



Автоматы и формальные языки

Карпов Юрий Глебович
профессор, д.т.н., зав.кафедрой
“Распределенные вычисления и компьютерные сети”
Санкт-Петербургского Политехнического университета

karpov@dcn.infos.ru



Структура курса

■ Конечные автоматы-распознаватели	– 4 л
■ Порождающие грамматики Хомского	– 3 л
■ Атрибутные трансляции и двусмысленные КС-грамматики	– 2 л
■ Распознаватели КС-языков и трансляция	– 6 л
■ Дополнительные лекции	- 2 л
	=====
	17 л
■ Что должен знать и уметь студент (разбор задач по курсу)	1 л (если успеем)
	=====
	18 л

Требования к знаниям студента по разделам курса

Конечные
автоматы



Грамматики
Хомского

Атрибутная
семантика

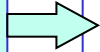
Синтаксичес-
кий анализ

Трансляция
ЯВУ

- понимать связь абстрактных автоматов и проблем теории формальных языков, уметь определять формальные языки
- знать определение грамматики, различия между порождающими и распознающими грамматиками
- знать определение автоматных языков и конечных автоматов (КА) как моделей задания формальных языков
- уметь использовать Лемму о накачке для доказательства неавтоматности языка
- уметь проверять эквивалентность двух автоматов-распознавателей (теорема Мура), выполнять минимизацию КА
- уметь строить синхронную и асинхронную композицию КА
- понимать идею модели недетерминированного КА-распознавателя, уметь строить эквивалентный детерминированный КА для заданного недетерминированного КА (теорема Рабина-Скотта)

Требования к знаниям студента по разделам курса

Конечные
автоматы



Грамматики
Хомского

Атрибутная
семантика

Синтаксичес
кий анализ

Трансляция
ЯВУ

- понимать связь КА и синтаксических диаграмм
- уметь строить трансляторы для автоматных языков, лексические анализаторы для языков высокого уровня
- понимать связь и различие регулярных множеств, регулярных выражений, КА и автоматных грамматик
- уметь доказывать теорему Клини и использовать ее
- уметь использовать возможности регулярных выражений и технологию работы с ними для задач обработки текстов
- понимать и уметь доказывать теоремы Мура (об эквивалентности КА), Рабина-Скотта (НДКА и ДКА), Майхилла-Нероуда (правый инвариант, эквивалентность на множ цепочек языка), Клини (регВыр)



Требования к знаниям студента по разделам курса

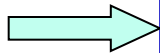
Конечные
автоматы

Грамматики
Хомского

Атрибутная
семантика

Синтаксический
анализ

Трансляция
ЯВУ



- иметь ясное представление о понятиях: формальный язык, грамматика, синтаксис, семантика
- понимать проблему двусмысленности фраз языков и идеи Хомского о синтаксически-ориентированном подходе к трансляции
- понимать идеи порождающих грамматик Хомского, как 'работает' порождающая грамматика Хомского
- знать классификацию грамматик, распознающих автоматов
- знать примеры использования грамматик типов 0 и 1 для определения языков

Требования к знаниям студента по разделам курса

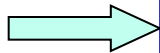
Конечные
автоматы

Грамматика
Хомского

Атрибутная
семантика

Синтаксический
анализ

Трансляция
ЯВУ



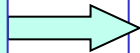
- уметь строить КА по автоматной грамматике, МП-автомат по КС-грамматике и обратно
- уметь упрощать КС-грамматику и преобразовывать ее к нормальным формам Хомского и Грейбах
- уметь выполнять левый и правый канонические выводы цепочек в КС-грамматике
- знать примеры порождающих грамматик других типов (сети Петри, БНФ-нотация, синтаксические диаграммы)
- уметь строить различные деревья вывода для неоднозначных КС-грамматик
- уметь строить КС-грамматики арифметических выражений

Требования к знаниям студента по разделам курса

Конечные
автоматы

Грамматики
Хомского

Атрибутная
семантика



Синтаксический
анализ

Трансляция
ЯВУ

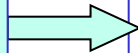
- понимать идею трансляции автоматных языков как частного случая синтаксически-ориентированной трансляции
- понимать идеи атрибутной семантики, различие и использование синтезируемых и наследуемых атрибутов
- уметь строить зависимости семантических атрибутов на дереве синтаксического анализа
- уметь проверять корректность атрибутной семантики заданной грамматики алгоритмом Кнута
- уметь использовать атрибутную семантику в трансляции КС-языков

Требования к знаниям студента по разделам курса

Конечные
автоматы

Грамматики
Хомского

Атрибутная
семантика



Синтаксический
анализ

Трансляция
ЯВУ

- уметь строить семантические атрибуты в различных задачах трансляции арифметических выражений
- понимать организацию виртуальной стековой машины (как целевой машины) и уметь строить программу стековой машины по программам на языках типа Паскаль (для языка Милан)
- уметь строить КС-грамматику, семантические атрибуты и выполнять трансляцию арифметических выражений для языка APL
- понимать другие методы определения семантики языков: операционную семантику, аксиоматическую семантику Хоара

Требования к знаниям студента по разделам курса

Конечные
автоматы

Грамматики
Хомского

Атрибутная
семантика

Синтаксический
анализ

Трансляция
ЯВУ

- понимать проблемы восходящих и нисходящих алгоритмов синтаксического анализа
- уметь определять класс s -грамматик, восстанавливать вывод цепочек языков, порождаемых s -грамматиками
- уметь строить множества $FIRST()$ и $FOLLOW()$ по КС-грамматике, множества выбора $LOOKAHEAD$ по заданной КС-грамматике
- уметь распознавать класс $LL(k)$ -грамматик, восстанавливать вывод строк, порождаемых $LL(k)$ -грамматиками
- уметь распознавать класс грамматик рекурсивного спуска, понимать идеи синтаксического анализа этого класса языков
- уметь строить синтаксические диаграммы и распознаватели, использовать операционную семантику для трансляции языков рекурсивного спуска



Требования к знаниям студента по разделам курса

Конечные
автоматы

Грамматики
Хомского

Атрибутная
семантика

Синтаксический
анализ

Трансляция
ЯВУ

- уметь строить матрицу отношений предшествования для грамматик предшествования и использовать ее для синтаксического анализа языков простого предшествования
- понимать идеи синтаксического анализа LR(k)-грамматик Д.Кнута, уметь восстанавливать вывод цепочек языков, порождаемых LR(0)-грамматиками
- уметь строить синтаксические анализаторы для SLR(1), LALR(1) и LR(1) грамматик
- понимать возможности и ограничения атрибутных вычислений при нисходящем и восходящем синтаксическом анализе
- уметь выполнять синтаксический анализ двусмысленных цепочек методами Эрли и Кока-Янгера-Касами



Требования к знаниям студента по разделам курса

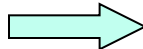
Конечные
автоматы

Грамматики
Хомского

Атрибутная
семантика

Синтаксический
анализ

Трансляция
ЯВУ



- знать способы описания синтаксиса языков программирования: синтаксические диаграммы (Вирт), расширенную форму Бэкуса-Наура (БНФ)
- понимать ограниченность КС грамматик при задании языков высокого уровня, необходимость выполнения контекстного анализа при трансляции ЯВУ
- знать формы промежуточного представления программ: синтаксическое дерево, атрибутно-синтаксическое дерево, триадное представление, тетрады, обратная польская запись. Байт-коды внутреннего представления (Java-код, Р-код и др.)
- знать структуру и функции основных блоков компиляторов языков программирования



Требования к знаниям студента по разделам курса

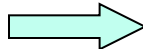
Конечные
автоматы

Грамматики
Хомского

Атрибутная
семантика

Синтаксический
анализ

Трансляция
ЯВУ



- знать основные идеи, структуру и функции генераторов лексических анализаторов (на примере LEX, FLEX)
- знать основные идеи, структуру и функции компиляторов компиляторов (на примере YACC, Bison)
- уметь использовать алгоритмы синтаксического анализа основных классов КС языков для компиляции основных управляющих структур языков высокого уровня
- уметь использовать подходящие структуры данных и алгоритмы для обработки типовых конструкций современных языков программирования



Самостоятельные работы по разделам курса

1. Проверка эквивалентности двух заданных КА-распознавателей
2. Минимизировать заданный КА-распознаватель
3. По заданному НДКА построить эквивалентный ДКА
4. По заданному КА построить регулярное выражение, задающее распознаваемый автоматом язык, и обратно, восстановить КА по построенному регулярному выражению. Проверить эквивалентность исходного и полученного автоматов
5. Упростить заданную КС-грамматику, выбросив бесполезные продукции
6. По заданной грамматике и семантике построить атрибутивную грамматику
7. Построить для заданной $LL(1)$ -грамматики таблицу разбора. Выполнить синтаксический анализ цепочки в этой грамматике
8. Построить для заданной грамматики рекурсивного спуска синтаксические диаграммы и по ним алгоритм разбора. Выполнить синтаксический анализ цепочки в этой грамматике
9. Построить для заданной грамматики простого предшествования матрицу отношений предшествования. Выполнить синтаксический анализ цепочки в этой грамматике
10. Построить (вручную) для заданной грамматики $LR(1)$ и $LALR(1)$ - автомат разбора. Выполнить синтаксический анализ цепочки

ПОЛУЧЕНИЕ ДОПОЛНИТЕЛЬНЫХ БАЛЛОВ (до 15 и более дополнительных баллов)

1) Решить задачу распознавания в случайном потоке символов цепочек простых регулярных языков в одной из библиотек (Java, Perl, ...) для работы с регулярными выражениями :

- даты в форматах *yyyy-mm-dd*, *dd/mm/yyyy* или других
- адресов электронной почты
- чисел с плавающей запятой
- римских чисел
- номеров телефонов в форматах *+ц (ццц) ццц цц цц*, или *+ц-ццц-ццц-цц-цц*, ...
- ...

2) Построить программу, которая по введенному КА строит регулярное выражение, описывающее допустимый автоматом язык, и по введенному регулярному выражению строит КА, распознающий все цепочки, определяемые этим РВ



Курсовая работа – дополнительные баллы II

- 3) Реализовать компилятор языка обработки констант и переменных типа множеств (задания множеств с операциями объединения, пересечения, дополнения)
- 4) Реализовать компилятор в Yacc (Bison) полного языка типа Паскаль
- 5) Построить свой конструктор для какого-нибудь класса КС-грамматик (конструктор – программа, которая по введенной грамматике проверяет принадлежность ее заданному классу грамматик (s-грамматики, LL(1)-грамматики, грамматик простого предшествования, грамматик LR(1), ...), и выполняет синтаксический анализ любой входной цепочки этого языка)
- 6) Реализовать в своем компиляторе Милана какой-нибудь алгоритм обработки синтаксических ошибок (например, с использованием маяков) и продолжением трансляции после обнаружения ошибок (с возможным их исправлением)
- 7) Реализовать сложные расширения языка Милан:
 - массивы
 - описания переменных и области видимости
 - функции
 - ссылочные переменные

Требование для сдачи экзамена/зачета:
уметь решать задачи из прилагаемого
сборника задач



Спасибо за внимание