

**Федеральное государственное автономное образовательное учреждение
высшего образования
«Санкт-Петербургский политехнический университет Петра Великого»**

УТВЕРЖДАЮ
Директор ИКНК
_____ Д.П. Зегжда
«17» июня 2024 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

«Теория автоматов и формальных языков»

Разработчик	Высшая школа программной инженерии
Направление (специальность) подготовки	09.03.04 Программная инженерия
Наименование ООП	09.03.04_01 Технология разработки и сопровождения качественного программного продукта
Квалификация (степень) выпускника	бакалавр
Образовательный стандарт	СУОС
Форма обучения	Очная

СОГЛАСОВАНО	Соответствует СУОС
Руководитель ОП	Утверждена протоколом заседания
_____ А.В. Петров	высшей школы "ВШПИ" от «21» мая 2024 г. № 1

РПД разработал:

Специалист по учебно-методической работе 1 категории Т.А. Вишневская

1. Цели и планируемые результаты изучения дисциплины

Цели освоения дисциплины

сформировать специалистов, умеющих обоснованно и результативно применять существующие и осваивать новые методы и модели для анализа формальных языков (в том числе языков программирования), уметь использовать алгоритмы их синтаксического и семантического анализа для построения трансляторов произвольных языков.

Результаты обучения выпускника

Код	Результат обучения (компетенция) выпускника ООП
ОПК-1	Способен применять естественнонаучные и общеинженерные знания, методы математического анализа и моделирования, теоретического и экспериментального исследования в профессиональной деятельности
ИД-7 ОПК-1	Применяет методы и модели анализа формальных языков (в том числе языков программирования) для построения формальной грамматики при решении задач трансляции

Планируемые результаты изучения дисциплины

знания:

- Знает методы постановки задачи трансляции формальных языков и способы построения формальной грамматики для специализированного языка интерфейса пакета прикладных программ;

умения:

- Умеет строить формальную модель (грамматику) специализированного языка и языка программирования

навыки:

- Владеет методами выполнения синтаксического анализа предложений формального языка

2. Место дисциплины в структуре ООП

В учебном плане дисциплина «Теория автоматов и формальных языков» относится к модулю «Математические основы программной инженерии».

Изучение дисциплины базируется на результатах освоения следующих дисциплин:

- Математическая логика

3. Распределение трудоёмкости освоения дисциплины по видам учебной работы и формы текущего контроля и промежуточной аттестации

3.1. Виды учебной работы

Виды учебной работы	Трудоемкость по семестрам
	Очная форма
Лекционные занятия	30
Практические занятия	12
Самостоятельная работа	56
Промежуточная аттестация (зачет с оценкой)	6
Курсовое проектирование	4
Общая трудоемкость освоения дисциплины	108, ач
	3, зет

3.2. Формы текущего контроля и промежуточной аттестации

Формы текущего контроля и промежуточной аттестации	Количество по семестрам
	Очная форма
Текущий контроль	
Контрольные, шт.	1
Курсовые работы, шт.	1
Промежуточная аттестация	
Зачеты с оценкой, шт.	1

4. Содержание и результаты обучения

4.1 Разделы дисциплины и виды учебной работы

№ раздела	Разделы дисциплины, мероприятия текущего контроля	Очная форма		
		Лек, ач	Пр, ач	СР, ач
1.	Конечные автоматы и языки			

1.1.	Автоматы как модели задания языков	2	1	3
1.2.	Теория конечных автоматов-распознавателей формальных языков	2	1	3
1.3.	Трансляторы автоматных языков.	2	1	3
1.4.	Регулярные выражения	2	1	2
2.	Грамматики Хомского			
2.1.	Язык, грамматика, синтаксис, семантика	2	1	2
2.2.	Иерархия распознающих автоматов	2	1	3
3.	Атрибутная семантика			
3.1.	Синтаксически-ориентированная трансляция. Атрибуты.	2	1	2
3.2.	Грамматики арифметических выражений	2	1	3
4.	Синтаксический анализ			
4.1.	Нисходящий синтаксический анализ	4	1	3
4.2.	Восходящий синтаксический анализ	2	1	3
5.	Трансляция языков высокого уровня			
5.1.	Методы трасляции конструкций языков высокого уровня.	2	1	3
5.2.	Компиляторы компиляторов. Структура и функции компиляторов компиляторов. Задание синтаксиса языка. Задание семантики языка.	2	1	2
5.3.	Методы продолжения трансляции после обнаружения ошибок. Наведенные ошибки.	2	0	4
5.4.	Оптимизация в трансляторах.	2	0	4
Итого по видам учебной работы:		30	12	56
Зачеты с оценкой, ач				10
Часы на контроль, ач				0
Курсовое проектирование				4
Промежуточная аттестация (зачет с оценкой)				6
Общая трудоёмкость освоения: ач / зет				108 / 3

4.2. Содержание разделов и результаты изучения дисциплины

Раздел дисциплины	Содержание
1. Конечные автоматы и языки	
1.1. Автоматы как модели задания языков	Понимать связь абстрактных автоматов и проблем теории формальных языков. Уметь приводить примеры формальных языков. Знать определение грамматики, различия между порождающими и распознающими грамматиками. Знать определение автоматных языков и конечных автоматов-распознавателей как конечных моделей задания (распознавания) формальных языков. Понимать лемму о накачке и уметь ее применять для определения автоматности формального языка
1.2. Теория конечных автоматов-распознавателей формальных языков	Уметь проверять эквивалентность двух автоматов-распознавателей, выполнять минимизацию автоматов-распознавателей. Понимать идею модели недетерминированных конечных автоматов-распознавателей. Уметь строить эквивалентный детерминированный КА для недетерминированного КА
1.3. Трансляторы автоматных языков.	Уметь строить трансляторы для автоматных языков. Знать примеры построения трансляторов для автоматных языков. Язык записи римских чисел. Язык описания систем дифференциальных уравнений. Язык описания выводимых страниц в трансляторе печати программы документов Windows.
1.4. Регулярные выражения	Понимать связь регулярных множеств и регулярных выражений. Уметь доказывать теорему Клини. Знать возможности и технологию использования регулярных выражений для задач обработки текстов
2. Грамматики Хомского	
2.1. Язык, грамматика, синтаксис, семантика	Иметь ясное представление понятий языка, грамматики, синтаксиса и семантики. Понимать проблему двусмысленности фраз языков и идеи Хомского о синтаксически-ориентированном подходе к трансляции. Понимать идеи порождающих грамматик Хомского, как 'работает' порождающая грамматика. Знать классификацию грамматик, распознающих автоматов и порождающих грамматик

2.2. Иерархия распознающих автоматов	<p>Уметь строить конечный автомат по автоматной грамматике, МП-автомат по КС-грамматике Уметь использовать лемму о накачке для определения неавтоматности языков Уметь преобразовывать КС-грамматику к нормальным формам Хомского и Грейбах Уметь выполнять левый и правый канонический вывод в КС-грамматике Знать примеры порождающих грамматик других типов (сетей Петри) Уметь строить различные деревья вывода для неоднозначных КС-грамматик Знать КС-грамматики конструкций языков высокого уровня, грамматику арифметических выражений</p>
3. Атрибутная семантика	
3.1. Синтаксически-ориентированная трансляция. Атрибуты.	<p>Понимать трансляции автоматных языков как частный случай синтаксически-ориентированной трансляции Понимать идеи атрибутной семантики, использование синтезируемых и наследуемых атрибутов Уметь проверять корректность атрибутной семантики</p>
3.2. Грамматики арифметических выражений	<p>Уметь использовать атрибутную семантику в трансляции языков Уметь строить семантические атрибуты в различных задачах трансляции арифметических выражений Уметь построить КС-грамматику, семантические атрибуты и трансляцию арифметических выражений для языка APL. Уметь использовать семантические атрибуты в задачах трансляции</p>
4. Синтаксический анализ	
4.1. Нисходящий синтаксический анализ	<p>Понимать проблемы восходящих и нисходящих алгоритмов синтаксического анализа Уметь восстанавливать вывод цепочек языков, порождаемых s-грамматиками Уметь строить множества First(k) и Follow(k) по КС-грамматике уметь восстанавливать вывод цепочек языков, порождаемых LL(k)-грамматиками. Уметь строить синтаксические диаграммы и распознаватели для грамматик рекурсивного спуска</p>
4.2. Восходящий синтаксический анализ	<p>Уметь строить матрицу отношений предшествования для грамматик предшествования уметь восстанавливать вывод цепочек языков, порождаемых LR(0)-грамматиками уметь строить синтаксические анализаторы для SLR(1), LALR(1) и LR(1) грамматик уметь выполнять синтаксический анализ двусмысленных цепочек методами Эрли и Кока-Янгера-Касами</p>
5. Трансляция языков высокого уровня	

5.1. Методы трансляции конструкций языков высокого уровня.	Уметь разрабатывать семантические процедуры для распространенных конструкций языков высокого уровня: арифметические выражения/, конструкция условный оператор, оператор цикла, оператор перехода.
5.2. Компиляторы компиляторов. Структура и функции компиляторов компиляторов. Задание синтаксиса языка. Задание семантики языка.	Уметь использовать системы автоматического построения компиляторов для разработки трансляторов. Понимать структуру компилятора компиляторов, методы задания синтаксиса, методы задания семантики языка.
5.3. Методы продолжения трансляции после обнаружения ошибок. Наведенные ошибки.	Уметь использовать методы продолжения трансляции после обнаружения синтаксических ошибок. Понимать различие между анализируемым входным текстом программы и содержимом стека вызова рекурсивных процедур при трансляции. Приемы борьбы с наведенными ошибками при трансляции языков.
5.4. Оптимизация в трансляторах.	Знать основные методы оптимизации при построении трансляторов. Уметь использовать регистры для запоминания промежуточных значений выражения при трансляции сложных арифметических выражений. Знать методы оптимизации вычисления логических выражений.

5. Образовательные технологии

В преподавании курса используются преимущественно традиционные образовательные технологии: • лекции, • лабораторные занятия, • выполнение курсовой работы. Вместе с тем, нетрадиционным для соответствующего курса являются: • лекции и лабораторные занятия с использованием слайдов • использование модулей автоматизированного контроля знаний, доступных в системе управления курсами Moodle Студенты выполняют курсовой проект по разработке своего собственного транслятора с языка Милан. Фактически, курсовая работа состоит в том, что студенту дается свой вариант дополнений, которые нужно добавить в язык Милан, и далее, студент должен изменить лексический анализатор, синтаксический анализатор и написать семантические процедуры, которые позволяют реализовать дополнения в трансляторе языка Милан.

6. Лабораторный практикум

Не предусмотрено

7. Практические занятия

№ раздела	Наименование практических занятий (семинаров)	Трудоемкость, ач
		Очная форма
1.	Построение грамматик формальных языков.	1
2.	Конечные автоматы-распознаватели. Автоматные языки.	1
3.	Построение трансляторов простых автоматных языков.	1
4.	Регулярные выражения. Преобразования регулярных выражений в конечный автомат и обратно.	1
5.	Порождающие грамматики Хомского. Построение грамматик Хомского для формальных языков.	1
6.	Иерархия грамматик Хомского. Построение различных грамматик для одного и того же языка.	1
7.	Построение КС-грамматик для формальных языков.	1
8.	Построение машины Тьюринга, распознающей формальные языки.	1
9.	Атрибутные грамматики для трансляции КС-языков.	0
10.	Атрибутные грамматики арифметических выражений.	1
11.	Построение синтаксических анализаторов для s-грамматик.	1
12.	LL(k)-грамматики и их синтаксический анализ.	1
13.	Восходящий синтаксический анализ. Примеры построения вывода цепочек формальных языков, порождаемых грамматикой простого предшествования	1
14.	.Примеры синтаксического анализа LR(0), :R(1), SLR(1) и LALR(1) языков 11	0
Итого часов		12

8. Организация и учебно-методическое обеспечение самостоятельной работы

Самостоятельная проработка каждого из разделов курса. Контроль проработки - на практических занятиях при выполнении упражнений и при защите курсовой работы:

1. работа с лекционным материалом, с учебной литературой - слайды лекций выложены для возможного доступа студентов, контроль - при опросе на практических занятиях.
- 2 опережающая самостоятельная работа (изучение нового материала до его изложения на занятиях) - на сайте курса выложены пособия и оригинальные статьи, содержащие этот материал. Самоконтроль.
- 3 самостоятельное изучение разделов дисциплины - на сайте выложено учебное пособие автора курса, по которому можно изучить некоторые технические вопросы. Контроль при опросе на практических занятиях.
4. выполнение домашних заданий, домашних контрольных работ - проверяется преподавателем.
5. подготовка к лабораторным работам, к практическим и семинарским занятиям - проверяется при выборочном опросе на практических занятиях.
6. подготовка к контрольным работам, коллоквиумам, зачётам - проверяется по результатам контрольных и зачета.
- 7.. подготовка к экзаменам состоит в просмотре всего корпуса лекций для понимания основных концепций дисциплины.

Примерное распределение времени самостоятельной работы студентов

Вид самостоятельной работы	Примерная трудоемкость, ач
	Очная форма
Текущая СР	
работа с лекционным материалом, с учебной литературой	6
опережающая самостоятельная работа (изучение нового материала до его изложения на занятиях)	2
самостоятельное изучение разделов дисциплины	3
выполнение домашних заданий, домашних контрольных работ	7
подготовка к лабораторным работам, к практическим и семинарским занятиям	5
подготовка к контрольным работам, коллоквиумам	8
Итого текущей СР:	31
Творческая проблемно-ориентированная СР	
выполнение расчётно-графических работ	0
выполнение курсового проекта или курсовой работы	9
поиск, изучение и презентация информации по заданной проблеме, анализ научных публикаций по заданной теме	0
работа над междисциплинарным проектом	0
исследовательская работа, участие в конференциях, семинарах, олимпиадах	0
анализ данных по заданной теме, выполнение расчётов, составление схем и моделей на основе собранных данных	0
Итого творческой СР:	9
Общая трудоемкость СР:	56

9. Учебно-методическое обеспечение дисциплины

9.1. Адрес сайта курса

<https://dl.spbstu.ru/course/view.php?id=6461>

9.2. Рекомендуемая литература

Основная литература

№	Автор, название, место издания, издательство, год (годы) издания	Год изд.	Источник
1	Карпов Ю.Г. Model checking. Верификация параллельных и распределенных программных систем: СПб.: БХВ-Петербург, 2010.	2010	ИБК СПбПУ
2	Карпов Ю.Г. Автоматы и формальные языки: Санкт-Петербург: Изд-во Политехн. ун-та, 2015.	2015	ИБК СПбПУ

Ресурсы Интернета

1. Методические указания по выполнению курсовой работы по разработке компилятора:
<http://dcn.icc.spbstu.ru/~karpov>
2. Вводный материал поо разработке компиляторов.: <http://turbo51.com/download/Compilers-Principles-Techniques-and-Tools-Book1-Preview.pdf>

9.3. Технические средства обеспечения дисциплины

Студенты могут выполнять курсовую работу, используя компьютерный класс, оборудованный стандартными компьютерами с установленным на нем свободным программным обеспечением.

10. Материально-техническое обеспечение дисциплины

Студенты могут выполнять курсовую работу, используя компьютерный класс, оборудованный стандартными компьютерами с установленным на нем свободным программным обеспечением.

11. Критерии оценивания и оценочные средства

11.1. Критерии оценивания

Для дисциплины «Теория автоматов и формальных языков» формой аттестации является зачёт с оценкой. Дисциплина реализуется с применением системы индивидуальных достижений.

Текущий контроль успеваемости

Максимальное значение персонального суммарного результата обучения (ПСРО) по приведенной шкале - 100 баллов

Максимальное количество баллов приведенной шкалы по результатам прохождения двух точек контроля - 80 баллов.

Подробное описание правил проведения текущего контроля с указанием баллов по каждому контрольному мероприятию и критериев выставления оценки размещается в СДО в навигационном курсе дисциплины.

Промежуточная аттестация по дисциплине

Максимальное количество баллов по результатам проведения аттестационного испытания в период промежуточной аттестации – 20 баллов приведенной шкалы.

Промежуточная аттестация по дисциплине проводится в соответствии с расписанием.

Экзамен проводится в форме решения задач , которые случайно выбраны из задачника, изначально доступного для студентов.

Для получения зачета по курсу студент должен полностью выполнить домашние расчетно-контрольные работы (см. раздел самостоятельная работа), а также выполнить курсовую работу.

Экзамен состоит в решении задач, которые для каждого варианта берутся из задачника, заранее передаваемого студентам. Задачи построены таким образом, чтобы их нельзя было решить без понимания теоретических вопросов по материалу лекций

Результаты промежуточной аттестации, определяются на основе баллов, набранных в рамках применения, СИД

Баллы по приведенной шкале в рамках применения СИД (ПСРО+ ПА)	Оценка по результатам промежуточной аттестации
	Экзамен/диф.зачет/зачет
0 - 60 баллов	Неудовлетворительно/не зачленено
61 - 75 баллов	Удовлетворительно/зачленено
76 - 89 баллов	Хорошо/зачленено
90 и более	Отлично/зачленено

11.2. Оценочные средства

Оценочные средства по дисциплине представлены в фонде оценочных средств, который является неотъемлемой частью основной образовательной программы и размещается в электронной информационно-образовательной среде СПбПУ на портале etk.spbstu.ru

12. Методические рекомендации по организации изучения дисциплины

Во время лекций и лабораторных занятий должны приводиться примеры использования грамматик и формальных языков в различных отраслях информатики. Существует два типа грамматик: порождающие и распознающие

Порождающая грамматика языка L - конечный набор правил, позволяющих строить все “правильные” предложения языка L, и применение которых не дает ни одного “неправильного” предложения, не принадлежащего L.

Распознающая грамматика – алгоритм, принимающий произвольную цепочку над словарем V и дающий на выходе один из двух возможных ответов: “данная цепочка принадлежит языку L”, либо “данная цепочка не принадлежит языку L”.

Важно также понимать различие синтезированных и наследуемых семантических атрибутов. Синтезируемые атрибуты определяются через атрибуты потомков нетерминала. В дереве вывода они вычисляются снизу вверх. Пример: значение выражения: для листьев дерева (операндов) значения известны, а для родителей (подвыражений) вычисляется по значениям их детей и типу операции. Синтезируемые атрибуты нетерминала зависят от того, из чего строится эта конструкция. Это атрибуты нетерминала в левой части правила, которые определяются через атрибуты символов правой части правила. **Наследуемые атрибуты** нетерминала определяются через атрибуты его непосредственного предка, т.е. **от вида правила**, в которое нетерминал входит в правой части. В дереве вывода они вычисляются сверху вниз. Пример: тип переменной зависит не от того, какое у нее имя, а от того, в какой конструкции описания (**integer** либо **real**) определено ее имя. Наследуемые атрибуты зависят от того, в какую

конструкцию и как входит данная конструкция (нетерминал). Это атрибуты символов правой части правила, которые определяются через атрибуты других символов того же прправила (и в левой части, и в правой части правила).

13. Адаптация рабочей программы для лиц с ОВЗ

Адаптированная программа разрабатывается при наличии заявления со стороны обучающегося (родителей, законных представителей) и медицинских показаний (рекомендациями психолого-медицинской-педагогической комиссии). Для инвалидов адаптированная образовательная программа разрабатывается в соответствии с индивидуальной программой реабилитации.