


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
«31» августа 2020 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

ТЕОРИЯ ВЫЧИСЛИТЕЛЬНЫХ ПРОЦЕССОВ И СТРУКТУР

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, 2020

Вводная часть

1. Цели освоения дисциплины

Целью освоения дисциплины «Теория вычислительных процессов и структур» является формирование у обучающихся общепрофессиональных 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.О.11.01.« Теория вычислительных процессов и структур» относится к модулю Программирование III базовой части Блока 1.

2.2. Для изучения дисциплины «Теория вычислительных процессов и структур» необходимы предшествующие дисциплины:

- «Основы программирования»;
- «Дискретная математика»;
- «Структуры и алгоритмы компьютерной обработки данных»;
- «Операционные системы и оболочки».

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

- Подготовка к процедуре защиты и защита выпускной квалификационной работы;

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 компетенций (ОПК):

№	Индекс компетенции	Код и наименование индикатора достижения компетенции	Перечень планируемых результатов обучения по дисциплине В результате изучения дисциплины обучающиеся должны:		
			Знать:	Уметь:	Владеть:
1	ОПК-2. Способен применять современный математический аппарат, связанный с проектированием, разработкой, реализацией и оценкой качества программных продуктов и программных комплексов в различных областях человеческой деятельности	ОПК-2.1. Осуществляет отбор современных математических методов, моделей и алгоритмов, используемых при проектировании, разработке, реализации и оценке качества программных продуктов в различных областях человеческой деятельности	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
		ОПК-2.2. Способен применять современных математические методы, модели и алгоритмы при проектировании, разработке, реализации, оценке качества и анализа эффективности программных продуктов и программных комплексов	- 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

2	ОПК-3. Способен применять современные информационные технологии, в том числе отечественные, при создании программных продуктов и программных комплексов различного назначения	ОПК-3.3. Способен применять современные компиляторы, отладчики и оптимизаторы программного кода, осуществлять сборку модулей и компонентов программного кода, разрабатывать и реализовывать межмодульный интерфейс.	<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 	<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 	<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
---	---	---	--	---	---

ОСНОВНАЯ ЧАСТЬ

1. Объем дисциплины и виды учебной работы

Вид учебной работы	Всего часов	Семестры
		№7 часов
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	20	20
Подготовка к выполнению лабораторных работ	32	32
Подготовка к защите лабораторных работ	20	20
Контроль	36	36
Вид промежуточной аттестации	зачет (З), экзамен (Э)	+
ИТОГО: общая трудоемкость	часов зач. ед.	180 5

Дисциплина частично реализуется с применением дистанционных образовательных технологий с использованием платформы Microsoft Teams, ЭИОС Moodle, корпоративной электронной почты.

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.

1	2	3	4
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. Перечень лабораторных работ

Лабораторная работа № 1. Польская запись, как форма представления программных конструкций в трансляторах

Лабораторная работа № 2. Тетрадная форма представления программных конструкций в трансляторах

Лабораторная работа № 3. Изучение и программная реализация методов лексического анализа, используемых в компиляторах

Лабораторная работа № 4. Изучение и программная реализация нисходящих методов грамматического разбора на примере метода рекурсивного спуска

Лабораторная работа № 5. Изучение и программная реализация восходящих методов грамматического разбора на примере метода операторного предшествования

Лабораторная работа № 6. Исследование и программная реализация методов генерации и оптимизации объектного кода (для простой системы команд учебной ЭВМ)

Лабораторная работа № 7. Исследование и программная реализация методов машинно-независимой оптимизации программ (для линейных и циклических участков)

Лабораторная работа № 8. Исследование методов поиска данных в информационных таблицах

Лабораторная работа № 9. Исследование и программная реализация методов распределения регистровой памяти в компиляторах

3. САМОСТОЯТЕЛЬНАЯ РАБОТА СТУДЕНТА

Самостоятельная работа осуществляется в объеме 72 часов.

Видами СРС являются:

- изучение литературы и других источников
- подготовка к выполнению лабораторных работ
- подготовка к защите лабораторных работ

Формами текущего контроля успеваемости являются:

- защита лабораторной работы.

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
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: 31.08.2020).
2	Малявко, А. А. Системное программное обеспечение. Формальные языки и методы трансляции [Электронный ресурс] : учебное пособие : в 3 ч. Ч. 1. / А. А. Малявко. – Новосибирск : НГТУ, 2010. – 104 с. – Режим доступа: http://biblioclub.ru/index.php?page=book&id=228974&sr=1 (дата обращения: 31.08.2020).
3	Молчанов, А. Ю. Системное программное обеспечение [Текст] : учебник / А. Ю. Молчанов. – Санкт-Петербург : Питер, 2003. – 396 с.

5.2. Дополнительная литература

№ п/п	Автор (ы), наименование, место издания и издательство, год
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	Таненбаум, Э. Современные операционные системы [Текст] / Таненбаум Э. - СПб.: БХВ-Петербург, 2004. - 1040 с.

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: 31.08.2020).
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: 31.08.2020).
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: 31.08.2020).
4. Znanium.com [Электронный ресурс] : [база данных]. – Доступ к полным текстам по паролю. – Режим доступа: <http://znanium.com> (дата обращения: 31.08.2020).
5. «Издательство «Лань» [Электронный ресурс] : электронная библиотека. – Доступ к полным текстам по паролю. – Режим доступа: <http://e-lanbook.com> (дата обращения: 31.08.2020).
6. Университетская библиотека ONLINE [Электронный ресурс] : электронная библиотека. – Доступ к полным текстам по паролю. – Режим доступа: <http://www.biblioclub.ru> (дата обращения: 31.08.2020).
7. Юрайт [Электронный ресурс] : электронная библиотека. – Доступ к полным текстам по паролю. – Режим доступа: <http://www.biblio-online.ru> (дата обращения: 31.08.2020).
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: 31.08.2020).

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: 31.08.2020).
2. Единая коллекция цифровых образовательных ресурсов [Электронный ресурс] : федеральный портал. – Режим доступа: <http://school-collection.edu.ru/>, свободный (дата обращения: 31.08.2020).
3. Единое окно доступа к образовательным ресурсам [Электронный ресурс] : федеральный портал. – Режим доступа: <http://window.edu.ru/>, свободный (дата обращения: 31.08.2020).
4. Интернет Университет Информационных технологий. [Электронный ресурс] : сайт. – Режим доступа: <http://www.intuit.ru/>, свободный (дата обращения: 31.08.2020).
5. Петров Д.Н. Парадигмы программирования. [Электронный ресурс] : сайт. –

Режим доступа: <http://dnpetrov.narod.ru/>, свободный (дата обращения 31.08.2020).

6. Портал естественных наук. [Электронный ресурс] : сайт. – Режим доступа: <http://e-science11.ru>, свободный (дата обращения 31.08.2020).

7. Портал для программистов и администраторов информационных систем. [Электронный ресурс] : сайт. – Режим доступа: <http://www.coderpost.net/>, свободный (дата обращения 31.08.2020).

8. Российский общеобразовательный портал [Электронный ресурс] : образовательный портал. – Режим доступа: <http://www.school.edu.ru/>, свободный (дата обращения: 31.08.2020).

9. Сервер Информационных Технологий [Электронный ресурс] : сайт. – Режим доступа: <http://citforum.ru/>, свободный (дата обращения 31.08.2020).

10. Сайт программирования. [Электронный ресурс] : сайт. – Режим доступа: <http://www.cyberguru.ru/>, свободный (дата обращения 31.08.2020).

11. Федеральный центр информационно-образовательных ресурсов [Электронный ресурс] : Единое окно доступа к образовательным ресурсам. – Режим доступа: <http://fcior.edu.ru>, свободный (дата обращения: 31.08.2020).

5.5. Периодические издания

1. Компьютерные и информационные науки. Доступ: Киберленинка [Электронный ресурс] : научная электронная библиотека. – Режим доступа: <https://cyberleninka.ru/article/c/computer-and-information-sciences>, свободный (дата обращения: 31.08.2020).

2. Электротехника, электронная техника, информационные технологии. Доступ: Киберленинка [Электронный ресурс] : научная электронная библиотека. – Режим доступа: <https://cyberleninka.ru/article/c/electrical-electronic-information-engineering>, свободный (дата обращения: 31.08.2020).

3. Архив журнала «Системный администратор» [Электронный ресурс]. – Режим доступа: <http://samag.ru/archive>, свободный (дата обращения: 31.08.2020).

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, система программирования RAD Studio 10.1 Berlin Professional Concurrent ELC и др.

6.3. Требования к специализированному оборудованию: *отсутствует*

7. 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; пометать 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 вычислительных процессов, тестирование и верификация.</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 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.</p>

Подготовка к экзамену	При подготовке к экзамену необходимо ориентироваться на конспекты лекций, рекомендуемую литературу, результаты выполнения лабораторных работ и другие учебные материалы.
-----------------------	--

8. 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.

Для организации учебной работы может использоваться набор веб-сервисов MS office365, вебинарная платформа РГУ имени С.А. Есенина, университетская информационно-образовательная среда Moodle, облачные технологии. Координация учебной работы осуществляется через университетскую электронную почту.

9. Требования к программному обеспечению учебного процесса

1. Операционная система Windows Pro (договор №65/2019 от 02.10.2019);
2. Антивирус Kaspersky Endpoint Security (договор №14-ЗК-2020 от 06.07.2020г.);
3. Офисное приложение LibreOffice (свободно распространяемое ПО);
4. Среда разработки приложений RAD Studio 10.1 Berlin Professional Concurrent ELC (договор № 11\05\2016-9774 11.05.16г.)
5. Система программирования Python (свободно распространяемое ПО);
6. Система программирования PascalABC (свободно распространяемое ПО);
7. Архиватор 7-zip (свободно распространяемое ПО);
8. Браузер изображений Fast Stone Image Viewer (свободно распространяемое ПО);
9. PDFридер Foxit Reader (свободно распространяемое ПО);
10. Медиа проигрыватель VLC mediaplayer (свободно распространяемое ПО);

ПО);


11. Запись дисков ImageBurn (свободно распространяемое ПО);

12. DJVU браузер DjVu Browser Plug-in (свободно распространяемое ПО)

13. Набор веб-сервисов MS office365 (бесплатное ПО для учебных заведений <https://www.microsoft.com/ru-ru/education/products/office>);

14. Система электронного обучения Moodle (свободно распространяемое ПО).

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
«31» августа 2020 г.

АННОТАЦИЯ РАБОЧЕЙ ПРОГРАММЫ ДИСЦИПЛИНЫ
ТЕОРИЯ ВЫЧИСЛИТЕЛЬНЫХ ПРОЦЕССОВ И СТРУКТУР

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

Направленность (профиль) подготовки
Администрирование информационных систем

Квалификация
Бакалавриат

Форма обучения
Очная

Рязань, 2020

1. Цель освоения дисциплины

Целью освоения дисциплины «Теория вычислительных процессов и структур» является формирование у обучающихся общепрофессиональных компетенций в процессе изучения теоретических основ создания трансляторов, верификации и тестирования прикладных программ для последующего применения в учебной и практической деятельности.

2. Место дисциплины в структуре ОПОП

Дисциплина Б1.О.11.01 «Теория вычислительных процессов и структур» относится к модулю Программирование III базовой части Блока 1.

Дисциплина изучается на 4 курсе (7 семестр)

3. Трудоемкость дисциплины: 5 зачетных единиц, 180 академических часов.

4. Планируемые результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения образовательной программы и индикаторами достижения компетенций:

ОПК-2.1 – знать математические модели вычислительных процессов, основы формальных языков и грамматик, теорию трансляции программ; уметь использовать математические модели языков программирования для разработки алгоритмов лексического и синтаксического анализа; владеть навыками описания программных конструкций и их элементов средствами математического аппарата формальных грамматик и языков.

ОПК-2.2 – 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); уметь разрабатывать модели прикладных программ, используемые при структурном тестировании; оценивать эффективность алгоритмов по затратам времени и машинной памяти; владеть методами нисходящего и восходящего проектирования алгоритмов; навыками программирования и трансляции прикладных программ с использованием современных систем программирования.

ОПК-3.3 – знать 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; уметь использовать системное программное обеспечение персонального компьютера для решения прикладных задач; применять современные технологии программирования, отладки и тестирования программных модулей лексического и синтаксического анализа; владеть 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.

5. Форма промежуточной аттестации и семестр прохождения

Экзамен (7 семестр).

Дисциплина реализуется частично с применением дистанционных образовательных технологий.