

**ГОУ ВПО «ДОНЕЦКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ»**  
**ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ**

Кафедра теории упругости и вычислительной математики  
им. академика А.С. Космодамианского

**УТВЕРЖДАЮ:**

Проректор по научно-методической  
и учебной работе

Е.И. Скафа

«22» декабря 2016 г.



**Рабочая программа учебной дисциплины**  
**ТЕОРИЯ АВТОМАТОВ И ФОРМАЛЬНЫХ ЯЗЫКОВ**

Направление подготовки:	01.03.02 Прикладная математика и информатика.
Профиль подготовки:	Статистика.
Образовательная программа:	бакалавриат
Квалификация:	Академический бакалавр
Форма обучения:	<u>очная</u> , очно-заочная, заочная нужное подчеркнуть

Донецк 2020

**УТВЕРЖДАЮ:**

Декан факультета математики  
и информационных технологий

\_\_\_\_\_ И.А. Моисеенко

«16» апреля 2020



Программа учебной дисциплины «Теория автоматов и формальных языков» составлена на основании Государственного образовательного стандарта высшего профессионального образования (ГОС ВПО) Донецкой Народной Республики (ДНР) по направлению подготовки 01.03.02 Прикладная математика и информатика, утвержденного приказом Министерства образования и науки ДНР от «04» апреля 2016 г. № 280; Порядка организации учебного процесса в образовательных организациях высшего профессионального образования Донецкой Народной Республики, утвержденного приказом Министерства образования и науки ДНР № 1171 от «10» ноября 2017 г.; учебного плана и основной образовательной программы высшего профессионального образования направления подготовки 01.03.02 Прикладная математика и информатика (профиль: Статистика), разработанных в ГОУ ВПО «Донецкий национальный университет».

Разработчик:

проф. каф. ТУиВМ, , д-р. ф.-м.н., доцент,  
(должность, степень, звание, кафедра)

\_\_\_\_\_ И.А. Моисеенко

Программа учебной дисциплины утверждена на заседании кафедры теории упругости и вычислительной математики имени академика А.С. Космодамианского

Протокол № 11 от «9» апреля 2020 г.

Заведующий кафедрой

\_\_\_\_\_ В.И. Сторожев

Программа учебной дисциплины одобрена учебно-методической комиссией факультета математики и информационных технологий

Протокол № 8 от «15» апреля 2020 г.

Председатель учебно-методической  
комиссии факультета

\_\_\_\_\_

Л.И. Селякова

## 1. ОБЛАСТЬ ПРИМЕНЕНИЯ И МЕСТО ДИСЦИПЛИНЫ В УЧЕБНОМ ПРОЦЕССЕ

Учебная дисциплина «Теория автоматов и формальных языков» относится к вариативной части профессионального блока учебного плана по направлению подготовки 01.03.02 Прикладная математика и информатика профиль Статистика. Для изучения данной учебной дисциплины необходимы знания, умения и навыки, формируемые предшествующими и сопутствующими дисциплинами: "Языки программирования", "Основы программирования", "Архитектура вычислительных систем", "Дискретная математика", "Алгоритмы и анализ сложности". Освоение данной дисциплины необходимо как предшествующее для таких учебных дисциплин, как теория вероятностей и математическая статистика, математическая логика и теория множеств, теория автоматов и формальных языков, теория алгоритмов. Возникнув в результате усилий, направленных на выработку точных методов описания естественных языков, теория грамматик столкнулась, с одной стороны, с теоретической лингвистикой, с другой - с теорией алгоритмов и автоматов. Понятие и методы, используемые этими теориями, получили в формальных грамматиках своеобразную интерпретацию и были развиты в новых направлениях. Практические применения теории грамматик, прежде всего, связаны с лингвистикой, для которой формальные грамматики – это удобный метаязык. Кроме того, формальные грамматики широко применяются для определения языков программирования и разработки эффективных алгоритмов трансляции для них.

Предварительные требования к студентам.

1. Знание одного из классических процедурно-ориентированных или объектно-ориентированных языков, предпочтительно языка C/C++.
2. Освоение курсов "Языки программирования", "Основы программирования", "Архитектура вычислительных систем", "Дискретная математика", "Алгоритмы и анализ сложности".

## 2. СТРУКТУРА ДИСЦИПЛИНЫ (МОДУЛЯ)

<i>Характеристика учебной дисциплины</i>				
Направление подготовки	01.03.02 Прикладная математика и информатика			
Профиль	Статистика			
Образовательная программа	бакалавриат			
Квалификация	академический бакалавр			
Количество содержательных модулей				
Дисциплина базовой / вариативной части образовательной программы	вариативная часть			
Формы контроля (МК, экзамен, зачет)	1 модульный контроль в 1, 2 семестрах, 1 дифференциальный зачет в 1 семестре, 1 экзамен во втором семестре			
Показатели	очная форма обучения		заочная форма обучения	
		ускор. срок	нормат. срок	ускор. срок
Количество зачетных единиц (кредитов)	4			
Год подготовки	3			
Семестр	5			
Количество часов	144			
- лекционных	54			
- практических, семинарских				

- лабораторных	36			
- самостоятельной работы	54			
в т.ч. индивидуальное задание				
Недельное количество часов,	8			
в т.ч. аудиторных	5			

### 3. ОПИСАНИЕ ДИСЦИПЛИНЫ

#### Цели и задачи

##### Цели учебной дисциплины.

1. Изучение основных аспектов теории формальных языков, существенных с точки зрения трансляции.
2. Изучение свойств формальных моделей представления и обработки информации.
3. Изучение основного механизма определения языков – формальных грамматик, математическим формализмом для теоретического изучения в определенном аспекте "внутреннего строения" рекурсивных множеств некоторых специальных классов.
4. Изучение алгоритмов и технических приемов, имеющих широкое использование при построении современных трансляторов языков.

##### Задачи учебной дисциплины:

1. Исследование и разработка математических моделей, алгоритмов, методов, программного обеспечения.
2. Разработка программного и информационного обеспечения автоматизированных систем вычислительных комплексов.
3. Разработка и исследование алгоритмов, вычислительных моделей и моделей данных для реализации элементов новых (или известных) сервисов систем информационных технологий.
4. Разработка архитектуры, алгоритмических и программных решений системного и прикладного программного обеспечения.
5. Изучение и разработка языков программирования, алгоритмов, библиотек и пакетов программ, продуктов системного и прикладного программного обеспечения.

#### Требования к результатам освоения дисциплины

Процесс изучения дисциплины направлен на формирование элементов следующих компетенций в соответствии с ГОС ВПО по направлению подготовки 01.03.02 «Прикладная математика и информатика»:

##### **а) общекультурных (ОК):**

- способностью к самоорганизации и самообразованию (ОК-7);

##### **б) общепрофессиональных (ОПК):**

- способностью использовать базовые знания естественных наук, математики и информатики, основные факты, концепции, принципы теорий, связанных с фундаментальной информатикой и информационными технологиями (ОПК-1);
- способностью применять в профессиональной деятельности современные языки программирования и языки баз данных, методологии системной инженерии, системы автоматизации проектирования, электронные библиотеки и коллекции, сетевые технологии, библиотеки и пакеты программ, современные профессиональные стандарты информационных технологий (ОПК-2);
- способностью к разработке алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, созданию информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям (ОПК-3);

- способностью решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности (ОПК-4);

**в) профессиональных (ПК):**

**научно-исследовательская деятельность:**

- способностью понимать, совершенствовать и применять современный математический аппарат, фундаментальные концепции и системные методологии, международные и профессиональные стандарты в области информационных технологий (ПК-2);

**проектная и производственно-технологическая деятельность:**

- способностью разрабатывать и реализовывать процессы жизненного цикла информационных систем, программного обеспечения, сервисов информационных технологий, а также методы и механизмы оценки и анализа функционирования средств и систем информационных технологий (ПК-7);

**организационно-управленческая деятельность:**

- способностью разрабатывать, оценивать и реализовывать процессы жизненного цикла информационных систем, программного обеспечения, сервисов информационных технологий, а также реализовывать методы и механизмы оценки и анализа функционирования средств и информационных технологий; разрабатывать проектную и программную документацию, удовлетворяющую нормативным требованиям (ПК-9);

**В результате изучения учебной дисциплины студент должен**

**Знать:**

- типы языковых процессоров;
- основные фазы трансляции;
- методы работы и синтеза лексического анализатора;
- методы работы и синтеза синтаксического анализатора;
- методы работы и синтеза семантического анализатора;
- способы представления в программных системах конечных автоматов;
- формализм регулярных множеств и регулярных выражений;
- соотношения праволинейных грамматик и конечных автоматов;
- классификацию порождающих грамматик по Холмскому;
- методы работы и синтеза магазинных автоматов;
- свойства LL(k)-грамматик;
- методы денотационных и операционных семантик;
- методы семантических подпрограмм;
- методы задания семантики языка программирования;
- подходы к оптимизации кода программ;
- подходы к реализации ассемблеров.

**Уметь:**

- строить программы лексических анализаторов;
- программировать интерпретаторы формальных грамматик;
- программно реализовывать деревья вывода;
- строить LL(1) таблицы для работы синтаксического анализатора;
- представлять семантический терм программы.

**Владеть:**

- навыками использования контекстно-свободных грамматик и синтаксически управляемого перевода для программной реализации компиляторов.

#### **4. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ И ФОРМЫ ОРГАНИЗАЦИИ УЧЕБНОГО ПРОЦЕССА**

В рамках изучения дисциплины предусмотрены следующие формы организации учебного процесса: лекции, лабораторные занятия, самостоятельная работа студента.

Лекционные занятия направлены на овладение теоретическими основами дисциплины, лабораторные – на овладение методами (алгоритмами) практического решения задач профессиональной деятельности.

Самостоятельная работа студентов предусматривает выполнение индивидуальных заданий, подготовку к лекционным и лабораторным занятиям, изучение учебно-методической литературы, составление конспектов, подготовку презентаций и докладов.

Текущий контроль осуществляется путем защиты индивидуальных заданий, модульной контрольной работы по проверке знаний теоретических положений (понятий, определений, алгоритмов и доказательств их работы).

В учебном процессе применяются активные и интерактивные формы проведения занятий, внеаудиторная самостоятельная работа, балльно-рейтинговая система оценки успеваемости, личностно-ориентированное обучение, проблемное обучение.

Материал излагается с использованием объяснительно-иллюстративных, эвристических и исследовательских методов преподавания. При проведении лекций для обсуждения материала широко используются методы математического и компьютерного моделирования. Также проводятся лекции проблемные, с заранее запланированными ошибками.

<b>Порядковый номер и тема</b>	<b>Краткое содержание темы</b>
<b>Модуль 1. Модель идеализированного компилятора и лексический аспект в языковых процессорах</b>	
<b>Содержательный модуль 1.1. Математическая и информационная модель идеализированного компилятора</b>	
<b>Тема 1.</b> Модель языкового процессора	<p>Описание и цель курса. Характеристика его роли в схеме подготовки специалиста. Связь с другими курсами рабочего учебного плана. Тематический план курса. Знания и умения. Литература основная и дополнительная. Понятие и типы языковых процессоров. Способы задания языков программирования.</p> <p>Основные фазы языкового процессора: лексический анализатор; синтаксический анализатор; семантический анализатор; оптимизатор кода. Интерпретация семантического термина программы в форме семантического дерева. Критерии оптимизации семантического термина программы. Машинно-независимые и машинно-зависимые методы оптимизации.</p> <p>Использование таблиц для накопления информации о структуре и типе переменных в программе. Анализ ошибок. Генератор объектного кода. Группировка фаз и проходы компилятора. Стадии компилятора. Проходы компилятора. Математическая модель перевода.</p>
<b>Тема 2.</b> Языки	Языки. Способы задания языка. Операции над языками.
<b>Тема 3.</b> Формальные грамматики	Формальные грамматики. Терминалы, нетерминалы, правила, выводимые цепочки. Язык, определяемый грамматикой. Форма Бэкуса-Наура. Грамматики с ограничениями на правила. Классификация грамматик по Хомскому.
<b>Тема 4.</b> Абстрактные распознаватели	Классификация абстрактных распознавателей. Машина Тьюринга. Схема абстрактного распознавателя. Конфигурации и такты. Допускаемые и не допускаемые цепочки. Языки, определяемые

	<p>(распознаваемые, допускаемые) абстрактным распознавателем.</p> <p>Детерминированные и недетерминированные конечные распознаватели. <u>Алгоритм</u> преобразования недетерминированного конечного распознавателя в детерминированный.</p> <p>Детерминированные и недетерминированные распознаватели с магазинной памятью. Линейно ограниченные распознаватели. Классы эквивалентности формальных грамматик и абстрактных распознавателей.</p>
<p align="center"><b>Содержательный модуль 1.2.</b></p> <p align="center"><b>Лексический аспект в языковых процессорах. Разработка лексических анализаторов на основе конечных распознавателей</b></p>	
<p><b>Тема 5.</b></p> <p>Регулярные выражения и праволинейные грамматики</p>	<p>Регулярные множества и регулярные выражения. Свойства регулярных выражений. Регулярные выражения и праволинейные грамматики. Алгоритм построения праволинейной грамматики <math>G</math> для заданных двух произвольных праволинейных грамматик <math>G_1</math> и <math>G_2</math> так, чтобы <math>L(G) = L(G_1) \cup L(G_2)</math>. Доказательство эквивалентности порождаемых языков.</p> <p>Алгоритм построения праволинейной грамматики <math>G</math> для заданных двух произвольных праволинейных грамматик <math>G_1</math> и <math>G_2</math> так, чтобы <math>L(G) = L(G_1)L(G_2)</math>. Алгоритм построения праволинейной грамматики <math>G</math> для заданной произвольной праволинейной грамматики <math>G_1</math> так, чтобы <math>L(G) = L(G_1)^*</math>. Доказательство эквивалентности порождаемых языков.</p> <p>Уравнения с регулярными коэффициентами. Системы уравнений с регулярными коэффициентами. Методы построения решений. Алгоритм построения регулярного выражения для заданной произвольной праволинейной грамматики.</p>
<p><b>Тема 6.</b></p> <p>Конечные распознаватели и регулярные выражения</p>	<p>Конечные распознаватели и регулярные выражения. Алгоритм построения конечного распознавателя <math>M</math> для заданных двух произвольных конечных распознавателей <math>M_1</math> и <math>M_2</math> так, чтобы <math>L(M) = L(M_1) \cup L(M_2)</math>. Доказательство эквивалентности порождаемых языков.</p> <p>Алгоритм построения конечного распознавателя <math>M</math> для заданных двух произвольных конечных распознавателей <math>M_1</math> и <math>M_2</math> так, чтобы <math>L(M) = L(M_1)L(M_2)</math>. Алгоритм построения конечного распознавателя <math>M</math> для заданного произвольного конечного распознавателя <math>M_1</math> так, чтобы <math>L(M) = L(M_1)^*</math>. Доказательство эквивалентности порождаемых языков.</p>
<p><b>Тема 7.</b></p> <p>Конечные распознаватели и праволинейные грамматики</p>	<p>Конечные распознаватели и праволинейные грамматики. Алгоритм построения конечного распознавателя для заданной произвольной праволинейной грамматики. Алгоритм построения праволинейной грамматики для заданного произвольного конечного распознавателя. Доказательство эквивалентности порождаемых языков.</p>

<p><b>Тема 8.</b> Распознающие и обрабатывающие автоматы</p>	<p>Распознающие и обрабатывающие автоматы. Проблемы идентификации. Расширяющиеся конечные автоматы. Транслитераторы. Идентификация слов (метод автомата). Идентификация слов с нечёткими границами. Задача обнаружения префиксов.</p>
<p align="center"><b>Модуль 2.</b> <b>Синтаксический и семантический аспект в языковых процессорах</b></p>	
<p align="center"><b>Содержательный модуль 2.1.</b> <b>Разработка синтаксических анализаторов для класса <math>LL(1)</math>-грамматик на основе <math>ДМП_e</math>-распознавателей</b></p>	
<p><b>Тема 9.</b> КС-грамматики</p>	<p>КС-грамматики. Выводы. Деревья вывода. Левосторонняя и правосторонняя стратегии вывода. Однозначность грамматики. Приведение грамматик. <u>Алгоритм</u> поиска непродуктивных нетерминалов для КС-грамматик. <u>Алгоритм</u> устранения непродуктивных нетерминалов для КС-грамматик. <u>Алгоритм</u> поиска недостижимых символов для КС-грамматик. <u>Алгоритм</u> устранения недостижимых символов для КС-грамматик.</p> <p><u>Алгоритм</u> устранения бесполезных символов для КС-грамматик. <u>Алгоритм</u> построения множества аннулирующих нетерминалов для КС-грамматики. <u>Алгоритм</u> преобразования КС-грамматики к неукорачивающему виду. <u>Алгоритм</u> устранения цепных правил для КС-грамматик.</p> <p><u>Алгоритм</u> устранения непосредственной левой рекурсии для КС-грамматик. <u>Алгоритм</u> устранения левой рекурсии в общем виде для КС-грамматик. <u>Алгоритм</u> левой факторизации для КС-грамматик.</p> <p>Распознаватели с магазинной памятью, допускающим цепочку опустошением магазина. Такты и конфигурации <math>МП_e</math>-распознавателя. <u>Алгоритм</u> построения <math>МП</math>-распознавателя для заданного произвольного <math>МП_e</math>-распознавателя. <u>Алгоритм</u> построения <math>МП_e</math>-распознавателя для заданного произвольного <math>МП</math>-распознавателя. Классы эквивалентности <math>МП</math> и <math>МП_e</math> распознавателей.</p> <p>КС-грамматики и <math>МП</math>-распознаватели. <u>Алгоритм</u> построения <math>МП_e</math>-распознавателя для заданной произвольной КС-грамматики. <u>Алгоритм</u> построения КС-грамматики для заданного произвольного <math>МП_e</math>-распознавателя. Доказательство эквивалентности порождаемых языков.</p>
<p><b>Тема 10.</b> Предиктивный синтаксический анализатор</p>	<p>Методы обработки языка. Нисходящий анализ. S-грамматики. Определение. <u>Алгоритм</u> построения <math>ДМП_e</math>-распознавателя для заданной произвольной S-грамматики. Множества <math>FIRST_k(a)</math>. <math>LL(k)</math>-грамматики. <math>LL</math>-грамматики.</p> <p><math>LL(1)</math>-грамматики. Множества <math>FIRST(a)</math>. <u>Алгоритм</u> построения множеств <math>FIRST(a)</math>. Множества <math>FOLLOW(X)</math>. <u>Алгоритм</u> построения множеств <math>FOLLOW(X)</math>. Управляющая таблица разбора. <u>Алгоритм</u> построения управляющей таблицы разбора.</p>



	<p>Предиктивный синтаксический анализатор. <u>Алгоритм</u> построения предиктивного синтаксического анализатора для <math>LL(1)</math>-грамматик.</p> <p>Пример построения предиктивного синтаксического анализатора.</p>
<p align="center"><b>Содержательный модуль 2.2.</b></p> <p align="center"><b>Семантический аспект языков программирования</b></p>	
<p><b>Тема 11.</b></p> <p>Синтаксически управляемая трансляция</p>	<p>Синтаксически управляемая трансляция. Синтаксис выражений. Проблемы неоднозначности грамматик. Ассоциативность и приоритет операций. Синтаксически управляемые определения. Синтаксически управляемая трансляция.</p> <p>Информационная модель синтаксически управляемой трансляции.</p> <p>Практическая реализация синтаксически управляемой трансляции.</p>

## Тематический план

Названия содержательных модулей и тем	Содержательный модуль 1.1										
	Количество часов										
	Очная форма						Заочная форма				
	всего	в т.ч.					всего				
		лекции	практические	лабораторные	самостоятельная работа	индивидуальная работа		лекции	практические	лабораторные	самостоятельная работа
<b>Тема 1.</b> Модель языкового процессора	12	4	0	4	4						
<b>Тема 2.</b> Языки	11	2	0	4	5						
<b>Тема 3.</b> Формальные грамматики	11	2	0	4	5						
<b>Тема 4.</b> Абстрактные распознаватели	13	4	0	4	5						
<b>Итого по содержательному модулю 1.1.</b>	47	12	0	16	19						

Названия содержательных модулей и тем	Содержательный модуль 1.2.										
	Количество часов										
	Очная форма						Заочная форма				
	всего	в т.ч.					всего				
		лекции	практические	лабораторные	самостоятельная работа	индивидуальная работа		лекции	практические	лабораторные	самостоятельная работа
<b>Тема 5.</b> Регулярные выражения и праволинейные грамматики	12	6	0	2	4						
<b>Тема 6.</b> Конечные распознаватели и регулярные выражения	12	6	0	2	4						
<b>Тема 7.</b> Конечные распознаватели и праволинейные грамматики	10	4	0	2	4						
<b>Тема 8.</b> Распознающие и обрабатывающие автоматы	8	4	0	2	2						
<b>Итого по содержательному модулю 1.2.</b>	42	20	0	8	14						

Названия содержательных модулей и тем	Содержательный модуль 2.1.										
	Количество часов										
	Очная форма						Заочная форма				
	всего	в т.ч.					всего				
		лекции	практические	лабораторные	самостоятельная работа	индивидуальная работа		лекции	практические	лабораторные	самостоятельная работа
<b>Тема 9.</b> КС- грамматики	19	10	0	4	5						
<b>Тема 10.</b> Предиктивный синтаксический анализатор	19	8	0	4	7						
<b>Итого по содержательному модулю 2.1.</b>	38	18	0	8	12						

Названия содержательных модулей и тем	Содержательный модуль 2.2.										
	Количество часов										
	Очная форма						Заочная форма				
	всего	в т.ч.					всего				
		лекции	практические	лабораторные	самостоятельная работа	индивидуальная работа		лекции	практические	лабораторные	самостоятельная работа
<b>Тема 11.</b> Синтаксически управляемая трансляция	17	4	0	4	9						
<b>Итого по содержательному модулю 2.2.</b>	17	4	0	4	9						
<b>Итого по дисциплине</b>	144	54	0	36	54						

## 5. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ДЛЯ ПРОВЕДЕНИЯ ЛЕКЦИОННЫХ И ЛАБОРАТОРНЫХ ЗАНЯТИЙ

### Темы лекционных занятий

№ п/п	Название темы	Количество часов
1	Основные фазы языкового процессора: лексический анализатор; синтаксический анализатор; семантический анализатор; оптимизатор кода. Интерпретация семантического терма программы в форме семантического дерева.	2
2	Критерии оптимизации семантического терма программы. Машинно-независимые и машинно-зависимые методы оптимизации. Использование таблиц для накопления информации о структуре и типе переменных в программе.	2
3	Анализ ошибок. Генератор объектного кода. Группировка фаз и проходы компилятора. Стадии компилятора	2
4	Языки. Способы задания языка. Операции над языками.	2
5	Формальные грамматики. Терминалы, нетерминалы, правила, выводимые цепочки. Язык, определяемый грамматикой	2
6	Форма Бэкуса-Наура. Грамматики с ограничениями на правила. Классификация грамматик по Хомскому	2
7	Классификация абстрактных распознавателей. Машина Тьюринга. Схема абстрактного распознавателя. Конфигурации и такты. Допускаемые и не допускаемые цепочки	2
8	Языки, определяемые (распознаваемые, допускаемые) абстрактным распознавателем. Детерминированные и недетерминированные конечные распознаватели	2
9	Алгоритм преобразования недетерминированного конечного распознавателя в детерминированный. Детерминированные и недетерминированные распознаватели с магазинной памятью	2
10	Линейно ограниченные распознаватели. Классы эквивалентности формальных грамматик и абстрактных распознавателей	2
11	Регулярные множества и регулярные выражения. Свойства регулярных выражений. Регулярные выражения и праволинейные грамматики. Эквивалентность порождаемых языков	2
12	Уравнения с регулярными коэффициентами. Системы уравнений с регулярными коэффициентами. Методы построения решений. Алгоритм построения регулярного выражения для заданной произвольной праволинейной грамматики	2
13	Конечные распознаватели и регулярные выражения. Алгоритм построения конечного распознавателя $M$ для заданных двух произвольных конечных распознавателей $M_1$ и $M_2$ так, чтобы $L(M) = L(M_1) \cup L(M_2)$ . Доказательство эквивалентности порождаемых языков	2
14	Алгоритм построения конечного распознавателя $M$ для заданных двух произвольных конечных распознавателей $M_1$ и $M_2$ так, чтобы $L(M) = L(M_1)L(M_2)$ . Алгоритм построения конечного распознавателя $M$ для заданного произвольного	2

	конечного распознавателя $M_1$ так, чтобы $L(M) = L(M_1)^*$ . Доказательство эквивалентности порождаемых языков	
15	Конечные распознаватели и праволинейные грамматики. Алгоритм построения конечного распознавателя для заданной произвольной праволинейной грамматики.	2
16	Алгоритм построения праволинейной грамматики для заданного произвольного конечного распознавателя. Доказательство эквивалентности порождаемых языков	2
17	Распознающие и обрабатывающие автоматы. Проблемы идентификации. Расширяющиеся конечные автоматы	2
18	Транслитераторы. Идентификация слов (метод автомата). Идентификация слов с нечёткими границами. Задача обнаружения префиксов	2
19	КС-грамматики. Выводы. Деревья вывода. Левосторонняя и правосторонняя стратегии вывода. Однозначность грамматики. Приведение грамматик.	2
20	Распознаватели с магазинной памятью, допускающим цепочку опустошением магазина. Такты и конфигурации $МП_e$ - распознавателя.	2
21	Методы обработки языка. Нисходящий анализ. S-грамматики. Управляющая таблица разбора	2
22	Алгоритм построения управляющей таблицы разбора	2
23	Предиктивный синтаксический анализатор. Пример построения предиктивного синтаксического анализатора	2
24	Алгоритм построения предиктивного синтаксического анализатора для $LL(1)$ -грамматик.	2
25	Синтаксически управляемая трансляция. Синтаксис выражений. Проблемы неоднозначности грамматик. Ассоциативность и приоритет операций.	2
26	Синтаксически управляемые определения. Синтаксически управляемая трансляция.	2
27	Информационная модель синтаксически управляемой трансляции. Практическая реализация синтаксически управляемой трансляции	2
<b>Всего</b>		<b>54</b>

#### Темы лабораторных занятий

№ п/п	Название темы	Количество часов
1	Интерпретация семантического термина программы в форме семантического дерева.	2
2	Использование таблиц для накопления информации о структуре и типе переменных в программе.	2
3	Анализ ошибок. Генератор объектного кода. Группировка фаз и проходы компилятора. Стадии компилятора	2
4	Способы задания языка. Операции над языками.	2
5	Терминалы, нетерминалы, правила, выводимые цепочки. Язык, определяемый грамматикой	2

6	Алгоритм преобразования недетерминированного конечного распознавателя в детерминированный.	2
7	Алгоритм построения конечного распознавателя $M$ для заданных двух произвольных конечных распознавателей $M_1$ и $M_2$ так, чтобы $L(M) = L(M_1) \cup L(M_2)$ . Доказательство эквивалентности порождаемых языков	2
8	Алгоритм построения конечного распознавателя $M$ для заданных двух произвольных конечных распознавателей $M_1$ и $M_2$ так, чтобы $L(M) = L(M_1)L(M_2)$ . Алгоритм построения конечного распознавателя $M$ для заданного произвольного конечного распознавателя $M_1$ так, чтобы $L(M) = L(M_1)^*$ . Доказательство эквивалентности порождаемых языков	2
9	Алгоритм построения конечного распознавателя для заданной произвольной праволинейной грамматики.	2
10	Алгоритм построения праволинейной грамматики для заданного произвольного конечного распознавателя.	2
11	Алгоритм поиска непродуктивных нетерминалов для КС-грамматик. Алгоритм устранения непродуктивных нетерминалов для КС-грамматик.	2
12	Алгоритм поиска недостижимых символов для КС-грамматик. Алгоритм устранения недостижимых символов для КС-грамматик.	2
13	Алгоритм устранения бесполезных символов для КС-грамматик. Алгоритм построения множества аннулирующих нетерминалов для КС-грамматики. Алгоритм преобразования КС-грамматики к неукорачивающему виду.	2
14	Алгоритм устранения цепных правил для КС-грамматик. Алгоритм устранения непосредственной левой рекурсии для КС-грамматик.	2
15	Алгоритм устранения левой рекурсии в общем виде для КС-грамматик. Алгоритм левой факторизации для КС-грамматик	2
16	Алгоритм построения $MPP_e$ -распознавателя для заданного произвольного $MPP_e$ -распознавателя.	2
17	Алгоритм построения $MPP_e$ -распознавателя для заданного произвольного $MPP$ -распознавателя.	2
18	Алгоритм построения $MPP_e$ -распознавателя для заданной произвольной КС-грамматики. Алгоритм построения КС-грамматики для заданного произвольного $MPP_e$ -распознавателя.	2
<b>Всего</b>		<b>36</b>



## 6. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ПО ОРГАНИЗАЦИИ САМОСТОЯТЕЛЬНОЙ РАБОТЫ СТУДЕНТОВ

№ п/п	Название темы	Количество часов
1	Интерпретация семантического терма программы в форме семантического дерева.	2
2	Использование таблиц для накопления информации о структуре и типе переменных в программе.	2
3	Анализ ошибок. Генератор объектного кода. Группировка фаз и проходы компилятора. Стадии компилятора	3
4	Способы задания языка. Операции над языками.	3
5	Терминалы, нетерминалы, правила, выводимые цепочки. Язык, определяемый грамматикой	3
6	Алгоритм преобразования недетерминированного конечного распознавателя в детерминированный.	3
7	Алгоритм построения конечного распознавателя $M$ для заданных двух произвольных конечных распознавателей $M_1$ и $M_2$ так, чтобы $L(M) = L(M_1) \cup L(M_2)$ . Доказательство эквивалентности порождаемых языков	3
8	Алгоритм построения конечного распознавателя $M$ для заданных двух произвольных конечных распознавателей $M_1$ и $M_2$ так, чтобы $L(M) = L(M_1)L(M_2)$ . Алгоритм построения конечного распознавателя $M$ для заданного произвольного конечного распознавателя $M_1$ так, чтобы $L(M) = L(M_1)^*$ . Доказательство эквивалентности порождаемых языков	3
9	Алгоритм построения конечного распознавателя для заданной произвольной праволинейной грамматики.	3
10	Алгоритм построения праволинейной грамматики для заданного произвольного конечного распознавателя.	3
11	Алгоритм поиска непродуктивных нетерминалов для КС-грамматик. Алгоритм устранения непродуктивных нетерминалов для КС-грамматик. Алгоритм поиска недостижимых символов для КС-грамматик. Алгоритм устранения недостижимых символов для КС-грамматик.	3
12	Алгоритм устранения бесполезных символов для КС-грамматик. Алгоритм построения множества аннулирующих нетерминалов для КС-грамматики. Алгоритм преобразования КС-грамматики к неукорачивающему виду.	3
13	Алгоритм устранения цепных правил для КС-грамматик. Алгоритм устранения непосредственной левой рекурсии для КС-грамматик.	2
14	Алгоритм устранения левой рекурсии в общем виде для КС-грамматик. Алгоритм левой факторизации для КС-грамматик	2
15	Алгоритм построения $MPI$ -распознавателя для заданного произвольного $MPI_e$ -распознавателя.	2

16	Алгоритм построения $МП_e$ -распознавателя для заданного произвольного $МП$ -распознавателя	2
17	Алгоритм построения $МП_e$ -распознавателя для заданной произвольной КС-грамматики.	2
18	Алгоритм построения КС-грамматики для заданного произвольного $МП_e$ -распознавателя	2
19	Методы обработки языка. Нисходящий анализ. S-грамматики. Управляющая таблица разбора. Алгоритм построения управляющей таблицы разбора	2
20	Предиктивный синтаксический анализатор. Алгоритм построения предиктивного синтаксического анализатора для $LL(1)$ -грамматик. Пример построения предиктивного синтаксического анализатора	2
21	Синтаксически управляемая трансляция. Синтаксис выражений. Проблемы неоднозначности грамматик. Ассоциативность и приоритет операций. Синтаксически управляемые определения. Синтаксически управляемая трансляция.	2
22	Информационная модель синтаксически управляемой трансляции. Практическая реализация синтаксически управляемой трансляции	2
<b>Всего</b>		<b>54</b>

Полный перечень 40 вариантов для каждого из четырех индивидуальных заданий, выполнение которых предусмотрено в семестре, с подробными примерами их решения приведены в [1].

## 7. ИНДИВИДУАЛЬНЫЕ ЗАДАНИЯ

Не предусмотрены

## 8. КОНТРОЛЬНЫЕ ВОПРОСЫ К ЭКЗАМЕНУ И МОДУЛЬНОМУ КОНТРОЛЮ

Контрольные вопросы в форме тестовых заданий закрытого типа приведены в [1].

1. Основные фазы языкового процессора: лексический анализатор; синтаксический анализатор; семантический анализатор; оптимизатор кода.
2. Интерпретация семантического терма программы в форме семантического дерева.
3. Критерии оптимизации семантического терма программы. Машинно-независимые и машинно-зависимые методы оптимизации.
4. Использование таблиц для накопления информации о структуре и типе переменных в программе.
5. Генератор объектного кода. Группировка фаз и проходы компилятора. Стадии компилятора
6. Языки. Способы задания языка. Операции над языками.
7. Формальные грамматики. Терминалы, нетерминалы, правила, выводимые цепочки.
8. Язык, определяемый грамматикой
9. Форма Бэкуса-Наура.
10. Грамматики с ограничениями на правила. Классификация грамматик по Хомскому

11. Классификация абстрактных распознавателей. Машина Тьюринга.
12. Схема абстрактного распознавателя. Конфигурации и такты. Допускаемые и не допускаемые цепочки.
13. Языки, определяемые (расознаваемые, допускаемые) абстрактным распознавателем. Детерминированные и недетерминированные конечные распознаватели
14. Алгоритм преобразования недетерминированного конечного распознавателя в детерминированный.
15. Детерминированные и недетерминированные распознаватели с магазинной памятью
16. Линейно ограниченные распознаватели.
17. Классы эквивалентности формальных грамматик и абстрактных распознавателей
18. Регулярные множества и регулярные выражения. Свойства регулярных выражений. Регулярные выражения и праволинейные грамматики.
19. Уравнения с регулярными коэффициентами. Системы уравнений с регулярными коэффициентами. Методы построения решений.
20. Алгоритм построения регулярного выражения для заданной произвольной праволинейной грамматики
21. Конечные распознаватели и регулярные выражения. Алгоритм построения конечного распознавателя  $M$  для заданных двух произвольных конечных распознавателей  $M_1$  и  $M_2$  так, чтобы  $L(M) = L(M_1) \cup L(M_2)$ . Доказательство эквивалентности порождаемых языков.
22. Алгоритм построения конечного распознавателя  $M$  для заданных двух произвольных конечных распознавателей  $M_1$  и  $M_2$  так, чтобы  $L(M) = L(M_1)L(M_2)$ .
23. Алгоритм построения конечного распознавателя  $M$  для заданного произвольного конечного распознавателя  $M_1$  так, чтобы  $L(M) = L(M_1)^*$ . Доказательство эквивалентности порождаемых языков
24. Конечные распознаватели и праволинейные грамматики. Алгоритм построения конечного распознавателя для заданной произвольной праволинейной грамматики.
25. Алгоритм построения праволинейной грамматики для заданного произвольного конечного распознавателя. Доказательство эквивалентности порождаемых языков
26. Распознающие и обрабатывающие автоматы. Проблемы идентификации. Расширяющиеся конечные автоматы
27. Транслитераторы. Идентификация слов (метод автомата). Идентификация слов с нечёткими границами. Задача обнаружения префиксов
28. КС-грамматики. Выводы. Деревья вывода.
29. Левосторонняя и правосторонняя стратегии вывода. Однозначность грамматики. Приведение грамматик.
30. Распознаватели с магазинной памятью, допускающим цепочку опустошением магазина. Такты и конфигурации  $МП_e$ -распознавателя.
31. Методы обработки языка. Нисходящий анализ. S-грамматики. Управляющая таблица разбора. Алгоритм построения управляющей таблицы разбора
32. Предиктивный синтаксический анализатор.
33. Алгоритм построения предиктивного синтаксического анализатора для  $LL(1)$ -грамматик. Пример построения предиктивного синтаксического анализатора
34. Синтаксически управляемая трансляция. Синтаксис выражений. Проблемы неоднозначности грамматик.
35. Ассоциативность и приоритет операций. Синтаксически управляемые определения. Синтаксически управляемая трансляция.
36. Информационная модель синтаксически управляемой трансляции. Практическая реализация синтаксически управляемой трансляции

## 9. ОБРАЗЕЦ МОДУЛЬНОЙ КОНТРОЛЬНОЙ РАБОТЫ ГОУ ВПО «ДОНЕЦКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ»

Факультет математики и информационных технологий

*Направление подготовки:* **01.03.02 Прикладная математика и информатика**  
*Профиль:* **Статистика**  
*Программа подготовки:* **бакалавриат**  
*Семестр* **5**  
*Учебная дисциплина* **Теория автоматов и формальных языков**

### МОДУЛЬНАЯ КОНТРОЛЬНАЯ РАБОТА ВАРИАНТ №1

1. Интерпретация семантического терма программы в форме семантического дерева.
2. Алгоритм построения конечного распознавателя для заданной произвольной праволинейной грамматики.
3. Управляющая таблица разбора. Алгоритм построения управляющей таблицы разбора.
4. Информационная модель синтаксически управляемой трансляции.

Утверждено на заседании кафедры ТУ и ВМ  
протокол № \_\_\_\_ от «\_\_» \_\_\_\_\_ 20\_\_ г.

Заведующий кафедрой  
Преподаватель

#### Критерии оценивания модульного контроля

<i>Номер задания</i>	<i>Количество баллов</i>
1	10
2	10
3	10
4	10
<b>Всего</b>	<b>40</b>

## 10. ОБРАЗЕЦ ЭКЗАМЕНАЦИОННОГО БИЛЕТА

ГОУ ВПО «ДОНЕЦКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ»  
факультет математики и информационных технологий

Образовательный уровень: Бакалавр  
 Направления подготовки: 01.03.02 Прикладная математика и информатика  
 Профиль подготовки: Статистика  
 Семестр: 5  
 Учебная дисциплина: «Теория автоматов и формальных языков»

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № \_\_

- 1 Теоретическое задание открытого типа: **Алгоритм** построения конечного распознавателя  $M$  для заданного произвольного конечного распознавателя  $M_1$  так, чтобы  $L(M) = L(M_1)^*$ . Доказательство эквивалентности порождаемых языков.  
**Пример.**
- 2 Теоретико-практические задания закрытого типа: Вариант задания ККР (40 тестовых заданий закрытого типа).

Утверждено на заседании кафедры теории  
упругости и вычислительной математики  
Протокол № \_\_\_\_ от „\_\_” \_\_\_\_\_ 20\_\_ года  
Заведующий \_\_\_\_\_  
Экзаменатор \_\_\_\_\_

## 11. ОБРАЗЕЦ ТЕСТОВОГО ЗАДАНИЯ ЗАКРЫТОГО ТИПА.

Построить грамматику эквивалентную  $G = (\{S, A, B, C\}, \{a, b, c, d, e\}, P, S)$  без цепных правил.  
 $P = \{S \rightarrow cAdB|dBA|C; A \rightarrow bcA|d; B \rightarrow caB|bC; C \rightarrow eA|dB|A\}$

а)  $P = \{S \rightarrow cAdB|dBA|eA; A \rightarrow bcA|d; B \rightarrow caB|bC; C \rightarrow eA|dB|bcA\}$

б)  $P = \{S \rightarrow cAdB|dBA|eA|dB|bcA|d; A \rightarrow bcA|d; B \rightarrow caB|bC; C \rightarrow eA|dB|bcA|d\}$

в)  $P = \{S \rightarrow cAdB|dBA|bcA; A \rightarrow bcA|d; B \rightarrow caB|bC; C \rightarrow bcA|d\}$

г)  $P = \{S \rightarrow cAdB|dBA|eA; A \rightarrow bcA|d; B \rightarrow caB|bC; C \rightarrow eA|dB|d\}$

## 12. КРИТЕРИИ ОЦЕНИВАНИЯ.

Формы контроля по курсу:

- Теоретический опрос по четырем содержательным модулям (СМ) и защита четырех индивидуальных заданий (ИЗ) – промежуточный контроль;
- проверка теоретических знаний и практических умений в режиме тестового модульного контроля по заданиям комплексной контрольной работы (ККР);
- письменный экзамен.

## РАСПРЕДЕЛЕНИЕ БАЛЛОВ

МОДУЛЬ 1				МОДУЛЬ 2				ВСЕГО		
Промежуточный контроль				МКР	Промежуточный контроль				МКР	
СМ 1.1		СМ 1.2		Тест	СМ 2.1		СМ 2.2		ККР	
12		18		0	15		15		40	100
ТО	ЗП	ТО	ЗП		ТО	ЗП	ТО	ЗП		
12	0	14	4		11	4	11	4		

Сокращения: ТО – теоретический опрос; ЗП – защита программы; МКР – модульная контрольная работа.

Экзамен не является обязательной формой контроля знаний студента. Экзамен есть форма повышения общей оценки, заработанной студентом в течение семестра как результат продемонстрированного студентом на экзамене более высокого уровня теоретических знаний и практических умений. Максимальное количество баллов, на которое студент может повысить свою оценку по итогам экзамена составляет 60 баллов. При этом уровень знаний и умений, продемонстрированный студентом по итогам промежуточных и модульных контролей в течение семестра повторно на экзамене не оценивается, окончательная оценка за курс с учетом экзамена не может превосходить 100 баллов.

Экзаменационный билет письменного экзамена включает одно теоретическое задание и 40 теоретико-практических тестовых заданий закрытого типа (ККР):

- выбран правильный ответ - **1 балл**;
- выбран неправильный ответ - **0 баллов**.

одно теоретическое задание открытого типа	40 теоретико-практических тестовых заданий закрытого типа	ВСЕГО
20	40	60

**Шкала соответствия баллов национальной шкале**

Оценка по шкале ECTS	Оценка по 100-балльной шкале	Оценка по государственной шкале (экзамен, дифференцированный зачет)	Оценка по государственной шкале (зачет)
<b>A</b>	90-100	5 (отлично)	зачтено
<b>B</b>	80-89	4 (хорошо)	зачтено
<b>C</b>	75-79	4 (хорошо)	зачтено
<b>D</b>	70-74	3 (удовлетворительно)	зачтено
<b>E</b>	60-69	3 (удовлетворительно)	зачтено
<b>FX</b>	35-59	2 (неудовлетворительно) с возможностью повторной сдачи	не зачтено
<b>F</b>	0-34	2 (неудовлетворительно) с возможностью повторной сдачи при условии обязательного набора дополнительных баллов	не зачтено

### 13. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ УЧЕБНОГО ПРОЦЕССА.

Лекционные занятия проводятся в аудитории, оснащенной доской. Лабораторные занятия проводятся в специализированном компьютерном классе, оборудованном компьютерами с лицензионным программным обеспечением, доступом к сети Интернет, столами, доской.

### 14. РЕКОМЕНДОВАННАЯ ЛИТЕРАТУРА.

№ п/п	Наименование	Кол-во экземпляров в библиотеке ДонНУ	Наличие электронной версии в ЭБС
<b>Основная литература</b>			
1.	Авдюшина О.В. Системне програмування. Формальні граматики та методи синтаксичного аналізу: навчальний посібник (Рекомендовано МОН України як навчальний посібник для студентів ВНЗ). / Авдюшина О.В., Дзундза А.І., Моїсеєнко І.О., Нескородеєв Р.М.. – Донецьк: ДонНУ, 2011. – 188 с.	61	да
2.	Ахо, Альфред В. Компиляторы: Принципы, технологии, инструменты / Альфред Ахо, Рави Сети, Джеффри Ульман ; [Пер. с англ. И. В. Красикова]. - М. и др.: Вильямс, 2003. - 767 с.	5	нет
3.	Ахо, Альфред В. Теория синтаксического анализа, перевода и компиляции. Т. 1 : Синтаксический анализ / А. В. Ахо, Д. Д. Ульман ; Пер. с англ. В. Н.	6	нет

	Агафонова ; Под ред. В. М. Курочкина. - М. : Мир, 1978. - 612 с.		
4.	Ахо, Альфред В. Теория синтаксического анализа, перевода и компиляции Т. 2 : Компиляция / А. В. Ахо, Д. Д. Ульман ; Пер. с англ. А. Н. Бирюкова, В. А. Серебрякова ; Под ред. В. М. Курочкина. - М. : Мир, 1978. - 487 с.	6	нет
5.	Калоеров С. А. Программирование на языке С++ : учеб. пособие / С. А. Калоеров ; Донецкий нац. ун-т. - Изд. 3-е. - Донецк : Юго-Восток, 2009. - 298 с.	85	да
6.	Павловская Т. А. С/С++. Программирование на языке высокого уровня : учеб. для студентов вузов, обучающихся по направлению "Информатика и вычислит. техника" / Т. А. Павловская. - Москва [и др.] : Питер, 2010. - 460 с.	32	нет
<b>Дополнительная литература</b>			
7.	Методические указания и задания к практическим и лабораторным занятиям по программированию для студентов специальности «Прикладная математика» / Сост. С.А.Калоеров, Е.В.Авдюшина, Л.А.Нестерова, Л.Н.Шкодина. – Донецк: ДонНУ, 2004. – 92с.	15	да
8.	Учебно-методическое пособие к изучению курса «Системное программирование» для студентов специальности 080202 Прикладная математика / Сост.: И.А.Моисеенко, Л.А.Нестерова, Е.В.Авдюшина, Р.Н.Нескородев, О.Д.Фесенко. – Донецк: ДонНУ, 2006. – 150 с.	65	да
9.	Задания для занятий по программированию на языке С++ / Сост.: С.А.Калоеров, Е.В.Авдюшина, А.И.Ануфриева, Л.Н.Шкодина, А.В.Петренко. – Донецк: Юго-Восток, 2010. – 96с.	18	да

## 15. ИНФОРМАЦИОННЫЕ РЕСУРСЫ.

1. [http://msdn.microsoft.com/ru-ru/library/z1zx9t92\(v=vs.90\).aspx](http://msdn.microsoft.com/ru-ru/library/z1zx9t92(v=vs.90).aspx)
2. Портал математических интернет-ресурсов - <http://www.math.ru/>
3. Портал математических интернет-ресурсов - <http://www.exponenta.ru>
4. Портал математических интернет-ресурсов - <http://www.allmath.com/>
5. Портал ресурсов по математике, алгоритмике и ИТ - <http://algolist.manual.ru>
6. <http://math.sgu.ru/sites/chairs/prinf/materials/java/index.htm>
7. <http://www.eclipse.org/eclipse/>

## 16. ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ.

1. Трансляторы языка С++

Рабочая программа рассмотрена и переутверждена на заседании кафедры ТУ и ВМ  
с изменениями (без изменений) на 20 \_\_\_\_\_ год.

Протокол № \_\_\_\_ от «\_\_\_\_» \_\_\_\_\_ 20\_\_\_\_ г.

Заведующий кафедрой

\_\_\_\_\_