



МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение высшего образования
«Дальневосточный федеральный университет»
(ДВФУ)

ШКОЛА ЕСТЕСТВЕННЫХ НАУК

«СОГЛАСОВАНО»
Руководитель ОП

И.Л. Артемьева

«УТВЕРЖДАЮ»
Заведующая кафедрой прикладной
математики, механики, управления
и программного обеспечения

И.Л. Артемьева



« 10 » июля 2019 г.

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ

Теория вычислительных процессов и структур

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

(Технология программирования)

Форма подготовки очная

курс 3, 4 семестр 5, 6, 7

лекции 36 час.

практические занятия 36 час.

лабораторные работы 36 час.

в том числе с использованием МАО лек. 0 /пр. 0 /лаб. 0 час.

в том числе в электронной форме лек. 0 /пр. 0 /лаб. 0 час.

всего часов аудиторной нагрузки 108 час.

в том числе с использованием МАО 0 час.

в том числе в электронной форме 0 час.

самостоятельная работа 108 час.

в том числе на подготовку к экзамену 54 час.

курсовая работа / курсовой проект 6 семестр

зачет семестр

экзамен 5, 7 семестр

Рабочая программа составлена в соответствии с требованиями Федерального государственного образовательного стандарта высшего образования, утвержденного приказом Министерства образования и науки Российской Федерации от 23 августа 2017 г. № 809

Рабочая программа обсуждена на заседании кафедры прикладной математики, механики, управления и программного обеспечения, протокол № 7.1 от « 04 » июля 2019 г.

Заведующий кафедрой прикладной математики, механики, управления и программного обеспечения д.т.н., профессор Артемьева И.Л.


Составители: д.т.н., профессор Артемьева И.Л., ст. преподаватель Лемза А.О.

Оборотная сторона титульного листа РПУД

I. Рабочая программа пересмотрена на заседании кафедры прикладной математики, механики, управления и программного обеспечения:

Протокол от «09» июля 2021 г. № 7.1

Заведующий кафедрой _____

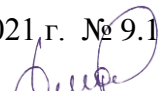

(подпись)

Артемьева И.Л.
(И.О. Фамилия)

II. Рабочая программа пересмотрена на заседании департамента программной инженерии и искусственного интеллекта:

Протокол от «17» сентября 2021 г. № 9.1

И.о. директора департамента _____


(подпись)

Смагин С.В.
(И.О. Фамилия)

ЦЕЛИ И ЗАДАЧИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Цель дисциплины – познакомить студентов с существующей теорией формальных языков и трансляций; с типами автоматов и преобразователей; с понятием трансляции, схемой и методами построения компиляторов.

Задачи:

- Изучение методов задания формального языка с использованием регулярных выражений, порождающих грамматик и распознавателей.
- Изучение методов синтаксического анализа заданного формального языка.
- Изучение методов продолжения разбора в случае возникновения ошибочной ситуации при разборе цепочки.
- Изучение типов контекстных условий языков программирования.
- Изучение методов разработки языковых процессоров.

Планируемые результаты обучения по данной дисциплине (знания, умения, владения), соотнесенные с планируемыми результатами освоения образовательной программы.

Общепрофессиональные компетенции выпускников и индикаторы их достижения

Наименование категории (группы) общепрофессиональных компетенций	Код и наименование общепрофессиональной компетенции	Код и наименование индикатора достижения общепрофессиональной компетенции
Теоретические и практические основы профессиональной деятельности	ОПК-2. Способен применять современный математический аппарат, связанный проектированием, разработкой, реализацией и оценкой качества программных продуктов и программных комплексов в различных областях человеческой деятельности	<p>ОПК-2.1. Знает: понятие языка и свойства; теоретические способы задания языков; теоретические основы информационных процессов преобразования грамматик и распознаватели.</p> <p>ОПК-2.2. Умеет применять математический аппарат для формального представления языка.</p> <p>ОПК-2.3. Имеет навыки применения данного математического аппарата при решении конкретных задач</p>

Профессиональные компетенции выпускников и индикаторы их достижения:

Задача профессиональной деятельности	Объект или область знания	Код и наименование профессиональной компетенции	Код и наименование индикатора достижения профессиональной компетенции	Основание (ПС, анализ иных требований, предъявляемых к выпускникам)
Тип задач профессиональной деятельности: научно-исследовательский				
<p>Применение фундаментальных знаний, полученных в области математических и (или) естественных наук. Создание, анализ и реализация новых компьютерных моделей в современном естествознании, технике, экономике и управлении.</p>	<p>Математические и алгоритмические модели, программы, программные системы и комплексы, методы их проектирования и реализации, способы производства, сопровождения, эксплуатации и администрирования в различных областях цифровой экономики</p>	<p>ПК-1. - Способен демонстрировать базовые знания математических и естественных наук, программирования и информационных технологий</p>	<p>ПК-1.1. знает современные методы разработки, проектирования и программирования методов трансляции ПК-1.2. Умеет выбирать, проектировать, реализовывать, оценивать качество и эффективность программного обеспечения для создания языковых процессоров ПК-1.3. владеет методами и средствами оценивания программного обеспечения.</p>	<p>06.003 Архитектор программного обеспечения; 06.004 Специалист по тестированию в области информационных технологий; 06.001 Программист</p>
Тип задач профессиональной деятельности: производственно-технологический				

<p>Разработка, отладка, проверка работоспособности и, модификация программного обеспечения. Создание и сопровождение архитектуры программных средств. Разработка и тестирование программного обеспечения. Проектирование, разработка и сопровождение компьютерных систем автоматизации производства и управления</p>	<p>Математические и алгоритмические модели, программы, программные системы и комплексы, методы их проектирования и реализации, способы производства, сопровождения, эксплуатации и администрирования в различных областях цифровой экономики.</p>	<p>ПК-5. Способен применять современные информационные технологии при проектировании, реализации, оценке качества и анализа эффективности и программного обеспечения для решения задач в различных предметных областях.</p>	<p>ПК-5.1. Знает средства разработки, проектирования и реализации компилятора языка. ПК-5.2. Умеет использовать языки программирования и специализированные средства и оболочки для создания приложений. ПК-5.3. Владеет средствами разработки, проектирования и реализации компилятора языка.</p>	<p>06.003 Архитектор программного обеспечения; 06.001 Программист;</p>
--	---	---	--	--

I. СТРУКТУРА И СОДЕРЖАНИЕ ТЕОРЕТИЧЕСКОЙ ЧАСТИ КУРСА

Тема 1 Алфавит. Цепочки (2 час.)

Длина цепочки. Пустая цепочка. Операции над цепочками. Подцепочка, суффикс и префикс цепочки. Позиция в цепочке. Вхождение подцепочки в цепочку. Языки. Конкатенация языков, операция итерации языка. Гомоморфизм, применение гомоморфизма к языку. Обращение гомоморфизма. Операция подстановки языков вместо символов.

Тема 2 Способы задания языков (2 час.)

Регулярные множества и регулярные выражения. Основные алгебраические свойства регулярных выражений. Формальные грамматики. Выводимость цепочки в грамматике. Вывод в грамматике. Полный и неполный выводы. Терминальное порождение символа. Язык, порождаемый грамматикой. Перестройка вывода. Размеченный вывод. Равносильность выводов. Эквивалентность грамматик. Классификация грамматик по Хомскому: праволинейные, левوليнейные, правоавтоматные, левоавтоматные, контекстно-свободные грамматики, грамматики непосредственных составляющих, грамматики общего вида. Связь регулярных множеств и праволинейных грамматик. Расщепление грамматик.

Тема 3 Контекстно-свободные языки и грамматики (2 час.)

Граф контекстно-свободной (КС) грамматики. Дерево вывода в КС-грамматике. Крона дерева вывода, сечение дерева вывода, крона сечения. Левый (правый) выводы, левовыводимая (правовыводимая) цепочка. Неоднозначные КС грамматики.

Тема 4 Преобразования КС-грамматик (2 час.)

Бесполезные и недостижимые символы грамматики. Цепные правила. Циклы в грамматике. Удаление бесполезных и недостижимых символов. Удаление цепных правил. Грамматики без λ -правил.

Тема 5 Леворекурсивные и праворекурсивные грамматики (2 час.)

Непосредственная левая рекурсия. Методы устранения левой рекурсии. Нормальная форма Хомского. Нормальная форма Грейбах.

Тема 6 Неоднозначные грамматики (2 час.)

Способы устранения неоднозначности. Контекстные условия языков.

Тема 7 Распознаватели, их структура и типы (2 час.)

Преобразователи, их типы. Конечные автоматы. Конфигурации конечного автомата. Недетерминированные и детерминированные конечные автоматы. Диаграмма конечного автомата. Полностью и не полностью определенный конечный автомат. Достижимые состояния конечного автомата. Язык, допускаемый конечным автоматом. Эквивалентность автоматов. Связь между конечными автоматами и праволинейными грамматиками. Канонический конечный автомат.

Тема 8 Автоматы с магазинной памятью (4 час.)

Конфигурации автомата с магазинной памятью. Язык, допускаемый автоматом с магазинной памятью. Недетерминированный и детерминированный автомат с магазинной памятью. Расширенный автомат с магазинной памятью. Связь между автоматами с магазинной памятью и КС-грамматиками.

Тема 9 Понятие трансляции. Обзор процесса трансляции (2 час.)

Формальный язык и языковой процессор на его основе. Теоретический язык и входной язык языкового процессора. Типы языковых процессоров (компиляторы, интерпретаторы, смешанные формы). Структура языкового процессора. Задачи, решаемые лексическим, синтаксическим и семантическим анализаторами. Задачи, решаемые генератором кода. Задачи блока оптимизации.

Тема 10 Разработка лексического анализатора языка (2 час.)

Реализация лексического анализа в языковом процессоре. Лексические ошибки. Работа с таблицей имен. Способы хранения длинных имен и строк. Токены, шаблоны, лексемы. Атрибуты токенов. Их вычисление при разборе и хранение. Язык спецификации лексических анализаторов.

Тема 11 Разработка синтаксического анализатора языка (6 час.)

Понятие о стратегии синтаксического анализа. Важнейшие классы грамматик и соответствующие им анализаторы. Формализмы для задания грамматик. Нисходящий и восходящий разбор. Обработка синтаксических ошибок. Методы продолжения разбора при ошибках. Способы представления результатов анализа: дерево разбора, синтаксическое дерево, польская запись. Работа с таблицей имен при синтаксическом анализе.

Тема 12 Проверка контекстных условий (2 час.)

Контекстные условия языков. Атрибутные грамматики. Задание контекстных условий как предикатов над атрибутами. Организация вычисления значений атрибутов при семантическом анализе. Организация проверки контекстных условий. Особенности семантического анализа для языков с полиморфными операциями. Преобразования типов. Область видимости имен в анализируемых программах. Работа с таблицей имен.

Тема 13 Генерация кода (2 час.)

Среды периода выполнения. Стеки управления, деревья активации. Организация памяти для объектной программы. Общая схема генерации и внутреннее представление команд. Генерация выражений и операторов присваивания. Генерация управления вычислениями. Генерация подпрограмм и обращений к ним. Распределение памяти.

Тема 14 Оптимизация кода (2 час.)

Схемы программ. Поточковый анализ программ. Оптимизирующие преобразования. Организация трансляторов. Схемы процесса трансляции и организация трансляторов: однопросмотровая схема, двухпросмотровая схема, многопросмотровые схемы. Влияние типов выходных языков на организацию трансляторов. Распараллеливающие компиляторы.

Тема 15 Системы автоматического построения трансляторов (2 час.)

Атрибутные грамматики. Алгоритмы заполнения атрибутных деревьев, атрибутные СПТ. Примеры.

II. СТРУКТУРА И СОДЕРЖАНИЕ ПРАКТИЧЕСКОЙ ЧАСТИ КУРСА

Практические занятия (18 часов, 5 семестр)

Практическое занятие № 1. Разработка внешних спецификаций «классического» компилятора (8 час.)

Практическое занятие № 2. Эскизное проектирование компилятора (4 час.)

Практическое занятие № 3. Проектирование лексического анализатора (6 час.)

Практические занятия (18 часов, 7 семестр)

Практическое занятие № 4. Проектирование генератора (4 час.)

Практическое занятие № 5. Реализация генератора (6 час.)

Практическое занятие № 6. Сборка, отладка и тестирование «классического» компилятора (4 час.)

Практическое занятие № 7. Подготовка документации по разработке компилятора и оформление отчета (4 час.)

Лабораторные работы (36 часов, 6 семестр)

Лабораторная работа № 1. Реализация лексического анализатора (6 час.)

Лабораторная работа № 2. Проектирование синтаксического анализатора (6 час.)

Лабораторная работа № 3. Реализация синтаксического анализатора (12 час.)

Лабораторная работа № 4. Реализация семантического анализатора (12 час.)

III. УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ САМОСТОЯТЕЛЬНОЙ РАБОТЫ ОБУЧАЮЩИХСЯ

Учебно-методическое обеспечение самостоятельной работы обучающихся по дисциплине «Теория вычислительных процессов и структур» представлено в Приложении 1 и включает в себя:

план-график выполнения самостоятельной работы по дисциплине, в том числе примерные нормы времени на выполнение по каждому заданию;

характеристика заданий для самостоятельной работы обучающихся и методические рекомендации по их выполнению;

требования к представлению и оформлению результатов самостоятельной работы;

критерии оценки выполнения самостоятельной работы.

IV. КОНТРОЛЬ ДОСТИЖЕНИЯ ЦЕЛЕЙ КУРСА

№ п/п	Контролируемые разделы/темы дисциплины	Коды и этапы формирования компетенций		Оценочные средства	
				текущий контроль	промежуточная аттестация
1	Тема 1. Алфавит. Цепочки.	ОПК-2	знает	ИЗ 1-4, тест	Экзамен, вопросы 1-2
	Тема 2. Способы задания языков	ОПК-2	знает	ИЗ 5-10, тест	Экзамен, вопросы 3-6
	Тема 3, 4. Контекстно-свободные языки и грамматики. Преобразования КС – грамматик	ОПК-2 ПК-1	знает	ИЗ 11-12, тест	Экзамен, вопросы 7-17
	Тема 5. Леворекурсивные и праворекурсивные грамматики.	ПК-1	знает	ИЗ 13-15, тест	Экзамен, вопросы 18-21
	Тема 6, 7, 8. Неоднозначные грамматики. Распознаватели, их структура и типы. Автоматы с магазинной памятью	ОПК-2 ПК-1	знает	ИЗ 16-19, тест	Экзамен, вопросы 22-29
	Тема 9. Понятие трансляции. Обзор процесса трансляции.	ПК-1	знает	Тест	Экзамен, вопрос 30
2	Тема 10. Разработка лексического анализатора языка	ПК-1 ПК-5	знает	Блиц-опрос	Экзамен, вопросы 31-37

	Тема 11. Разработка синтаксического анализатора языка	ПК-1 ПК-5	знает	Задания 1-6	Экзамен, вопросы 38-43
	Тема 12, 13. Проверка контекстных условий. Генерация кода	ПК-5	знает	Задание 7	Экзамен, вопросы 44-50
3	Тема 14, 15. Оптимизация кода. Системы автоматического построения трансляторов	ПК-5	знает	Блиц-опрос	Экзамен, вопросы 51-55
4	Практическое занятие № 1. Разработка внешних спецификаций «классического» компилятора	ОПК-2	умеет владеет	Неформальная постановка задачи, грамматика входного языка, требования к ПО	Курсовой проект
	Практическое занятие № 2. Эскизное проектирование компилятора	ПК-1 ПК-5	умеет владеет	АКД, проект верхнего уровня	Курсовой проект
	Практическое занятие № 3. Проектирование лексического анализатора	ОПК-2 ПК-1	умеет владеет	Проект лексического анализатора, набор тестов	Курсовой проект
	Лабораторная работа № 1. Реализация лексического анализатора	ПК-1 ПК-5	умеет владеет	Работающее программное средство	Курсовой проект
	Лабораторная работа № 2. Проектирование синтаксического анализатора	ОПК-2 ПК-1	умеет владеет	Проект синтаксического анализатора	Курсовой проект
	Лабораторная работа № 3. Реализация синтаксического анализатора	ПК-1 ПК-5	умеет владеет	Работающее программное средство	Курсовой проект

5	Лабораторная работа № 4. Реализация семантического анализатора.	ОПК-2 ПК-5	умеет владеет	Работающее программное средство	Курсовой проект
	Практическое занятие № 4. Проектирование генератора.	ПК-4, ОПК-11	умеет владеет	Проект генератора кода	Курсовой проект
6	Практическое занятие № 5. Реализация генератора	ПК-1 ПК-5	умеет владеет	Работающее программное средство	Курсовой проект
	Практическое занятие № 6. Сборка, отладка и тестирование «классического» компилятора	ПК-1 ПК-5	умеет владеет	Работающее программное средство	Курсовой проект
	Практическое занятие № 7. Подготовка документации по разработке компилятора и оформление отчета	ПК-5	умеет владеет	Отчет по разработке компилятора	Курсовой проект

Типовые контрольные задания, методические материалы, определяющие процедуры оценивания знаний, умений и навыков и (или) опыта деятельности, а также критерии и показатели, необходимые для оценки знаний, умений, навыков и характеризующие этапы формирования компетенций в процессе освоения образовательной программы, представлены в Приложении 2.

V. СПИСОК УЧЕБНОЙ ЛИТЕРАТУРЫ И ИНФОРМАЦИОННО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Основная литература

1. Артемьева И. Л. Теория вычислительных процессов и структур. Часть 1. Языки и способы их задания. Учебно-методическое пособие. – Владивосток : Издательство Дальневосточного федерального университета, 2011. – 60 с.
2. Артемьева И.Л. Теория вычислительных процессов и структур. Часть 2. Методы синтаксического анализа программ. Учебно-методическое

пособие. – Владивосток : Издательство Дальневосточного федерального университета, 2011. – 56 с.

3. Ахо, А. Теория синтаксического анализа, перевода и компиляции. Синтаксический анализ / А. Ахо, Дж. Ульман. – М. : Книга по требованию, 2012. – 613 с.

4. Ахо, А. Компиляторы. Принципы, технологии и инструментарий / А. Ахо, Р. Сети, Дж. Ульман. – М. : Издательский дом «Вильямс», 2015. – 768 с.

5. Серебряков, В. А. Теория и реализация языков программирования: учебное пособие / В. А. Серебряков, М. П. Галочкин, Д. Р. Гончар и др. – М. : МЗ Пресс, 2012. – 352 с.

<http://lib.dvfu.ru:8080/lib/item?id=chamo:704386&theme=FEFU>

6. Котляров, В. П. Основы тестирования программного обеспечения: учебное пособие / В. П. Котляров, Т. В. Коликова. – М. : Интернет-Университет Информационных Технологий: БИНОМ. Лаборатория знаний, 2012. – 285 с. <http://lib.dvfu.ru:8080/lib/item?id=chamo:668103&theme=FEFU>

7. Вирт, Н. Построение компиляторов / Н. Вирт // Пер. с англ. Е. В. Борисов, Л. Н. Чернышов. – М. : ДМК Пресс, 2010. – 192 с.

<http://www.studentlibrary.ru/book/ISBN9785940745853.html>

8. Гагарина, Л. Г. Введение в теорию алгоритмических языков и компиляторов: учеб. пособие / Л. Г. Гагарина, Е. В. Кокорева. – М. : ИД ФОРУМ, 2011. – 176 с. <http://znanium.com/catalog/product/265617>

Дополнительная литература

1. Липаев, В. В. Программная инженерия: методологические основы: учебник для вузов / В. В. Липаев. – М. : ТЕИС, 2006. – 606 с.

<http://lib.dvfu.ru:8080/lib/item?id=chamo:248067&theme=FEFU>

2. Ахо, А. Компиляторы. Принципы, технологии и инструментарий / А. Ахо, Р. Сети, Дж. Ульман. – М. : Издательский дом «Вильямс», 2001. – 767 с.

<https://lib.dvfu.ru:8443/lib/item?id=chamo:334968&theme=FEFU>

3. Ахо, А. Теория синтаксического анализа, перевода и компиляции. Компиляция / А. Ахо, Дж. Ульман. – М. : Мир, 1978. – 487 с.

<http://lib.dvfu.ru:8080/lib/item?id=chamo:295051&theme=FEFU>

4. Пентус, А. Е. Математическая теория формальных языков: Учебное пособие / А. Е. Пентус, М. Р. Пентус. – М. : БИНОМ. Лаборатория знаний. Интернет-университет информационных технологий, 2006. – 248 с.

<http://www.iprbookshop.ru/22411.html>

5. Ахо, А. Теория синтаксического анализа, перевода и компиляции. Синтаксический анализ / А. Ахо, Дж. Ульман. – М. : Мир, 1978. – 612 с.

<https://lib.dvfu.ru:8443/lib/item?id=chamo:295041&theme=FEFU>

Перечень ресурсов информационно-телекоммуникационной сети «Интернет»

1. <https://e.lanbook.com/book/62703> Федосеева, Л. И. Основы теории автоматов и формальных языков: учебное пособие / Л. И. Федосеева, Р. М. Адиллов, М. Н. Шмоткин. – Пенза : Издательство ПензГТУ, 2013. – 136 с.

2. <https://e.lanbook.com/book/66125> Залогова, Л. А. Разработка Паскаль-компилятора / Л. А. Залогова. – М. : Лаборатория знаний, 2014. – 186 с.

3. <https://e.lanbook.com/book/39992> Вирт, Н. Разработка операционной системы и компилятора. Проект Оберон / Н. Вирт, Ю. Гуткнехт. – М. : ДМК Пресс, 2012. – 560 с.

4. <http://www.intuit.ru/studies/courses/26/26/info> Разработка компиляторов.

5. <http://citforum.ru/programming/theory/serebryakov> Серебряков В. А., Галочкин М. П. Основы конструирования компиляторов.

Перечень информационных технологий и программного обеспечения

При осуществлении образовательного процесса студентами и профессорско-преподавательским составом используется следующее программное обеспечение:

1. Microsoft Office (PowerPoint, Word и Visio).
2. Open Office.
3. Skype.

При осуществлении образовательного процесса студентами и профессорско-преподавательским составом используются следующие информационно-справочные системы:

1. Научная электронная библиотека eLIBRARY.
2. Электронно-библиотечная система издательства «Лань».
3. Электронная библиотека "Консультант студента".
4. Электронно-библиотечная система IPRbooks.
5. Информационная система "ЕДИНОЕ ОКНО доступа к образовательным ресурсам".
6. Доступ к электронному заказу книг в библиотеке ДВФУ, доступ к нормативным документам ДВФУ, расписанию, рассылке писем.

Лекции проводятся с использованием проектора и мультимедийного комплекса для проведения лекций внутренней системы портала ДВФУ. Лабораторные занятия проводятся в специализированном компьютерном классе.

VI. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

Дисциплина изучается в следующих организационных формах: лекции, лабораторное занятие; курсовая работа, самостоятельное изучение теоретического материала; самостоятельное выполнение индивидуального задания; индивидуальные консультации.

Работа на лекции

Слушание и запись лекций – сложный вид вузовской аудиторной работы. Внимательное слушание и конспектирование лекций предполагает интенсивную умственную деятельность студента. Краткие записи лекций, их конспектирование помогает усвоить учебный материал. Конспект является полезным тогда, когда записано самое существенное, основное и сделано это самим студентом. Запись лекций рекомендуется вести по возможности собственными формулировками. Желательно запись осуществлять на одной странице, а следующую оставлять для проработки учебного материала самостоятельно в домашних условиях. Конспект лекции лучше подразделять на пункты, параграфы, соблюдая красную строку. Этому в большой степени будут способствовать пункты плана лекции, предложенные преподавателям. Принципиальные места, определения, формулы и другое следует сопровождать замечаниями «важно», «особо важно», «хорошо запомнить» и т.п. Можно делать это и с помощью разноцветных маркеров или ручек. Лучше если они будут собственными, чтобы не приходилось просить их у однокурсников и тем самым не отвлекать их во время лекции. Целесообразно разработать собственную «маркографию» (значки, символы), сокращения слов. Не лишним будет и изучение основ стенографии. Работая над конспектом лекций, всегда необходимо использовать не только учебник, но и ту литературу, которую дополнительно рекомендовал лектор. Именно такая серьезная, кропотливая работа с лекционным материалом позволит глубоко овладеть знаниями.

К прослушиванию лекции следует готовиться, для этого необходимо знать программу курса и рекомендованную литературу. Тогда в процессе лекции легче отделить главное от второстепенного, легче сориентироваться:

что записать, что самостоятельно проработать, что является трудным для понимания, а что легко усвоить.

Лабораторные работы и практические занятия

В результате выполнения лабораторных работ и практических занятий студент должен изучить основные возможности современных программных средств информационных технологий, позволяющих разрабатывать и создавать программные средства. Каждый этап работы должен быть задокументирован и отражен в соответствующей главе документации.

Результатом лабораторной работы и практического занятия является созданный документ, который демонстрируется преподавателю в конце работы, либо работающее программное средство. Студент должен уметь отвечать на вопросы преподавателя, поясняя процесс создания кода для возможности тестирования по «белому ящику».

Самостоятельная работа студента

Основной формой самостоятельной работы студента является изучение теоретического материала, его дополнение рекомендованной литературой, выполнение индивидуальных заданий, а также активная работа на лабораторных занятиях.

Целью выполнения индивидуальных заданий является

- проверка знаний теоретического лекционного материала
- закрепление практических навыков, полученных в процессе выполнения лабораторной работы.

Индивидуальное задание включает в себя описание той работы, которую необходимо проделать студенту. Примеры индивидуальных заданий приведены в приложении 2.

Контроль за выполнением самостоятельной работы студента производится в виде контроля каждого этапа работы (см. приложение 1). Студент должен планировать график самостоятельной работы по дисциплине и придерживаться его.

VII. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Лекции проводятся с использованием проектора и внутренней системы портала ДВФУ. Лабораторные занятия проходят в аудиториях, оборудованных компьютерами типа Lenovo C360G-i34164G500UDK с лицензионными программами MicrosoftOffice 2013 и аудиовизуальными средствами проектор Panasonic DLPProjectorPT-D2110XE, плазма LG

FLATRON M4716CCBAM4716CJ. Для выполнения самостоятельной работы студенты в жилых корпусах ДВФУ обеспечены Wi-Fi.

VIII. УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ САМОСТОЯТЕЛЬНОЙ РАБОТЫ ОБУЧАЮЩИХСЯ

План-график выполнения самостоятельной работы по дисциплине

№ п/п	Дата/сроки выполнения	Вид самостоятельной работы	Примерные нормы времени на выполнение	Форма контроля
1.	Индивидуальные домашние задания по теме: Алфавит. Цепочки.	1-2 неделя обучения	2 часа	Проверка заданий
2.	Индивидуальные домашние задания по теме: Способы задания языков	3-4 неделя обучения	4 часа	Проверка заданий
3.	Индивидуальные домашние задания по теме: Контекстно-свободные языки и грамматики	5-6 неделя обучения	4 часа	Проверка заданий
4.	Индивидуальные домашние задания по теме: Преобразования КС – грамматик	7-9 неделя обучения	6 часов	Проверка заданий
5.	Индивидуальные домашние задания по теме: Леворекурсивные и праворекурсивные грамматики	10-11 неделя обучения	6 часов	Проверка заданий
6.	Индивидуальное домашнее задание по теме: Неоднозначные грамматики	12 неделя обучения	2 часа	Проверка заданий
7.	Индивидуальные домашние задания по теме: Распознаватели, их структура и типы	13-14 недели обучения	5 часов	Проверка заданий
8.	Индивидуальные домашние задания по теме: Автоматы с магазинной памятью	15-17 недели обучения	5 часов	Проверка заданий
9.	Работа с литературой, подготовка к тесту	18 неделя обучения	2	Тест
	ВСЕГО		36 часов	

Рекомендации по самостоятельной работе студентов

Самостоятельная работа студентов состоит в работе с литературой, подготовке к лабораторным работам и выполнении индивидуальных домашних заданий по темам.

Работа с литературой

В процессе подготовки к тесту студентам необходимо обратить особое внимание на самостоятельное изучение рекомендованной учебно-методической литературы. Самостоятельная работа с учебниками, учебными пособиями, научной, справочной литературой, материалами Интернета является наиболее эффективным методом получения знаний, позволяет значительно активизировать процесс овладения информацией, способствует более глубокому усвоению изучаемого материала, формирует у студентов свое отношение к конкретной проблеме.

Подготовка к лабораторным работам и практическим занятиям

Подготовку к каждой лабораторной работе и практическому занятию каждый студент должен начать с изучения теоретического материала и ознакомления с планом, который отражает содержание предложенной темы. Все новые понятия по изучаемой теме необходимо выучить наизусть и внести в глоссарий, который целесообразно вести с самого начала изучения курса. Результат такой работы должен проявиться в способности студента свободно ответить на теоретические вопросы по теме задания, правильном выполнении лабораторной работы.

Работа на лабораторных и практических занятиях ведется в малых группах 3-4 человека в каждой. В ходе выполнения лабораторной работы студенты учатся работе в группе, распределению ресурсов и установлению взаимодействия в коллективной работе. В процессе выполнения лабораторной работы студенты должны создать требуемый документ с помощью предлагаемого программного средства или выполнить кодирование в соответствии с заданием.

Индивидуальные задания

По теме лекции студентам выдаются индивидуальные задания, предназначенные для закрепления знаний, полученных на лекции. Задания выполняются самостоятельно. Выполненные задания сдаются преподавателю на проверку и возвращаются им на доработку, при совершении ошибок. Задание должно быть выполнено правильно (возможно за несколько

итераций). Структура индивидуального задания определяется темой лекции.

Пример индивидуального задания по теме «Алфавит. Цепочки».

Описать язык, являющийся результатом применения указанного гомоморфизма h к заданному языку L .

Вариант 1. $L = \{a^n b^{2^n} a^n \mid n \geq 1\}$, $h(a)=1$ и $h(bb)=2$

Вариант 2. $L = \{0^n 1^n 2^n \mid n \geq 1\}$, $h(0)=b$, $h(1)=a$, $h(2)=cc$

Вариант 3. $L = \{2^n 1^{2^n} 0^n \mid n \geq 1\}$, $h(0)=b$, $h(11)=c$, $h(2)=bb$

Вариант 4. $L = \{a^n b^n c^n \mid n \geq 1\}$, $h(a)=11$, $h(b)=2$, $h(c)=3$

Вариант 5. $L = \{3^n 2^n 1^n 2^n \mid n \geq 1\}$, $h(1)=aa$, $h(2)=c$, $h(3) = bb$

Вариант 6. $L = \{4^n 1^{2^n} 2^n \mid n \geq 1\}$, $h(11)=a$, $h(4)=c$, $h(2) = b$

Вариант 7. $L = \{1^n 3^{2^n} 1^n 2^n \mid n \geq 1\}$, $h(1)=aa$, $h(2)=c$, $h(3) = b$

Вариант 8. $L = \{a^n b^{2^n} c^n \mid n \geq 1\}$, $h(a)=22$, $h(b)=1$, $h(c) = 33$

Вариант 9. $L = \{a^n b^{2^n} c^{2^n} \mid n \geq 1\}$, $h(a)=3$, $h(bb)=4$, $h(c) = 1$

Вариант 10. $L = \{1^n 2^n 1^{2^n} \mid n \geq 1\}$, $h(1)=aa$, $h(2)=bb$

Вариант 11. $L = \{2^n 1^{2^n} 2^n 1^{2^n} \mid n \geq 1\}$, $h(2)=aa$, $h(1)=b$

Вариант 12. $L = \{a^{2^n} b^n a^{2^n} b^n \mid n \geq 1\}$, $h(aa)=1$, $h(b)=22$

Вариант 13. $L = \{a^n b^n c^{2^n} a^n \mid n \geq 1\}$, $h(a)=11$, $h(b)=2$, $h(cc) = 3$

Вариант 14. $L = \{1^n 2^n 1^n 3^{2^n} \mid n \geq 1\}$, $h(1)=aa$, $h(2)=c$, $h(33) = b$

Вариант 15. $L = \{2^{2^n} 1^{2^n} 2^{2^n} 3^n \mid n \geq 1\}$, $h(22)=a$, $h(1)=b$, $h(3) = cc$

Вариант 16. $L = \{a^n c^{2^n} b^{2^n} a^n \mid n \geq 1\}$, $h(a)=3$, $h(bb)=2$, $h(c) = 1$

Вариант 17. $L = \{c^n b^{2^n} c^n a^n \mid n \geq 1\}$, $h(a)=11$, $h(bb)=2$, $h(c) = 3$

Вариант 18. $L = \{b^{2^n} c^n a^n c^n \mid n \geq 1\}$, $h(a)=22$, $h(b)=1$, $h(c) = 3$

Вариант 19. $L = \{c^{2^n} b^{2^n} a^n c^n \mid n \geq 1\}$, $h(a)=1$, $h(bb)=2$, $h(cc) = 3$

Вариант 20. $L = \{a^{2^n} b^n c^{2^n} a^{2^n} \mid n \geq 1\}$, $h(aa)=1$, $h(b)=22$, $h(c) = 3$

Вариант 21. $L = \{2^{2^n} 1^n 3^{2^n} 1^n \mid n \geq 1\}$, $h(1)=cc$, $h(22)=b$, $h(3) = a$

Вариант 22. $L = \{3^n 2^{2^n} 1^{2^n} 3^n \mid n \geq 1\}$, $h(11)=a$, $h(2)=b$, $h(3) = cc$

Вариант 23. $L = \{a^n b^n c^{2^n} a^n \mid n \geq 1\}$, $h(a)=11$, $h(b)=2$, $h(c) = 3$

Вариант 24. $L = \{2^n 3^{2^n} 1^n 2^n \mid n \geq 1\}$, $h(1)=aa$, $h(2)=b$, $h(3) = c$

Критерии оценки индивидуальных заданий (проектов)

– 100-86 баллов выставляется, если содержание и составляющие части соответствуют выданному заданию. Продемонстрировано владение навыками подготовки документа по теме. Фактических ошибок, связанных с пониманием структуры и содержания задания нет.

– 85-76 баллов выставляется, если при выполнении задания допущено не более одной ошибки. Продемонстрировано владение навыками

подготовки документа по теме. Фактических ошибок, связанных с пониманием структуры и содержания задания нет.

– 75-61 балл выставляется, если при выполнении задания допущено не более двух ошибок. Продемонстрировано знание и владение навыками подготовки документа по теме. Допущено не более 2 ошибок, связанных с пониманием структуры и содержания задания.

– 60-50 баллов выставляется, если структура и содержание задания не соответствуют требуемым.

Шкала оценивания

Менее 60 баллов	незачтено	неудовлетворительно
От 61 до 75 баллов	зачтено	удовлетворительно
От 76 до 85 баллов	зачтено	хорошо
От 86 до 100 баллов	зачтено	отлично

Проекты выполняются в рамках курсовой работы в 6 семестре.

Тематика проектов связана с разработкой учебной версии языкового процессора, выполняющего разбор программы на исходном языке и формирование программы на объектном языке.

Содержание курсовой работы.

1. Разработка грамматики исходного языка – подмножества одного из существующих алгоритмических
2. Разработка грамматики объектного языка
3. Разработка контекстных условий исходного языка
4. Проектирование лексического анализатора исходного языка, определение ошибочных ситуаций
5. Выбор метода синтаксического анализа и проектирование синтаксического анализатора исходного языка
6. Проектирование семантического анализатора исходного языка
7. Определение соответствия конструкций исходного и объектного языков, проектирование генератора программы на объектном языке
8. Проектирование всех требуемых структур данных
9. Реализация программной системы, состоящей из лексического, синтаксического и семантического анализаторов, и генератора объектной программы
10. Разработка тестов для проверки работоспособности программной системы
11. Написание отчета по курсовой работе

IX. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

№ п/п	Контролируемые разделы/темы дисциплины	Коды и этапы формирования компетенций		Оценочные средства	
				текущий контроль	промежуточная аттестация
1	Тема 1. Алфавит. Цепочки.	ОПК-2	знает	ИЗ 1-4, тест	Экзамен, вопросы 1-2
	Тема 2. Способы задания языков	ОПК-2	знает	ИЗ 5-10, тест	Экзамен, вопросы 3-6
	Тема 3, 4. Контекстно-свободные языки и грамматики. Преобразования КС – грамматик	ОПК-2 ПК-1	знает	ИЗ 11-12, тест	Экзамен, вопросы 7-17
	Тема 5. Леворекурсивные и праворекурсивные грамматики.	ПК-1	знает	ИЗ 13-15, тест	Экзамен, вопросы 18-21
	Тема 6, 7, 8. Неоднозначные грамматики. Распознаватели, их структура и типы. Автоматы с магазинной памятью	ОПК-2 ПК-1	знает	ИЗ 16-19, тест	Экзамен, вопросы 22-29
	Тема 9. Понятие трансляции. Обзор процесса трансляции.	ПК-1	знает	Тест	Экзамен, вопрос 30
2	Тема 10. Разработка лексического анализатора языка	ПК-1 ПК-5	знает	Блиц-опрос	Экзамен, вопросы 31-37
	Тема 11. Разработка синтаксического анализатора языка	ПК-1 ПК-5	знает	Задания 1-6	Экзамен, вопросы 38-43
	Тема 12, 13. Проверка контекстных условий. Генерация кода	ПК-5	знает	Задание 7	Экзамен, вопросы 44-50

3	Тема 14, 15. Оптимизация кода. Системы автоматического построения трансляторов	ПК-5	знает	Блиц-опрос	Экзамен, вопросы 51-55
4	Практическое занятие № 1. Разработка внешних спецификаций «классического» компилятора	ОПК-2	умеет владеет	Неформальная постановка задачи, грамматика входного языка, требования к ПО	Курсовой проект
	Практическое занятие № 2. Эскизное проектирование компилятора	ПК-1 ПК-5	умеет владеет	АКД, проект верхнего уровня	Курсовой проект
	Практическое занятие № 3. Проектирование лексического анализатора	ОПК-2 ПК-1	умеет владеет	Проект лексического анализатора, набор тестов	Курсовой проект
	Лабораторная работа № 1. Реализация лексического анализатора	ПК-1 ПК-5	умеет владеет	Работающее программное средство	Курсовой проект
	Лабораторная работа № 2. Проектирование синтаксического анализатора	ОПК-2 ПК-1	умеет владеет	Проект синтаксического анализатора	Курсовой проект
	Лабораторная работа № 3. Реализация синтаксического анализатора	ПК-1 ПК-5	умеет владеет	Работающее программное средство	Курсовой проект
5	Лабораторная работа № 4. Реализация семантического анализатора.	ОПК-2 ПК-5	умеет владеет	Работающее программное средство	Курсовой проект
	Практическое занятие № 4. Проектирование генератора.	ПК-4, ОПК-11	умеет владеет	Проект генератора кода	Курсовой проект

6	Практическое занятие № 5. Реализация генератора	ПК-1 ПК-5	умеет владеет	Работающее программное средство	Курсовой проект
	Практическое занятие № 6. Сборка, отладка и тестирование «классического» компилятора	ПК-1 ПК-5	умеет владеет	Работающее программное средство	Курсовой проект
	Практическое занятие № 7. Подготовка документации по разработке компилятора и оформление отчета	ПК-5	умеет владеет	Отчет по разработке компилятора	Курсовой проект

Шкала оценивания уровня сформированности компетенций

Код и формулировка компетенции	Этапы формирования компетенции		критерии	Показатели
ОПК-2 – Способность применять современный математический аппарат, связанный с проектированием, разработкой, реализацией и оценкой качества программных продуктов и программных комплексов в различных областях человеческой деятельности	знает (пороговый уровень)	понятие языка и его свойства; теоретические способы задания языков; теоретические основы информационных процессов преобразования грамматик и распознаватели	Знание основных определений и формулировок свойств	Способность дать определения и формулировки свойств
	умеет (продвинутый)	применять математический аппарат для записи формального представления языка	Умение формально определить языка с использованием математических формализмов	Наличие выполненных заданий
	владеет (высокий)	навыками применения данного математического аппарата при решении конкретных задач	Владение способами задания языков	Наличие отчетов по выполненным заданиям

ПК-1 – Способность демонстрировать базовые знания математических и естественных наук, программирования и информационных технологий	знает (пороговый уровень)	современные методы разработки, проектирования и программирования методов трансляции	Знание процесса создания трансляторов	Способность дать ответы на вопросы
	умеет (продвинутый)	выбирать, проектировать, реализовывать, оценивать качество и анализ эффективности программного обеспечения для решения задач трансляции различных языков программирования	Умение создавать наборы тестов для оценивания качества программ	Наличие разработанных тестов
	владеет (высокий)	автоматизированными средствами верификации программного обеспечения	Владение методами организации компьютерного тестирования программ	Наличие соответствующих фрагментов в документации
ПК-5 – Способность применять современные информационные технологии при проектировании, реализации, оценке качества и анализа эффективности программного обеспечения для решения задач в различных предметных областях	знает (пороговый уровень)	средства разработки, проектирования и реализации компилятора языка	Знание процесса разработки компилятора	Способность пояснить этапы создания компилятора
	умеет (продвинутый)	использовать языки программирования и специализированные средства и оболочки для создания приложений	Умение использовать существующие программные системы при создании компиляторов	Наличие созданной программной системы
	владеет (высокий)	средствами разработки, проектирования и реализации компилятора языка	Владение методами проектирования и программирования компиляторов	Наличие проектов компилятора

Методические рекомендации, определяющие процедуры оценивания результатов освоения дисциплины

Текущая аттестация студентов. Текущая аттестация студентов проводится в соответствии с локальными нормативными актами ДВФУ и

является обязательной.

Текущая аттестация проводится в форме проверки индивидуальных домашних заданий, теста для проверки теоретических знаний, а также в форме защиты курсового проекта, выполняемого в рамках самостоятельной работы параллельно с лабораторными работами и осуществляется ведущим преподавателем.

Объектами оценивания выступают:

- степень усвоения теоретических знаний - оценивается в форме контрольной работы или тестирования;
- уровень овладения практическими умениями и навыками – оценивается в форме защиты индивидуального задания (проекта).

Промежуточная аттестация студентов. Промежуточная аттестация студентов проводится в соответствии с локальными нормативными актами ДВФУ и является обязательной.

По дисциплине предусмотрены экзамены, курсовой проект и зачет, которые проводятся в устной форме.

Критерии выставления оценки студенту на зачете/экзамене

Баллы (рейтингов ой оценки)	Оценка зачета (стандартная)	Требования к сформированным компетенциям
86-100	«зачтено»/ «отлично»	Оценка «зачтено» / «отлично» выставляется студенту, 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, правильно обосновывает принятое решение, владеет разносторонними навыками и приемами выполнения практических задач.
76-85	«зачтено»/ «хорошо»	Оценка «зачтено» / «хорошо» выставляется студенту, если он твердо знает материал, грамотно и по существу излагает его, не допуская существенных неточностей в ответе на вопрос, правильно применяет теоретические положения при решении практических вопросов и задач, владеет необходимыми навыками и приемами их выполнения.

61-75	«зачтено»/ «удовлетворительно»	Оценка «зачтено» / «удовлетворительно» выставляется студенту, если он имеет знания только основного материала, но не усвоил его деталей, допускает неточности, недостаточно правильные формулировки, нарушения логической последовательности в изложении программного материала, испытывает затруднения при выполнении практических работ.
0-60	«не зачтено»/ «неудовлетворительно»	Оценка «не зачтено» / «неудовлетворительно» выставляется студенту, который не знает значительной части программного материала, допускает существенные ошибки, неуверенно, с большими затруднениями выполняет практические работы. Как правило, оценка «неудовлетворительно» ставится студентам, которые не могут продолжить обучение без дополнительных занятий по соответствующей дисциплине.

ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

ВОПРОСЫ К ЭКЗАМЕНУ

1. Алфавит. Цепочка. Длина цепочки. Пустая цепочка.
2. Операции над цепочками. Подцепочка, суффикс и префикс цепочки. Позиция в цепочке. Вхождение подцепочки в цепочку.
3. Языки. Конкатенация языков, операция итерации языка.
4. Гомоморфизм, применение гомоморфизма к языку. Обращение гомоморфизма.
5. Операция подстановки языков вместо символов.
6. Регулярные множества и регулярные выражения. Основные алгебраические свойства регулярных выражений.
7. Формальные грамматики. Выводимость цепочки в грамматике.
8. Вывод в грамматике. Полный и неполный выводы.
9. Терминальное порождение символа. Язык, порождаемый грамматикой.
10. Перестройка вывода. Размеченный вывод. Равносильность выводов.
11. Эквивалентность грамматик. Классификация грамматик по Хомскому: праволинейные, леволинейные, правоавтоматные, левоавтоматные, контекстно-свободные грамматики, грамматики непосредственных составляющих, грамматики общего вида.
12. Связь регулярных множеств и праволинейных грамматик.
13. Контекстно-свободные языки и грамматики. Граф контекстно-свободной (КС) грамматики. Дерево вывода в КС-грамматике. Крона дерева вывода, сечение дерева вывода, крона сечения. Левый (правый) выводы, левывыводимая (правывыводимая) цепочка. Неоднозначные КС грамматики.

14. Удаление бесполезных символов в грамматике
15. Удаление недостижимых символов грамматики.
16. Цепные правила. Циклы в грамматике. Удаление цепных правил.
17. Грамматики без λ -правил.
18. Леворекурсивные и праворекурсивные грамматики.
Непосредственная левая рекурсия. Методы устранения левой рекурсии.
19. Нормальная форма Хомского.
20. Нормальная форма Грейбах
21. Расщепление грамматик.
22. Неоднозначные грамматики. Контекстные условия языков.
23. Распознаватели, их структура и типы.
24. Конечные автоматы. Конфигурации конечного автомата.
Недетерминированные и детерминированные конечные автоматы.
Диаграмма конечного автомата. Полностью и неполностью определенный конечный автомат.
25. Достижимые состояния конечного автомата. Язык, допускаемый конечным автоматом.
26. Эквивалентность автоматов. Канонический конечный автомат.
27. Связь между конечными автоматами и праволинейными грамматиками.
28. Автоматы с магазинной памятью. Конфигурации автомата с магазинной памятью. Недетерминированный и детерминированный автомат с магазинной памятью. Расширенный автомат с магазинной памятью.
29. Язык, допускаемый автоматом с магазинной памятью. Связь между автоматами с магазинной памятью и КС-грамматиками.
30. Формальный язык и языковой процессор на его основе. Теоретический язык и входной язык языкового процессора.
31. Типы языковых процессоров (компиляторы, интерпретаторы, смешанные формы).
32. Структура языкового процессора.
33. Реализация лексического анализа в языковом процессоре.
34. Лексические ошибки.
35. Работа с таблицей имен. Способы хранения длинных имен и строк.
36. Токены, шаблоны, лексемы. Атрибуты токенов. Их вычисление при разборе и хранение.
37. Язык спецификации лексических анализаторов.
38. Понятие о стратегии синтаксического анализа. Нисходящий и восходящий разбор.
39. Важнейшие классы грамматик и соответствующие им анализаторы.
40. Формализмы для задания грамматик.
41. Обработка синтаксических ошибок. Методы продолжения разбора при ошибках.
42. Способы представления результатов анализа: дерево разбора, синтаксическое дерево, польская запись.
43. Работа с таблицей имен при синтаксическом анализе.

44. Контекстные условия языков. Их классы. Атрибутные грамматики.
45. Задание контекстных условий как предикатов над атрибутами.
46. Организация вычисления значений атрибутов при семантическом анализе. Организация проверки контекстных условий.
47. Особенности семантического анализа для языков с полиморфными операциями.
48. Преобразования типов.
49. Область видимости имен в анализируемых программах.
50. Среды периода выполнения. Стеки управления, деревья активации.
51. Организация памяти для объектной программы.
52. Общая схема генерации и внутреннее представление команд.
53. Генерация выражений и операторов присваивания.
54. Генерация управления вычислениями.
55. Генерация подпрограмм и обращений к ним.

Образец экзаменационного билета

Структура экзаменационного билета по курсу «Теория вычислительных процессов и структур»

1. Теоретический вопрос (1-15 и 31-42) из списка вопросов к экзамену).
2. Теоретический вопрос (16-30 и 43-55) из списка вопросов к экзамену).

Экзаменационный билет № __

1. Языки. Конкатенация языков, операция итерации языка.
2. Грамматики без λ -правил.

ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ТЕКУЩЕЙ АТТЕСТАЦИИ

Примеры тестовых заданий

1. Алфавит – это непустое множество символов, которое
 - 1) бесконечное
 - 2) конечное
 - 3) счетное
2. Всякая конечная последовательность символов алфавита A называется
 - 1) предложением
 - 2) цепочкой
 - 3) строкой
3. Не содержащая ни одного символа цепочка является

- 1) укороченной
 - 2) не укороченной
 - 3) пустой
4. Порядок символов в цепочке
- 1) важен
 - 2) не важен.
5. Длина цепочки X равна
- 1) количеству символов в цепочке
 - 2) количеству символов в цепочке минус один
6. Если x и y цепочки, то xy является цепочкой, которая называется
- 1) конкатенацией
 - 2) произведением
7. Если $z=xy$ – цепочка, то x
- 1) голова цепочки z
 - 2) начало цепочки z
 - 3) фраза цепочки z
8. Если $z=xy$ – цепочка, то y
- 1) фраза цепочки z
 - 2) хвост цепочки z
 - 3) конец цепочки z
9. Пусть $z=xy$ – цепочка, то x - правильная голова, если y
- 1) пустая цепочка
 - 2) не пустая цепочка
10. Пусть $z=xy$ – цепочка, то y - правильный хвост, если x
- 1) пустая цепочка
 - 2) не пустая цепочка.
11. Продукцией или правилом подстановки называется упорядоченная пара, которая обычно записывается так $U ::= X$, где U
- 1) символ
 - 2) цепочка символов
 - 3) не пустая цепочка символов
12. Продукцией или правилом подстановки называется упорядоченная пара, которая обычно записывается так $U ::= X$, где X
- 1) символ
 - 2) цепочка символов
 - 3) не пустая конечная цепочка символов
13. Грамматикой $G[z]$ называется, непустое конечное множество правил, где z – начальный символ грамматики, который должен встречаться в левой части правил

- 1) один раз
- 2) более одного раза
- 3) один или более раз

14. Все символы, которые встречаются в левых и правых частях правил грамматики, образуют V

- 1) множество терминальных символов
- 2) множество нетерминальных символов
- 3) алфавит символов

15. В заданной грамматике G символы, которые встречаются в левой части правил называются

- 1) терминальными
- 2) нетерминальными
- 3) начальными

16. В заданной грамматике G символы, которые не входят во множество V_N нетерминальных символов называются

- 1) входными
- 2) терминальными
- 3) конечными

17. Мы говорим в грамматике G цепочка V порождает цепочку w и обозначаем

- 1) $V \Rightarrow w$
- 2) $V \Rightarrow +w$
- 3) $V \Rightarrow -w$

18. Пока в цепочке есть хотя бы один нетерминальный символ из нее можно вывести

- 1) терминальную цепочку
- 2) новую цепочку

19. В грамматике $G[z]$ цепочка x , выводимая из начального символа z , называется

- 1) предложением языка
- 2) сентенциальной формой
- 3) терминальной цепочкой

20. Если две различные грамматики порождают один и тот же язык, то эти грамматики называются

- 1) совпадающими
- 2) эквивалентными
- 3) различающимися

Критерии оценки выполнения тестов

Процент правильных ответов	Оценка
От 95% до 100%	отлично
От 76% до 95%	хорошо
От 61% до 75%	удовлетворительно
Менее 61 %	неудовлетворительно

Примеры индивидуальных заданий

Задание 1. Выпишите все префиксы, суффиксы указанной цепочки. Выпишите все подцепочки указанной цепочки, имеющие длину 3. Указать все вхождения символа a в цепочку w .

Вариант 1. $w = abcaabca$

Вариант 2. $w = acdbacdb$

Вариант 3. $w = cabccabc$

Вариант 4. $w = cccacca$

Вариант 5. $w = bbcabbca$

Вариант 6. $w = aaacaac$

Вариант 7. $w = aaccacc$

Вариант 8. $w = cbabcba$

Вариант 9. $w = caaaca$

Вариант 10. $w = cbaccbac$

Вариант 11. $w = cacbcacb$

Вариант 12. $w = bbaabbaa$

Вариант 13. $w = dbacdbac$

Вариант 14. $w = babcbabc$

Вариант 15. $w = accdaccd$

Вариант 16. $w = cbcacba$

Вариант 17. $w = acbcacbc$

Вариант 18. $w = cabccabc$

Вариант 19. $w = abccabcc$

Вариант 20. $w = adcdadcd$

Задание 2. Привести примеры языков в алфавите Σ

Вариант 1. $\Sigma = \{a, d, b, c, m\}$

Вариант 2. $\Sigma = \{a, d, b, f, k\}$

Вариант 3. $\Sigma = \{a, d, c, s, f\}$

Вариант 4. $\Sigma = \{a, b, c, n, m\}$

Вариант 5. $\Sigma = \{d, b, c, m, k\}$

Вариант 6. $\Sigma = \{b, c, s, k, l\}$

Вариант 7. $\Sigma = \{b, c, n, m, k\}$

Вариант 8. $\Sigma = \{c, k, f, n, l\}$

- Вариант 9. $\Sigma = \{f, h, k, g, l\}$
 Вариант 10. $\Sigma = \{a, b, c, f, i\}$
 Вариант 11. $\Sigma = \{a, b, c, j, i\}$
 Вариант 12. $\Sigma = \{a, b, c, k, l\}$
 Вариант 13. $\Sigma = \{a, d, c, j, i\}$
 Вариант 14. $\Sigma = \{a, d, c, k, i\}$
 Вариант 15. $\Sigma = \{a, d, b, j, i\}$
 Вариант 16. $\Sigma = \{a, d, b, k, f\}$
 Вариант 17. $\Sigma = \{a, b, c, k, m\}$
 Вариант 18. $\Sigma = \{a, c, k, j, i\}$
 Вариант 19. $\Sigma = \{a, c, n, m, f\}$
 Вариант 20. $\Sigma = \{a, c, h, g, k\}$

Задание 3. Устранить левую рекурсию у грамматики с заданной схемой.
 Представить результирующую грамматику.

1. $S \rightarrow Aab; A \rightarrow Sabc; A \rightarrow abc$
2. $S \rightarrow Aa; A \rightarrow SAb; A \rightarrow aa$
3. $S \rightarrow Aaa; A \rightarrow SaAb; A \rightarrow cAc$
4. $S \rightarrow Aa; A \rightarrow b; A \rightarrow Cc; C \rightarrow Sc$
5. $S \rightarrow Aa; A \rightarrow b; A \rightarrow Cc; C \rightarrow c$
6. $S \rightarrow Aa; A \rightarrow Bb; B \rightarrow Saa; B \rightarrow c$
7. $S \rightarrow AabA; A \rightarrow Sabc; A \rightarrow c$
8. $S \rightarrow AaA; A \rightarrow SbA; A \rightarrow aaa$
9. $S \rightarrow aA; A \rightarrow Bb; B \rightarrow Acd; B \rightarrow c$
10. $S \rightarrow aA; A \rightarrow Bb; B \rightarrow Cc; C \rightarrow Ac; C \rightarrow c$
11. $S \rightarrow aA; A \rightarrow Bab; B \rightarrow Cac; C \rightarrow Ac; C \rightarrow c$
12. $S \rightarrow AaA; A \rightarrow Bb; A \rightarrow Sb; B \rightarrow Sa; B \rightarrow c$
13. $S \rightarrow Aab; A \rightarrow Bab; B \rightarrow Sabc; B \rightarrow c$
14. $S \rightarrow Aa; A \rightarrow Bcaa; B \rightarrow Sbssa; B \rightarrow aa$
15. $S \rightarrow Aa; A \rightarrow Bd; A \rightarrow Sfs; B \rightarrow Sba; B \rightarrow ccc$
16. $S \rightarrow Aa; A \rightarrow Ba; B \rightarrow Sb; B \rightarrow BaCc; B \rightarrow dc; C \rightarrow c$
17. $S \rightarrow Aav; A \rightarrow Bas; B \rightarrow Bb; B \rightarrow Sc; B \rightarrow c$
18. $S \rightarrow Aa; A \rightarrow BabB; B \rightarrow Sb; B \rightarrow aad$
19. $S \rightarrow abcA; A \rightarrow BabB; B \rightarrow Aabc; B \rightarrow c$
20. $S \rightarrow aA; A \rightarrow Bab; B \rightarrow Ab; A \rightarrow aa$
21. $S \rightarrow AaA; A \rightarrow Ba; B \rightarrow Sb; S \rightarrow cd$
22. $S \rightarrow AaA; A \rightarrow Bba; A \rightarrow Sb; B \rightarrow Sc; B \rightarrow c; A \rightarrow aaa$
23. $S \rightarrow aA; A \rightarrow BdB; B \rightarrow Ab; A \rightarrow Ac; B \rightarrow c$
24. $S \rightarrow AaA; A \rightarrow BaA; B \rightarrow Ab; B \rightarrow ass$

Задание 4. Для КС-грамматики в нормальной форме Хомского и цепочки, принадлежащей языку, порождаемому данной грамматикой, построить таблицу разбора и левый разбор с использованием метода Кока-Янгера-Касами.

Задание 5. Для КС–грамматики и цепочки, принадлежащей языку, порождаемому данной грамматикой, построить таблицу разбора и правый разбор методом Эрли.

Задание 6. Дана КС–грамматика.

1. Проверить, что она принадлежит к классу LL(1) грамматик.
2. Построить для грамматики таблицу разбора.
3. Для цепочки, принадлежащей языку, порождаемому данной грамматикой, построить историю разбора цепочки.
4. Для цепочки, не принадлежащей языку, порождаемому грамматикой, построить историю разбора. Для этого модифицировать таблицу разбора, указав диагностические сообщения об ошибке в требуемых клетках таблицы.

Задание 7. Для КС–грамматика класса LL(1) определить таблицу разбора, управляющую анализатором, продолжающим разбор при обнаружении ошибок. Для этого определить синхронизирующие множества и действия при ошибках. Построить историю разбора цепочки с ошибками.

Задание 8. Для КС–грамматика класса LL(1) построить анализатор методом рекурсивного спуска. Анализатор должен продолжать разбор в случае наличия ошибок

Задание 9. Дана КС–грамматика.

1. Проверить, что она принадлежит к классу грамматик простого предшествования.
2. Построить для грамматики отношения простого предшествования.
3. Для цепочки, принадлежащей языку, порождаемому данной грамматикой, построить историю разбора цепочки.
4. Для цепочки, не принадлежащей языку, порождаемому грамматикой, построить историю разбора. Определить диагностические сообщения об ошибках.