**Язык** – все множество правильно построенных текстов. Более строго – множество конечных слов (строк, цепочек) над конечным алфавитом, т. е. множество конечных последовательностей элементов алфавита.

Над языками определены операции, позволяющими определять новые языки на основе существующих.

* Конкатенация: слово нового языка есть конкатенация слова первого языка со словом из второго
* Пересечение: слово нового языка обязано присутствовать в обоих старых
* И другие

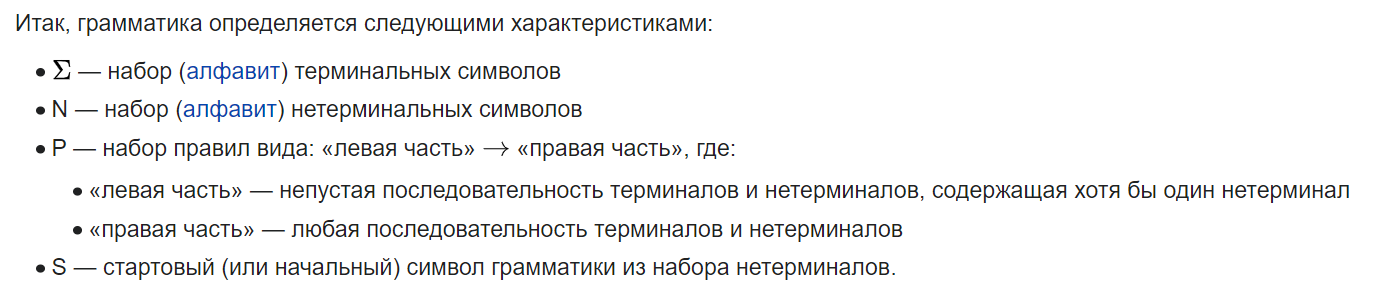
Язык может быть описан как:

1. Перечисление всех элементов (допустимых цепочек)
2. Слова, порожденные формальной грамматики (см ниже)
3. Слова, порожденные регулярным выражением – если язык является регулярным языком
4. Слова, распознаваемые конечным автоматом – если язык является автоматным языком
5. Слова, порожденные формой Бэкуса-Наура (?)

Согласно теореме Клини, множество регулярных языков совпадает с множеством автоматных, то есть определение языка регуляркой эквивалентно определению с помощью некоторого конечного автомата и наоборот.

Для описания языков удобно использовать **формальную грамматику** (или просто грамматику). По сути, она представляет собой набор правил вывода цепочек.

Грамматики бывают порождающие (задают правила порождения слов) и распознающие (аналитические) (по слову определяют, входит ли оно в язык).



Обычно стоит задача не построить все тексты, а распознать, правильно ли построено предложение, а потом желательно еще и разобрать его, построить дерево. Поэтому распознающие грамматики имеют больше приложений.

**Иерархия по Хомскому**. Каждый класс грамматик определяет, в свою очередь, соответствующий класс языков. И чем дальше, чем более богатым и сложным является язык.

* Регулярная (автоматная) – четкие правила, слева – всегда терминальный символ – однозначный разбор, можно использовать конечный автомат (или регулярку, помним про теорему Клини).
* Контекстно-свободная. Хотя нет однозначности, нет и зависимости от контекста (нет необходимости идти вперед) – регулярная грамматика с возможностью перебора. Можно строить дерево разбора. Широко применяются для синтаксиса языков программирования.
* Контекстно-зависимая – в правила входят фрагменты текста. Неплохо описывают предложения естественного языка, в компиляторах не используются из-за сложности. Для таких доказано, что по некоторому алгоритму можно за конечное число шагов определить, относится ли цепочка к языку.
* Рекурсивно-перечислимые (неограниченные). В силу сложности (неспецифицированности) практического применения не имеют.