IV Семинар. Формальные и аналитические грамматики

Формальный язык — это математическая модель реального языка.

Под реальным языком здесь понимается некий способ коммуникации субъектов друг с другом. Для общения субъекты используют конечный набор знаков, которые образуют линейные последовательности. Такие последовательности обычно называют словами или предложениями.

Таким образом, здесь рассматривается только т.н. коммуникативная функция языка, которая изучается с использованием математических методов. Другие функции языка здесь не изучаются и, потому, не рассматриваются.

Чтобы изучать языки с помощью математических методов, необходимо сначала выделить из языка его свойства, которые представляются важными для изучения, а затем эти свойства строго определить. Полученная таким образом абстракция будет называться формальным языком.

Алфавит - непустое множество символов. Не путать алфавит со словарём. Словарь в отличие от алфавита может быть пустым.

Для обозначения алфавита обычно используется латинское V, а для обозначения символов алфавита — начальные строчные буквы латинского алфавита. Выражение V = {a,b} обозначает алфавит из двух символов a и b.

Цепочка представляет собой конечную последовательность символов, может быть пустой.

Формальный язык L над алфавитом V — это произвольное множеств цепочек, составленных из символов алфавита V.

Произвольность здесь означает тот факт, что язык может быть как пустым, так и бесконечным.

Итак, грамматика языка описывает законы внутреннего строения его цепочек. Такие законы обычно называют синтаксическими закономерностями.

Формальные грамматики - средства описания формальных языков.

В теории формальных языков основными являются две задачи:

1. каким образом строить правильные предложения заданного языка?;
2. как установить, является ли данное предложение синтаксически правильным, допустимым для данного языка?

Первая задача решается при помощи формальных грамматик. Формальная грамматика содержит строгие правила порождения правильных предложений языка. Определение формальной грамматики имеет следующий вид:

Формальная (порождающая) грамматика G - это формальная система, определяемая четверкой символов:

G = { N, T, P, S },

где:

N – конечное непустое множество нетерминальных (или вспомогательных) символов;

T – конечное непустое множество терминальных (или конечных) символов;
S – начальный или корневой символ;
P – конечное множество правил вывода (правил продукции).

- **Терминал** (терминальный символ) — объект, непосредственно присутствующий в словах языка, соответствующего грамматике, и имеющий конкретное, неизменяемое значение (обобщение понятия «буквы»). В формальных языках, используемых на компьютере, в качестве терминалов обычно берут все или часть стандартных символов ASCII — латинские буквы, цифры и спец. символы.
- **Нетерминал** (нетерминальный символ) — объект, обозначающий какую-либо сущность языка (например: формула, арифметическое выражение, команда) и не имеющий конкретного символьного значения.

Множества N и T являются непересекающимися множествами: N Ç T =Æ.
Множества N и T представляют собой алфавиты нетерминальных и терминальных символов.
Начальный символ S является исходным символом, из которого при помощи правил вывода строятся все предложения (цепочки) данного языка.
Правила вывода P определяют синтаксис формального языка, т.е. правила построения правильных предложений языка.

Иерархия Хомского — классификация формальных языков и формальных грамматик, согласно которой они делятся на 4 типа по их условной сложности.

Порождающая грамматика Хомского задается как множество «правил порождения» (продукций). Каждое правило является просто парой цепочек (w', w'') и задает возможность замены левой цепочки на правую при генерации цепочек языка, задаваемого грамматикой. По этой причине, правила обычно записывают в виде w' --> w'', указывая конкретно, что на что можно заменять.

Тип определяется совокупностью правил вывода.
Типы грамматик:

- неограниченные 0 типа

- контекстно-зависимые 1 типа
  Контекстно-зависимая грамматика имеет правила вида `w' A w'' --> w' alpha w''`. Здесь `w'` и `w''` — цепочки (может быть пустые), составленные из терминальных и нетерминальных символов грамматики, `alpha` — непустая цепочка из тех же символов. Иначе говоря, нетерминальный символ `A` заменяется на цепочку `alpha` в контексте цепочек `w'` и `w''`.
- контекстно-свободные 2 типа
  Контекстно-свободные грамматики имеют правила вида: `A --> alpha`. В

левой части правила должен стоять один символ (нетерминальный), а справа может быть любая цепочка из терминальных и нетерминальных символов (в том числе и пустая).

- замкнутые
  Этот класс грамматик задает алгоритм порождения цепочек присоединением некоторого количества терминальных символов с правого или левого края порождаемой цепочки. Очевидно, что правила для такого метода порождения должны иметь вид `A --> alpha B` или `A --> B alpha`, где `alpha` — цепочка, состоящая из терминальных символов. В этом случае, если имеется промежуточная (в процессе порождения) цепочка `X1..Xn A`, то замена в соответствии с правилом `A --> alpha B` даст цепочку `X1..Xn alpha B`. Например, для правил `S --> aaaA`, `A --> abcA` и `A --> bbb`можно задать порождение `S => aaaA => aaaabcA => aaaabcbbb`.

Различают порождающие и распознающие (аналитические) грамматики. *Порождающие* задают правила, с помощью которых можно построить любое слово языка.
Аналитические позволяют по данному слову определить, входит оно в язык или нет.

Аналитическая грамматика задает алгоритм, позволяющий определить, принадлежит ли данное слово языку. Например, любой регулярный язык может быть распознан при помощи грамматики, задаваемой конечным автоматом, а любая контекстно-свободная грамматика — с помощью автомата со стековой памятью. Если слово принадлежит языку, то такой автомат строит его вывод в явном виде, что позволяет анализировать семантику этого слова.

Этапы трансляции: лексический, синтаксический и семантический анализ.

Лексема (лексическая единица языка) — это структурная единица языка, которая состоит из элементарных символов языка и не содержит в своем составе других структурных единиц языка. Лексемами языков естественного общения являются слова. Лексемами языков программирования являются идентификаторы, константы, ключевые слова языка, знаки операций и разделители. Состав возможных лексем каждого конкретного языка программирования определяется синтаксисом этого языка.
Лексический анализатор (или сканер) — это часть компилятора, которая читает исходную программу и выделяет в ее тексте лексемы входного языка. На вход лексического анализатора поступает текст исходной программы. Результатом работы лексического анализатора является перечень всех найденных в тексте исходной программы лексем. Этот перечень лексем можно представить в виде таблицы, называемой таблицей лексем.

Синтаксический анализ – это процесс, который определяет, принадлежит ли некоторая последовательность лексем языку, порождаемому грамматикой.

Вход синтаксического анализатора – последовательность лексических и таблицы, например, таблица внешних представлений, которые являются выходом лексического анализатора.
Выход синтаксического анализатора – дерево разбора и таблицы, например, таблица идентификаторов и таблица типов, которые являются входом для следующего просмотра компилятора (например, это может быть просмотр, осуществляющий контроль типов).