

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
«ВЫЧИСЛИТЕЛЬНЫЕ ПРОЦЕССЫ И СТРУКТУРЫ»**

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 «Вычислительные процессы и структуры» является формирование у обучающихся общепрофессиональных и профессиональных компетенций в процессе изучения теоретических основ создания трансляторов, верификации и тестирования прикладных программ для последующего применения в учебной и практической деятельности.

Задачи дисциплины можно разделить на две группы:

- при изучении теоретических аспектов дисциплины основное внимание должно уделяться математическим моделям формальных языков, их синтаксиса и семантики, включая аппарат формальных грамматик, метаязыков, конечных автоматов и их приложений, а также моделям вычислений и программ, применяемых при генерации объектного кода в трансляторах, в процессе верификации и тестирования программ;

- практическая направленность дисциплины должна выражаться в изучении алгоритмов и информационных структур, обеспечивающих разработку современных средств синтаксического анализа различных описаний, применяемых в современных информационных системах, включая тексты исходных программ и их отдельных фрагментов, а также создание трансляторов (компиляторов, интерпретаторов или ассемблеров) для современных и перспективных моделей ЭВМ и микропроцессоров.

2. Место дисциплины в структуре ООП бакалавриата

2.1. Дисциплина Б1.Б.20.«Вычислительные процессы и структуры» относится к базовой части Блока 1.

2.2. Для изучения дисциплины «Вычислительные процессы и структуры» необходимы знания, умения, навыки, формируемые предшествующими дисциплинами:

- «Информатика и программирование»;
- «Дискретная математика»;
- «Структуры и алгоритмы компьютерной обработки данных»;
- «Архитектура вычислительных систем и компьютерных сетей»;
- «Операционные системы и оболочки».

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

- производственная практика;
- государственная итоговая аттестация.

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 общепрофессиональных (ОПК) и профессиональных (ПК) компетенций:

№	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-2	Способность применять в профессиональной деятельности знания математических основ информатики	математические модели вычислительных процессов, основы формальных языков и грамматик, теорию трансляции программ	использовать математические модели языков программирования для разработки алгоритмов лексического и синтаксического анализа	навыками описания программных конструкций и их элементов средствами математического аппарата формальных грамматик и языков
2	ОПК-4	Способность применять в профессиональной деятельности основные методы и средства автоматизации проектирования, производства, испытаний и оценки качества программного обеспечения	- математические модели, используемые при разработке, отладке, верификации и тестировании программ; - базовые методы и подходы к построению алгоритмов (итерация и рекурсия, модульный, структурный и объектно-ориентированный методы алгоритмизации)	- разрабатывать модели прикладных программ, используемые при структурном тестировании; - оценивать эффективность алгоритмов по затратам времени и машинной памяти	- методами нисходящего и восходящего проектирования алгоритмов; - навыками программирования и трансляции прикладных программ с использованием современных систем программирования
3	ПК-5	Готовность к использованию современных системных программных средств: операционных систем, операционных и сетевых оболочек, сервисных программ	- основные возможности и особенности современных операционных систем и систем программирования; - функции и команды операционных систем, оболочек и сервисных программ, основные компоненты систем программирования и их назначение	- использовать системное программное обеспечение персонального компьютера для решения прикладных задач; - применять современные технологии программирования, отладки и тестирования программных модулей лексического и синтаксического анализа	- средствами операционных систем и оболочек для организации выполнения прикладных программ на ЭВМ; - навыками настройки систем программирования и операционных систем при разработке и отладке прикладных программ

2.5.Карта компетенций

КАРТА КОМПЕТЕНЦИЙ ДИСЦИПЛИНЫ

НАИМЕНОВАНИЕ ДИСЦИПЛИНЫ: ВЫЧИСЛИТЕЛЬНЫЕ ПРОЦЕССЫ И СТРУКТУРЫ

Цель дисциплины Целью освоения дисциплины является формирование у обучающихся общепрофессиональных и профессиональных компетенций в процессе изучения теоретических основ создания трансляторов, верификации и тестирования прикладных программ 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				
1	2	3	4	5	6
ОПК-2	Способность применять в профессиональной деятельности знания математических основ информатики	<p>Знать: математические модели вычислительных процессов, основы формальных языков и грамматик, теорию трансляции программ</p> <p>Уметь: использовать математические модели языков программирования для разработки алгоритмов лексического и синтаксического анализа</p> <p>Владеть: навыками описания программных конструкций и их элементов средствами математического аппарата формальных грамматик и языков</p>	Проведение лекционных и лабораторных занятий, применение новых образовательных технологий, организация самостоятельной работы студентов	Лабораторные работы, экзамен	<p>Пороговый: способен понимать формальное описание синтаксиса языков программирования с помощью аппарата формальных грамматик</p> <p>Повышенный: способен использовать метаязыки для самостоятельного описания синтаксиса языковых конструкций</p>
ОПК-4	Способность применять в профессиональной деятельности основные методы и средства автоматизации проектирования, производства, испытаний и оценки качества программного обеспечения	<p>Знать: - математические модели, используемые при разработке, отладке, верификации и тестировании программ; - базовые методы и подходы к построению алгоритмов (итерация и рекурсия, модульный, структурный и объектно-ориентированный методы алгоритмизации)</p> <p>Уметь: - разрабатывать модели прикладных программ, используемые при структурном тестировании; - оценивать эффективность алгоритмов по затратам времени и машинной памяти</p> <p>Владеть: - методами нисходящего и восходящего проектирования алгоритмов; - навыками программирования и трансляции прикладных программ с использованием современных систем программирования</p>	Проведение лекционных и лабораторных занятий, применение новых образовательных технологий, организация самостоятельной работы студентов	Лабораторные работы, экзамен	<p>Пороговый: - способен использовать структурный и объектно-ориентированный подходы к алгоритмизации; - может определить класс сложности алгоритма (полиномиальный, экспоненциальный)</p> <p>Повышенный: - способен использовать модели алгоритмов и программ, применяемые для верификации и тестирования программ; - может оценить эффективность алгоритма или программы по затратам машинного времени или памяти</p>

Профессиональные компетенции:

КОМПЕТЕНЦИИ		Перечень компонентов	Технологии формирования	Форма оценочного средства	Уровни освоения компетенций
ИНДЕКС	ФОРМУЛИРОВКА				
1	2	3	4	5	6
ПК-5	Готовность к использованию современных системных программных средств: операционных систем, операционных и сетевых оболочек, сервисных программ	<p>Знать:</p> <ul style="list-style-type: none"> - основные возможности и особенности современных операционных систем и систем программирования; - функции и команды операционных систем, оболочек и сервисных программ, основные компоненты систем программирования и их назначение <p>Уметь:</p> <ul style="list-style-type: none"> - использовать системное программное обеспечение персонального компьютера для решения прикладных задач; - применять современные технологии программирования, отладки и тестирования программных модулей лексического и синтаксического анализа <p>Владеть:</p> <ul style="list-style-type: none"> - средствами операционных систем и оболочек для организации выполнения прикладных программ на ЭВМ; - навыками настройки систем программирования и операционных систем при разработке и отладке прикладных программ 	Проведение лекционных и лабораторных занятий, применение новых образовательных технологий, организация самостоятельной работы студентов	Лабораторные работы, экзамен	<p>Пороговый:</p> <ul style="list-style-type: none"> - способен выполнить разработку программных модулей для решения задач дисциплины с использованием стандартных настроек системных программных средств; - способен использовать стандартные средства операционных систем и систем программирования при решении прикладных задач <p>Повышенный:</p> <ul style="list-style-type: none"> - способен осуществлять выполнять настройку системных программных средств, включая отладчики программ, для повышения эффективности разрабатываемых программных модулей; - способен выполнять настройку операционных систем и систем программирования для повышения эффективности компьютерной системы

ОСНОВНАЯ ЧАСТЬ

1. Объем дисциплины и виды учебной работы

Вид учебной работы		Всего часов	Семестр
			7
Контактная работа обучающихся с преподавателем (по видам учебных занятий) (всего)		72	72
В том числе:			
Лекции (Л)		36	36
Лабораторные работы (ЛР)		36	36
Самостоятельная работа студента (всего)		72	72
В том числе			
Изучение литературы и других источников		27	27
Подготовка к выполнению лабораторных работ		27	27
Подготовка к защите лабораторных работ		18	18
Контроль		36	36
Вид промежуточной аттестации	Экзамен (Э)		+
ИТОГО: Общая трудоемкость		часов	180
		Зач.ед.	5

Л – лекции, ЛР – лабораторные работы; СР – самостоятельная работа студента.

2. Содержание дисциплины

2.1. Содержание разделов дисциплины

№ семестра	№ раздела	Наименование раздела дисциплины	Содержание раздела в дидактических единицах
1	2	3	4
7	1	Организация вычислительных процессов и структур	Понятия вычислительного процесса и ресурса. Разделяемые и неразделяемые ресурсы. Состояния вычислительного процесса. Роль операционной системы в организации исполнения вычислительных процессов. Классификация вычислительных процессов по динамическому признаку (последовательные, параллельные, комбинированные), по месту развития (внутренние, внешние), по связности и др. Вычислительные процессы реального времени. Системные программные средства и вычислительные процессы. Основные компоненты системы программирования. Транслятор, редактор связей, загрузчик, связывающий загрузчик, отладчик, текстовый редактор. Основные виды трансляторов. Компилятор, интерпретатор, ассемблер, макроассемблер, кросс-ассемблер. Эффективность компиляторов и интерпретаторов.

1	2	3	4
7	2	Математические модели формальных языков	<p>Понятие формального языка. Алфавит, синтаксис и семантика языка. Средства описания синтаксиса формальных языков. Формальные языки и грамматики. Определение формальной грамматики. Математическое определение формального языка. Сентенциальная форма грамматики. Предложение (сентенция) языка, порождаемого грамматикой. Классификация формальных языков и порождающих их грамматик. Понятие метаязыка. Стандартная форма Бэкуса-Наура (БНФ) и ее расширения. Синтаксические диаграммы и правила их построения. Регулярные грамматики и конечные автоматы. Определение конечного автомата. Функционирование конечного автомата и способы задания функции переходов (множество команд, диаграмма состояний, матрица переходов). Связь регулярной грамматики и конечного автомата. Построение распознающего автомата для регулярных языков. Контекстно-свободные грамматики и проблема разбора языков. Левосторонний и правосторонний вывод. Дерево грамматического разбора. Восходящие и нисходящие методы разбора. Понятия основы сентенциальной формы и операции свертки. Проблемы однозначности и эквивалентности грамматик. Достаточные условия однозначности грамматик. Необходимые и достаточные условия детерминированного распознавания языков, описываемых контекстно-свободными грамматиками. LL(k)- и LL(1)-грамматики.</p>
7	3	Трансляторы и методы анализа программ	<p>Фазы трансляции программ. Принципы функционирования компиляторов, интерпретаторов и ассемблеров. Варианты построения компиляторов. Лексический анализ формальных языков. Функции лексического анализатора. Идентификация лексем формальных языков. Классы распознаваемых лексем. Дескрипторы лексем конечных и бесконечных классов. Принципы работы лексического анализатора. Обнаружение ошибок в лексемах. Проектирование лексических анализаторов. Синтаксический анализ формальных языков. Контекстно-свободные грамматики и их свойства. Методы грамматического разбора формальных языков. Нисходящий грамматический разбор с возвратами. Метод рекурсивного спуска. Нисходящий грамматический разбор на основе синтаксических диаграмм. Восходящие методы грамматического разбора. Грамматики и отношения предшествования. Грамматики простого предшествования. Фраза, простая фраза и основа сентенциальной формы. Рекурсия в грамматиках простого предшествования. Первичная фраза. Грамматики операторного предшествования. Матрица предшествования. Вычисление отношений предшествования для грамматик операторного предшествования. Грамматический разбор методом оперативного предшествования. Обнаружение синтаксических ошибок. Модели представления программ в трансляторах. Семантическое дерево, польская запись, тетрадная форма представления программ. Семантический анализ. Функции семантического анализатора. Построение семантического анализатора. Способы формирования польской записи (обход семантического дерева, алгоритм Дейкстры). Генерация тетрадного представления программы. Вычисление выражений в польской записи.</p>

1	2	3	4
7	4	Оптимизация программ и генерация кода	Машинно-независимая оптимизация линейных участков программ (константные вычисления, исключение общих подвыражений). Машинно-независимая оптимизация циклических участков программ (вынесение инвариантных операций, замена сложных операций на простые, слияние и развертывание циклов). Тетрадная форма представления циклов и их оптимизирующие преобразования. Генерация и оптимизация объектного кода в компиляторах. Модели программы для генерации кода. Классы вычислительных архитектур и системы команд. Формы объектного кода. Алгоритмы генерации объектного кода. Синтаксически-управляемая генерация кода. Машинно-зависимая оптимизация объектного кода.
7	5	Информационные структуры транслятора	Состав и назначение информационных таблиц транслятора. Таблицы идентификаторов (символов), констант, вложенности блоков и др. Общая структура информационных таблиц. Состав информации, сохраняемой в таблице идентификаторов. Варианты организации таблицы идентификаторов. Неупорядоченные и упорядоченные информационные таблицы. Линейный поиск в неупорядоченной таблице. Бинарный поиск в упорядоченной таблице. Реализация алгоритмов бинарного поиска. Эффективность поиска в упорядоченных и неупорядоченных таблицах. Информационные таблицы с хеш-адресацией. Алгоритмы вычисления хеш-функции. Коллизии в хеш-таблицах. Методы разрешения коллизий. Повторное хеширование (рехеширование), метод цепочек. Варианты рехеширования. Линейное, случайное, квадратичное рехеширование, рехеширование сложением. Реализация метода цепочек. Поиск и добавление элементов в хеш-таблицу по методу цепочек. Эффективность методов рехеширования. Особенности организации таблицы идентификаторов для блочно-структурированных языков программирования.
7	6	Распределение памяти при трансляции	Задачи транслятора по распределению памяти. Распределение регистровой памяти. Алгоритмы распределения регистров в последовательных и параллельных вычислениях. Область команд и область данных в оперативной памяти. Особенности организации статической, автоматической и динамической памяти данных. Размещение структурированных данных в оперативной памяти. Вычисление адресов элементов массивов. Особенности хранения динамических массивов. Информационный вектор массива. Передача данных между блоками программы. Область инициализации. Дисплей. Выделение памяти при выполнении программы. Организация кучи. Распределение памяти для динамических данных. Управление кучей. Метод помеченных границ. Сбор мусора.
7	7	Моделирование, верификация и тестирование программ	Анализ структуры программы. Зависимости по данным и управлению. Граф зависимости по данным. Управляющий граф программы. Поточковый граф программы. Выделение линейных и циклических участков программ. Модели программ и вычислительных процессов. Схемы программ. Методы формальной спецификации и верификации программ. Модели вычислительных процессов. Взаимодействие процессов. Асинхронные процессы. Сети Петри: принципы построения, алгоритмы поведения, способы реализации. Применение сетей Петри для моделирования взаимодействующих вычислительных процессов. Структурное тестирование программ (метод «белого ящика»). Цикломатическая сложность структурного тестирования. Функциональное тестирование (метод «черного ящика»).

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	
7	1	Организация вычислительных процессов и структур	2	2	4	8	1-2 неделя Лабораторная работа № 1 3-4 неделя Лабораторная работа № 2
7	2	Математические модели формальных языков	8	6	14	28	
7	3	Трансляторы и методы анализа программ	12	12	24	48	5-6 неделя Лабораторная работа № 3 7-8 неделя Лабораторная работа № 4 9-10 неделя Лабораторная работа № 5
7	4	Оптимизация программ и генерация кода	6	8	14	28	11-12 неделя Лабораторная работа № 6 13-14 неделя Лабораторная работа № 7
7	5	Информационные структуры транслятора	4	4	8	16	15-16 неделя Лабораторная работа № 8
7	6	Распределение памяти при трансляции	2	4	6	12	17-18 неделя Лабораторная работа № 9
7	7	Моделирование, верификация и тестирование программ	2	-	2	4	-
7		Контроль				36	экзамен
		ИТОГО	36	36	72	180	

2.3. Лабораторный практикум

№ семестра	№ раздела	Наименование раздела дисциплины	Наименование лабораторных работ	Всего часов
1	2	3	4	5
7	2	Математические модели формальных языков	Лабораторная работа № 1 Польская запись, как форма представления программных конструкций в трансляторах	4
			Лабораторная работа № 2 Тетрадная форма представления программных конструкций в трансляторах	4
7	3	Трансляторы и методы анализа программ	Лабораторная работа № 3 Изучение и программная реализация методов лексического анализа, используемых в компиляторах	4
			Лабораторная работа № 4 Изучение и программная реализация нисходящих методов грамматического разбора на примере метода рекурсивного спуска	4

			Лабораторная работа № 5 Изучение и программная реализация восходящих методов грамматического разбора на примере метода операторного предшествования	4
7	4	Оптимизация программ и генерация кода	Лабораторная работа № 6 Исследование и программная реализация методов генерации и оптимизации объектного кода (для простой системы команд учебной ЭВМ)	4
			Лабораторная работа № 7 Исследование и программная реализация методов машинно-независимой оптимизации программ (для линейных и циклических участков)	4
7	5	Информационные структуры транслятора	Лабораторная работа № 8 Исследование методов поиска данных в информационных таблицах	4
7	6	Распределение памяти при трансляции	Лабораторная работа № 9 Исследование и программная реализация методов распределения регистровой памяти в компиляторах	4
ИТОГО				36

2.4. КУРСОВЫЕ РАБОТЫ не предусмотрены.

3. Самостоятельная работа студента

3.1. Виды СРС

№ семестра	№ раздела	Наименование раздела дисциплины	Виды СРС	Всего часов
1	2	3	4	5
7	1	Организация вычислительных процессов и структур	Изучение литературы и других источников	4
7	2	Математические модели формальных языков	Изучение литературы и других источников по разделу	4
			Подготовка к выполнению лабораторной работы № 1	2
			Подготовка к защите лабораторной работы № 1	2
			Подготовка к выполнению лабораторной работы № 2	4
			Подготовка к защите лабораторной работы №2	2
7	3	Трансляторы и методы анализа программ	Изучение литературы и других источников по разделу	4
			Подготовка к выполнению лабораторной работы № 3	4
			Подготовка к защите лабораторной работы № 3	2
			Подготовка к выполнению лабораторной работы № 4	4
			Подготовка к защите лабораторной работы № 4	3
			Подготовка к выполнению лабораторной работы № 5	4
			Подготовка к защите лабораторной работы № 5	3
7	4	Оптимизация программ и генерация кода	Изучение литературы и других источников по разделу	4
			Подготовка к выполнению лабораторной работы № 6	3
			Подготовка к защите лабораторной работы № 6	2
			Подготовка к выполнению лабораторной работы № 7	3
			Подготовка к защите лабораторной работы № 7	2
7	5	Информационные структуры транслятора	Изучение литературы и других источников по разделу	3
			Подготовка к выполнению лабораторной работы № 8	3

			Подготовка к защите лабораторной работы № 8	2
7	6	Распределение памяти при трансляции	Изучение литературы и других источников по разделу	2
			Подготовка к выполнению лабораторной работы № 9	2
			Подготовка к защите лабораторной работы № 9	2
7	7	Моделирование, верификация и тестирование программ	Изучение литературы и других источников по разделу	2
ИТОГО				72

3.2. График работы студента

Семестр № 7

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	ЛР	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+

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)

4.1. Рейтинговая система оценки знаний обучающихся по учебной дисциплине

Рейтинговая система не используется.

5. Учебно-методическое и информационное обеспечение дисциплины

5.1. Основная литература

№ п/п	Автор (ы), наименование, место издания и издательство, год	Используется при изучении разделов	Семестр	Количество экземпляров	
				В библиотеке	На кафедре
1	2	3	4	5	6
1	Гулько, А. В. Системное программное обеспечение [Электронный ресурс] : конспект лекций / А. В. Гулько. – Новосибирск : НГТУ, 2011. – 138 с. – Режим доступа: http://biblioclub.ru/index.php?page=book&id=228965 (дата обращения: 30.08.2019).	2-4	7	ЭБС	
2	Малякко, А. А. Системное программное обеспечение. Формальные языки и методы трансляции [Электронный ресурс] : учебное пособие : в 3 ч. Ч. 1. / А. А. Малякко. – Новосибирск : НГТУ, 2010. – 104 с. – Режим доступа: http://biblioclub.ru/index.php?page=book&id=228974&sr=1 (дата обращения: 30.08.2019).	2-4	7	ЭБС	
3	Молчанов, А. Ю. Системное программное обеспечение [Текст] : учебник / А. Ю. Молчанов. – Санкт-Петербург : Питер, 2003. – 396 с.	1-6,7	7	8	1

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	Гавриков М.М. Теоретические основы разработки и реализации языков программирования [Текст]: учебное пособие/ М.М. Гавриков, А.Н.Иванченко, Д.В. Гриченков: под ред. А.Н.Иванченко. -М.-КНОРУС, 2013	2-7	7	5	
2	Таненбаум, Э. Современные операционные системы [Текст] / Таненбаум Э. - СПб.: БХВ-Петербург, 2004. - 1040 с.	1	7	4	1

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

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

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

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

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

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

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

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

8. Труды преподавателей [Электронный ресурс] : коллекция // Электронная библиотека Научной библиотеки РГУ имени С.А. Есенина. – Доступ к полным текстам по паролю. – Режим доступа: <http://dspace.rsu.edu.ru/xmlui/handle/123456789/3> (дата обращения: 30.08.2019).

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

1. eLIBRARY.RU [Электронный ресурс] : научная электронная библиотека. – Режим доступа: <http://elibrary.ru/defaultx.asp>, свободный (дата обращения: 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://dnpetrov.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. 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, системы программирования Turbo-Pascal, PascalABC.NET, Turbo-C++, C++ Builder и др.

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.</p> <p>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.</p> <p>Уделить внимание следующим понятиям: вычислительный процесс, транслятор, компилятор, интерпретатор, формальная грамматика, грамматический разбор, фазы трансляции программ, генерация и оптимизация объектного кода, таблицы транслятора и их организация, хеш-адресация, распределение памяти при трансляции, модели программ и</p>

	вычислительных процессов, тестирование и верификация.
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 алгоритмов 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 разработанных программ в компьютер, осуществить отладку и тестирование программ, выполнить решение контрольных примеров, продемонстрировать и объяснить полученные результаты преподавателю.</p> <p>Оформить лабораторную работу в тетради или на отдельных листах с указанием фамилии студента, номера лабораторной работы и номера варианта. Оформленная работа также должна содержать полный текст задания, необходимые математические модели (описание грамматик, синтаксические диаграммы и др.), блок-схему алгоритма, тексты отлаженных на компьютере программ, результаты решения контрольных примеров и выводы с учетом решаемой задачи.</p> <p>Защитить оформленную лабораторную работу, продемонстрировав теоретические и практические знания, умения и навыки по соответствующей теме, возможные варианты решения задачи, математические модели языков или их отдельных конструкций, схемы алгоритмов и тексты программных моделей. Перечень примерных контрольных заданий и вопросов для защиты лабораторных работ приведен в разделе 10.2.</p>
Подготовка к экзамену	При подготовке к экзамену необходимо ориентироваться на конспекты лекций, рекомендуемую литературу, результаты выполнения лабораторных работ и другие учебные материалы.

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 подкатало-

ге 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 №Tr000043844 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

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	Организация вычислительных процессов и структур	ОПК-2 ОПК-4 ПК-5	Экзамен
2	Математические модели формальных языков		
3	Трансляторы и методы анализа программ		
4	Оптимизация программ и генерация кода		
5	Информационные структуры транслятора		
6	Распределение памяти при трансляции		
7	Моделирование, верификация и тестирование программ		

ТРЕБОВАНИЯ К РЕЗУЛЬТАТАМ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ

Индекс компетенции	Содержание компетенции	Элементы компетенции	Индекс элемента
1	2	3	4
ОПК-2	Способность применять в профессиональной деятельности знания математических основ информатики	знать	
		З1 - математические модели вычислительных процессов, основы формальных языков и грамматик, теорию трансляции программ	ОПК-2 З1
		уметь	
		У1 - использовать математические модели языков программирования для разработки алгоритмов лексического и синтаксического анализа	ОПК-2 У1
		владеть	
		В1 - навыками описания программных конструкций и их элементов средствами математического аппарата формальных грамматик и языков	ОПК-2 В1

1	2	3	4
ОПК-4	Способность применять в профессиональной деятельности основные методы и средства автоматизации проектирования, производства, испытаний и оценки качества программного обеспечения	Знать	
		31 - математические модели, используемые при разработке, отладке, верификации и тестировании программ	ОПК-4 31
		32 - базовые методы и подходы к построению алгоритмов (итерация и рекурсия, модульный, структурный и объектно-ориентированный методы алгоритмизации)	ОПК-4 32
		уметь	
		У1 - разрабатывать модели прикладных программ, используемые при структурном тестировании	ОПК-4 У1
		У2 - оценивать эффективность алгоритмов по затратам времени и машинной памяти	ОПК-4 У2
		владеть	
		В1 - методами нисходящего и восходящего проектирования алгоритмов	ОПК-4 В1
		В2 - навыками программирования и трансляции прикладных программ с использованием современных систем программирования	ОПК-4 В2
ПК-5	Готовность к использованию современных системных программных средств: операционных систем, операционных и сетевых оболочек, сервисных программ	знать	
		31 - основные возможности и особенности современных операционных систем и систем программирования	ПК-5 31
		32 - функции и команды операционных систем, оболочек и сервисных программ, основные компоненты систем программирования и их назначение	ПК-5 32
		Уметь	
		У1 - использовать системное программное обеспечение персонального компьютера для решения прикладных задач	ПК-5 У1
		У2 - применять современные технологии программирования, отладки и тестирования программных модулей лексического и синтаксического анализа	ПК-5 У2
		Владеть	
		В1 - средствами операционных систем и оболочек для организации выполнения прикладных программ на ЭВМ	ПК-5 В1
		В2 - навыками настройки систем программирования и операционных систем при разработке и отладке прикладных программ	ПК-5 В2

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

№	Содержание оценочного средства	Индекс оцениваемой компетенции и ее элементов
1	Понятие вычислительного процесса и ресурса. Состояния процесса	ПК-5 31 У1 В1
2	Классификация вычислительных процессов	ПК-5 32 В1 В2
3	Вычислительные процессы и системное программное обеспечение	ОПК-4 31 В2 ПК-5 31 32 У1 В1 В2
4	Основные компоненты системы программирования	ОПК-4 В2 ПК-5 31 32 В2
5	Основные виды трансляторов и их эффективность	ОПК-2 31 ОПК-4 В2 ПК-5 32 В2
6	Понятие формального языка. Синтаксис и семантика	ОПК-2 31 У1 В1
7	Определение формальной грамматики	ОПК-2 31 У1 В1
8	Классификация формальных грамматик	ОПК-2 31 У1 В1
9	Математическое определение формального языка	ОПК-2 31 У1 В1
10	Использование формальных грамматик для описания синтаксиса языков программирования	ОПК-2 31 У1 В1 ПК-5 У2
11	Метаязыки и их применение для описания синтаксиса языков программирования	ОПК-2 31 У1 В1 ПК-5 У2
12	Синтаксические диаграммы	ОПК-2 31 У1 В1 ОПК-4 32 В1 В2 ПК-5 У2
13	Фазы трансляции программ	ОПК-2 31 У1 В1 ОПК-4 31 32 У1 В1 В2 ПК-5 31 32
14	Основные функции сканера	ОПК-2 31 У1 В1 ОПК-4 32 В2 ПК-5 У2
15	Регулярные грамматики и конечные автоматы	ОПК-2 31 У1 В1
16	Принципы работы сканера	ОПК-2 31 У1 В1 ПК-5 У2
17	Процедура сканирования для распознавания десятичных чисел	ОПК-2 31 У1 В1 ОПК-4 32 В1 В2 ПК-5 У2
18	Процедура сканирования для распознавания идентификаторов, целых и десятичных чисел	ОПК-2 31 У1 В1 ОПК-4 32 В1 В2 ПК-5 У2
19	Классификация методов грамматического разбора	ОПК-2 31 ОПК-4 31 32
20	Проблема однозначности грамматик	ОПК-2 31 ОПК-4 31 32

21	Нисходящий грамматический разбор с возвратами	ОПК-2 31 У1 В1 ОПК-4 32 В1 В2 ПК-5 У2
22	Необходимые условия детерминированного разбора языков, описываемых КС-грамматиками	ОПК-2 31 ОПК-4 31 32
23	Достаточные условия детерминированного разбора языков, описываемых КС-грамматиками	ОПК-2 31 ОПК-4 31 32
24	Синтаксический анализ методом рекурсивного спуска	ОПК-2 31 У1 В1 ОПК-4 32 У2 В1 В2 ПК-5 У2
25	Нисходящий грамматический разбор на основе синтаксических диаграмм	ОПК-2 31 У1 В1 ОПК-4 32 У2 В1 В2 ПК-5 У2
26	Основа сентенциальной формы и грамматики простого предшествования	ОПК-2 31 ОПК-4 31 32
27	Граматики предшествования операторов и первичная фраза сентенциальной формы	ОПК-2 31 ОПК-4 31 32
28	Вычисление отношений предшествования в грамматиках предшествования операторов	ОПК-2 31 ОПК-4 31 32
29	Грамматический разбор методом операторного предшествования	ОПК-2 31 У1 В1 ОПК-4 32 У2 В1 В2 ПК-5 У2
30	Синтаксический анализ скобочных структур	ОПК-2 31 У1 В1 ОПК-4 32 У2 В1 В2 ПК-5 У2
31	Основные функции и построение семантического анализатора программ	ОПК-2 31 ОПК-4 В2 ПК-5 31
32	Семантическое дерево как форма представления программ в трансляторах	ОПК-2 31 У1 В1 ОПК-4 31 ПК-5 32
33	Польская запись как форма представления программ в трансляторах	ОПК-2 31 У1 В1 ОПК-4 31 ПК-5 32
34	Тетрадная форма представления программ в трансляторах	ОПК-2 31 У1 В1 ОПК-4 31 ПК-5 32
35	Алгоритм перевода выражений в польскую запись	ОПК-2 31 У1 В1 ОПК-4 32 У1 У2 В1 В2 ПК-5 31 32
36	Вычисление выражений, представленных в польской записи	ОПК-2 31 У1 В1 ОПК-4 32 У1 У2 В1 В2 ПК-5 31 32
37	Генерация тетрадной формы выражений	ОПК-2 31 У1 В1 ОПК-4 32 У1 У2 В1 В2 ПК-5 31 32
38	Машинно-независимая оптимизация линейных участков программ	ОПК-2 31 У1 В1 ОПК-4 32 У1 У2 В1 В2 ПК-5 31 32
39	Машинно-независимая оптимизация циклических участков программ	ОПК-2 31 У1 В1 ОПК-4 32 У1 У2 В1 В2

		ПК-5 31 32
40	Генерация объектного кода по списку тетрад	ОПК-2 31 У1 В1 ОПК-4 32 У1 У2 В1 В2 ПК-5 31 32
41	Генерация объектного кода по семантическому дереву	ОПК-2 31 У1 В1 ОПК-4 32 У1 У2 В1 В2 ПК-5 31 32
42	Машинно-зависимая оптимизация объектного кода	ОПК-2 31 У1 В1 ОПК-4 32 У1 У2 В1 В2 ПК-5 У1 31 32
43	Организация информационных таблиц транслятора	ОПК-4 32 У2 В1 В2 ПК-5 31
44	Назначение и принципы построения таблицы идентификаторов	ОПК-4 32 У2 В1 В2 ПК-5 31
45	Упорядоченные и неупорядоченные таблицы. Линейный и бинарный поиск в информационных таблицах	ОПК-4 32 У2 В1 В2 ПК-5 31
46	Хеш-адресация в информационных таблицах	ОПК-2 31 ОПК-4 32 У2
47	Методы вычисления хеш-функции	ОПК-2 31 У1 В1 ОПК-4 32 У1 У2 В1 В2 ПК-5 31 32
48	Разрешение коллизий в хеш-таблицах методом разрешения	ОПК-2 31 У1 В1 ОПК-4 32 У1 У2 В1 В2 ПК-5 31 32
49	Разрешение коллизий в хеш-таблицах методом цепочек	ОПК-2 31 У1 В1 ОПК-4 32 У1 У2 В1 В2 ПК-5 31 32
50	Реализация операций поиска и записи в хеш-таблицах по методу цепочек	ОПК-2 31 У1 В1 ОПК-4 32 У1 У2 В1 В2 ПК-5 31 32
51	Задача распределения регистровой памяти	ОПК-2 31 У1 В1 ОПК-4 У2
52	Алгоритм оптимизации загрузки регистровой памяти	ОПК-2 31 У1 ОПК-4 32 У1 У2 В1 В2 ПК-5 31 32
53	Распределение оперативной памяти при трансляции программ	ОПК-2 31 У1 В1 ОПК-4 У2
54	Размещение структурированных данных в оперативной памяти	ОПК-2 31 У1 В1 ОПК-4 У2
55	Передача данных между блоками программы	ОПК-2 31 У1 В1 ОПК-4 У2
56	Распределение оперативной памяти при выполнении программы	ОПК-2 31 У1 В1 ОПК-4 У2
57	Графовые модели структурной организации программ.	ОПК-2 31 ОПК-4 31 32 У1
58	Структурное тестирование программ	ОПК-2 31 ОПК-4 31 32 У1 У2 В1 В2
59	Функциональное тестирование программ	ОПК-4 31 32 У1 У2 В1 В2 ПК-5 У1 У2 В1 В2

ПОКАЗАТЕЛИ И КРИТЕРИИ ОЦЕНИВАНИЯ КОМПЕТЕНЦИЙ

(Шкалы оценивания)

Результаты выполнения обучающимся заданий на экзамене оцениваются по пятибалльной шкале.

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

«Отлично» (5) – 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 разнообразными навыками и приемами выполнения практических задач.

«Хорошо» (4) - 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) - 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) - 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.