.	ОПК-3 У5 ОПК-8 313
31	Опишите синтаксис и семантику средств управления использованием динамической памятью под массивы в языке Си. Приведите примеры захвата и освобождения памяти под динамические массивы на языке Си.	ОПК-3 У5 ОПК-8 313
32	Опишите использование типа указатель для организации структуры список средствами языка Си. Приведите пример схемы и программы алгоритмического модуля на языке Си для организации односвязного списка.	ОПК-3 В1 ОПК-8 У1, У2, У5, У8, У9, В1
33	Охарактеризуйте файлы данных в языке Си. Опишите синтаксис указателя на файл в языке Си. Приведите пример соответствующего описания.	ОПК-3 У5 ОПК-8 311, 314
34	Опишите синтаксис и семантику функций, последовательность их вызова для организации вывода в текстовый файл и ввода из текстового файла в СП Turbo-C++. Приведите примеры организации вывода в текстовый файл и ввода из текстового файла средствами СП Turbo-C++.	ОПК-3 У5 ОПК-8 314, 315, У9
35	Составьте фрагменты схемы алгоритма и программы с соответствующими разделами описаний на языке Си для решения следующей задачи. Удалить один элемент из существующего списка данных с информационной частью типа массив и записать его информационную часть во вновь созданный файл.	ОПК-3 У1, У2, У3, В1 ОПК-8 311, 313, 315, У1, У2, У8, У9, В1
36	Составьте фрагменты схемы алгоритма и программы с соответствующими разделами описаний на языке Си для решения следующей задачи. Проверить гипотезу о том, что информационная часть целого типа каждого элемента существующего списка данных меньше соответствующего элемента существующего файла данных.	ОПК-3 У1, У2, У3, В1 ОПК-8 37, 311, 313, 315, У1, У2, У8, У9, В1
37	Составьте фрагменты схемы алгоритма и программы с соответствующими разделами описаний на языке Си для решения следующей задачи. Дан файл, который содержит информацию о значении элементов матрицы построчно. Вставить в существующий список данных с информационной частью типа массив вторую строку матрицы из фай-	ОПК-3 У1, У2, У3, В1 ОПК-8 311, 313, 315, У1, У2, У8, У9, В1

[Введите текст]

	ла.	
--	-----	--

[Введите текст]

38	Составьте фрагменты схемы алгоритма и программы с соответствующими разделами описаний на языке Си для решения следующей задачи. Удалить из существующего стека данных с информационной частью целого типа три элемента и добавить в этот же стек последний элемент существующего файла данных.	ОПК-3 У1, У2, У3, В1 ОПК-8 311, 313, 315, У1, У2, У8, У9, В1
39	Составьте фрагменты схемы алгоритма и программы с соответствующими разделами описаний на языке Си для решения следующей задачи. Добавить в существующую очередь введенное с клавиатуры данное типа структура и удалить из этой очереди элемент с выводом значений членов структуры информационной части на монитор.	ОПК-3 У1, У2, У3, В1 ОПК-8 311, 313, 315, У1, У2, У8, У9, В1
40	Составьте схему алгоритма и программу на языке Си для решения следующей задачи. Дан файл, который содержит информацию о значении элементов матрицы построчно. Используя структуру стек изменить порядок следования компонентов файла так, чтобы порядок строк изменился на противоположный без изменения порядка следования элементов внутри строки.	ОПК-3 У1, У2, У3, В1 ОПК-8 311, 313, 315, У1, У2, У8, У9, В1
41	Составьте фрагменты схемы алгоритма и программы на языке Си для решения следующей задачи. Используя односвязный список отсортировать элементы существующего файла по невозрастанию.	ОПК-3 У1, У2, У3, В1 ОПК-8 311, 313, 315, У1, У2, У8, У9, В1
42	Составьте фрагменты схемы алгоритма и программы с соответствующими разделами описаний на языке Си для решения следующей задачи. Дан файл, который содержит информацию о значении элементов матрицы построчно. Добавить в существующую очередь с информационной частью типа массив шестую строку матрицы из файла.	ОПК-3 У1, У2, У3, В1 ОПК-8 311, 313, 315, У1, У2, У8, У9, В1
43	Составьте фрагменты схемы алгоритма и программы с соответствующими разделами описаний на языке Си для решения следующей задачи. Ввести с клавиатуры и добавить в существующий стек с информационной частью типа массив три элемента.	ОПК-3 У1, У2, У3, В1 ОПК-8 311, 313, 315, У1, У2, У8, У9, В1
44	Составьте фрагменты схемы алгоритма и программы с соответствующими разделами описаний на языке Си для решения следующей задачи. Заменить в существующем кольце из элементов с информационной частью типа структура с пятью членами один элемент с указанными значениями членов структуры.	ОПК-3 У1, У2, У3, В1 ОПК-8 311, 313, 315, У1, У2, У8, У9, В1
45	Составьте схему алгоритма и программу на языке Си для решения следующей задачи. Построить структуру данных кольцо из 9-и элементов с информационной частью типа массив.	ОПК-3 У1, У2, У3, В1 ОПК-8 311, 313, 315, У1, У2, У8, У9, В1

46	Составьте фрагменты схемы алгоритма и программы с соответствующими разделами описаний на языке Си для решения следующей задачи. Дан файл, который содержит информацию о значении элементов матрицы построчно. Проверить гипотезу о том, что каждая строка матрицы из файла данных равна информационной части соответствующего элемента существующего списка данных.	ОПК-3 У1, У2, У3, В1 ОПК-8 37, 311, 313, 315, У1, У2, У8, У9, В1
47	Составьте фрагменты схемы алгоритма и программы с соответствующими разделами описаний на языке Си для решения следующей задачи. Удалить из существующей очереди элемент с информационной частью типа структура и записать члены структуры во вновь созданный файл.	ОПК-3 У1, У2, У3, В1 ОПК-8 311, 313, 315, У1, У2, У8, У9, В1
48	Составьте фрагменты схемы алгоритма и программы с соответствующими разделами описаний на языке Си для решения следующей задачи. Дан файл, который содержит информацию о значениях элементов массива типа структура с пятью членами. Шестой элемент массива этого файла данных добавить в существующую очередь с информационной частью типа структура.	ОПК-3 У1, У2, У3, В1 ОПК-8 311, 313, 315, У1, У2, У8, У9, В1
49	Составьте фрагменты схемы алгоритма и программы с соответствующими разделами описаний на языке Си для решения следующей задачи. Отсортировать информационную часть целого типа существующего списка таким образом, чтобы нечетные значения предшествовали четным без изменения порядка следования элементов одной четности.	ОПК-3 У1, У2, У3, В1 ОПК-8 311, 313, 315, У1, У2, У8, У9, В1
50	Составьте фрагменты схемы алгоритма и программы с соответствующими разделами описаний на языке Си для решения следующей задачи. Ввести с клавиатуры и добавить в существующий стек с информационной частью типа структура из трех членов пять элементов.	ОПК-3 У1, У2, У3, В1 ОПК-8 311, 313, 315, У1, У2, У8, У9, В1
51	Составьте фрагменты схемы алгоритма и программы с соответствующими разделами описаний на языке Си для решения следующей задачи. Проверить гипотезу о том, что информационная часть целого типа каждого элемента существующего списка данных является делителем соответствующего элемента существующего файла данных.	ОПК-3 У1, У2, У3, В1 ОПК-8 37, 311, 313, 315, У1, У2, У8, У9, В1
52	Составьте фрагменты схемы алгоритма и программы с соответствующими разделами описаний на языке Си для решения следующей задачи. Удалить из существующей очереди с информационной частью типа массив четыре элемента с выводом значений элементов удаленных массивов на дисплей.	ОПК-3 У1, У2, У3, В1 ОПК-8 311, 313, 315, У1, У2, У8, У9, В1

53	Составьте фрагменты схемы алгоритма и программы с соответствующими разделами описаний на языке Си для решения следующей задачи. Дан файл, который содержит информацию о значениях элементов массива типа структура. Проверить гипотезу о том, что информационная часть типа структура каждого элемента существующего списка данных не равна соответствующему элементу массива существующего файла данных.	ОПК-3 У1, У2, У3, В1 ОПК-8 37, 311, 313, 315, У1, У2, У8, У9, 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 «Информатика и программирование» (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.