Модели вычислений

В.А. Захаров

Лекция 7.

1. Формальные грамматики.
2. Иерархия Хомского.
3. Регулярные грамматики
4. Неограниченные грамматики
5. Контекстно-свободные грамматики
6. Нормальная форма Хомского КС-грамматик

Разнообразие формальных языков

конечные автоматы
регулярные языки

Разнообразие формальных языков

Быстрые алгоритмы анализа

Узкий класс языков

конечные автоматы

регулярные языки

Разнообразие формальных языков

машины Тьюринга

рекурсивно перечислимые языки

Быстрые алгоритмы анализа

Узкий класс языков

конечные автоматы

регулярные языки

Разнообразие формальных языков

машины Тьюринга

рекурсивно перечислимые языки

Широкий класс языков

Алгоритмическая неразрешимость задач анализа

Быстрые алгоритмы анализа

Узкий класс языков

конечные автоматы

регулярные языки

Разнообразие формальных языков

машины Тьюринга

рекурсивно перечислимые языки

Широкий класс языков

Алгоритмическая неразрешимость задач анализа

???

Широкий класс языков

Быстрые алгоритмы анализа

Быстрые алгоритмы анализа

Узкий класс языков

конечные автоматы

регулярные языки

Формальные грамматики

В 1957 г. американский лингвист Ноам Хомский (Noam Chomsky) опубликовал книгу «Syntactic Structures», в которой впервые предложил формальный подход к изучению структуры языков на основе строгого определения грамматик, порождающих языковые конструкции — словосочетания, предложения, фразы.

Эта работа радикально изменила взгляды лингвистов на способы описания и изучения естественных языков и оказалась чрезвычайно важной для создания и использования искусственных языков — программирования, описания данных, разметки текстов и пр.

1928 

AVRAM NOAM CHOMSKY

Формальные грамматики

Идея Хомского.

Как описать множество грамматически правильных конструкций языка?

Формальные грамматики

Идея Хомского.

Как описать множество грамматически правильных конструкций языка?

1. Нужно ввести грамматические понятия, обозначающие типы языковых конструкций.

Формальные грамматики

Идея Хомского.
Как описать множество грамматически правильных конструкций языка?

1. Нужно ввести грамматические понятия, обозначающие типы языковых конструкций.
2. Нужно определить грамматические правила, раскрывающие содержание одних грамматических понятий в терминах других грамматических понятий и слов описываемого языка.

Формальные грамматики

Идея Хомского.

Как описать множество грамматически правильных конструкций языка?

1. Нужно ввести грамматические понятия, обозначающие типы языковых конструкций.
2. Нужно определить грамматические правила, раскрывающие содержание одних грамматических понятий в терминах других грамматических понятий и слов описываемого языка.

Так возникли формальные грамматики, позволившие создавать математические методы и алгоритмы для решения задач синтаксического анализа и трансляции языков.

Формальные грамматики

Пример.

Грамматические понятия: *sentence*, *noun-group*, *verb-group*, *noun*, *verb*, *adjective*, *adverb*.

ФОРМАЛЬНЫЕ ГРАММАТИКИ

Пример.

Грамматические понятия: *sentence*, *noun-group*, *verb-group*, *noun*, *verb*, *adjective*, *adverb*.

Правила грамматики:

sentence ::= *noun-group verb-group noun-group* .
noun-group ::= *noun* | *adjective noun-group*
verb-group ::= *verb* | *adverb verb-group*
noun ::= **cats** | **bats** | **rats**
verb ::= **eat** | **like**
adjective ::= **black** | **white** | **big** | **small**
adverb ::= **easy** | **hardly** | **often** | **rare**

ФОРМАЛЬНЫЕ ГРАММАТИКИ

Пример.

Грамматические понятия: *sentence*, *noun-group*, *verb-group*, *noun*, *verb*, *adjective*, *adverb*.

Правила грамматики:

sentence ::= *noun-group verb-group noun-group* .
noun-group ::= *noun* | *adjective noun-group*
verb-group ::= *verb* | *adverb verb-group*
noun ::= **cats** | **bats** | **rats**
verb ::= **eat** | **like**
adjective ::= **black** | **white** | **big** | **small**
adverb ::= **easy** | **hardly** | **often** | **rare**

Грамматически правильные предложения:

black big cats often eat small white rats

black black bats often rare like big small cats

Пример.

Грамматический вывод:

sentence

ФОРМАЛЬНЫЕ ГРАММАТИКИ

Пример.

Правила вывода:

sentence ::= *noun-group verb-group noun-group*

Грамматический вывод:

sentence $\Rightarrow$ *noun-group verb-group noun-group*

Пример.

Правила вывода:

$\mathit{verb\text{-}group} ::= \mathit{verb} \mid \mathit{adverb}\ \mathit{verb\text{-}group}$

Грамматический вывод:

$\mathit{sentence} \Rightarrow \mathit{noun\text{-}group}\ \mathit{verb\text{-}group}\ \mathit{noun\text{-}group}$
$\Rightarrow \mathit{noun\text{-}group}\ \mathit{adverb}\ \mathit{verb\text{-}group}\ \mathit{noun\text{-}group}$

ФОРМАЛЬНЫЕ ГРАММАТИКИ

Пример.

Правила вывода:

$$\textit{noun-group} ::= \textit{noun} \mid \textit{adjective noun-group}$$

Грамматический вывод:

$\textit{sentence} \Rightarrow \textit{noun-group verb-group noun-group}$
$\Rightarrow \textit{noun-group adverb verb-group noun-group}$
$\Rightarrow \textit{adjective noun-group adverb verb-group noun-group}$

ФОРМАЛЬНЫЕ ГРАММАТИКИ

Пример.

Правила вывода:

noun-group ::= *noun* | *adjective noun-group*

Грамматический вывод:

sentence $\Rightarrow$ *noun-group verb-group noun-group*
$\Rightarrow$ *noun-group adverb verb-group noun-group*
$\Rightarrow$ *adjective noun-group adverb verb-group noun-group*
$\Rightarrow$ *adjective noun-group adverb verb-group adjective noun-group*

ФОРМАЛЬНЫЕ ГРАММАТИКИ

Пример.

Правила вывода:

verb-group ::= *verb* | *adverb verb-group*

Грамматический вывод:

sentence ⇒ *noun-group verb-group noun-group*
⇒ *noun-group adverb verb-group noun-group*
⇒ *adjective noun-group adverb verb-group noun-group*
⇒ *adjective noun-group adverb verb-group adjective noun-group*
⇒ *adjective noun-group adverb verb adjective noun-group*

ФОРМАЛЬНЫЕ ГРАММАТИКИ

Пример.

Правила вывода:

noun-group ::= *noun* | *adjective noun-group*

Грамматический вывод:

sentence ⇒ *noun-group verb-group noun-group*
⇒ *noun-group adverb verb-group noun-group*
⇒ *adjective noun-group adverb verb-group noun-group*
⇒ *adjective noun-group adverb verb-group adjective noun-group*
⇒ *adjective noun-group adverb verb adjective noun-group*
⇒ *adjective noun adverb verb adjective noun-group*

ФОРМАЛЬНЫЕ ГРАММАТИКИ

Пример.

Правила вывода:

$$\textit{noun-group} ::= \textit{noun} \mid \textit{adjective noun-group}$$

Грамматический вывод:

sentence ⇒ *noun-group verb-group noun-group*
⇒ *noun-group adverb verb-group noun-group*
⇒ *adjective noun-group adverb verb-group noun-group*
⇒ *adjective noun-group adverb verb-group adjective noun-group*
⇒ *adjective noun-group adverb verb adjective noun-group*
⇒ *adjective noun adverb verb adjective noun-group*
⇒ *adjective noun adverb verb adjective noun*

ФОРМАЛЬНЫЕ ГРАММАТИКИ

Пример.

Правила вывода:

Грамматический вывод:

sentence $\Rightarrow$ *noun-group verb-group noun-group*
$\Rightarrow$ *noun-group adverb verb-group noun-group*
$\Rightarrow$ *adjective noun-group adverb verb-group noun-group*
$\Rightarrow$ *adjective noun-group adverb verb-group adjective noun-group*
$\Rightarrow$ *adjective noun-group adverb verb adjective noun-group*
$\Rightarrow$ *adjective noun adverb verb adjective noun-group*
$\Rightarrow$ *adjective noun adverb verb adjective noun*

Пример (продолжение).

Грамматический вывод:

ФОРМАЛЬНЫЕ ГРАММАТИКИ

Пример (продолжение).

Правила вывода:

$$\mathit{adjective} ::= \textbf{black} \mid \textbf{white} \mid \textbf{big} \mid \textbf{small}$$

Грамматический вывод:

$$\Rightarrow \textbf{small}\ \mathit{noun}\ \mathit{adverb}\ \mathit{verb}\ \mathit{adjective}\ \mathit{noun}$$

ФОРМАЛЬНЫЕ ГРАММАТИКИ

Пример (продолжение).

Правила вывода:

$$\mathit{noun} ::= \textbf{cats} \mid \textbf{bats} \mid \textbf{rats}$$

Грамматический вывод:

$$\Rightarrow \textbf{small}\ \mathit{noun}\ \mathit{adverb}\ \mathit{verb}\ \mathit{adjective}\ \mathit{noun}$$
$$\Rightarrow \textbf{small}\ \textbf{rats}\ \mathit{adverb}\ \mathit{verb}\ \mathit{adjective}\ \mathit{noun}$$

Пример (продолжение).

Правила вывода:

$\mathit{adverb} ::= \textbf{easy} \mid \textbf{hardly} \mid \textbf{often} \mid \textbf{rare}$

Грамматический вывод:

$\Rightarrow$ **small** *noun adverb verb adjective noun*
$\Rightarrow$ **small rats** *adverb verb adjective noun*
$\Rightarrow$ **small rats often** *verb adjective noun*

Пример (продолжение).

Правила вывода:

$verb$ ::= **eat** | **like**

Грамматический вывод:

$\Rightarrow$ **small** *noun adverb verb adjective noun*
$\Rightarrow$ **small rats** *adverb verb adjective noun*
$\Rightarrow$ **small rats often** *verb adjective noun*
$\Rightarrow$ **small rats often eat** *adjective noun*

Пример (продолжение).

Правила вывода:

$\mathit{adjective} ::= \textbf{black} \mid \textbf{white} \mid \textbf{big} \mid \textbf{small}$

Грамматический вывод:

$\Rightarrow$ **small** *noun adverb verb adjective noun*
$\Rightarrow$ **small rats** *adverb verb adjective noun*
$\Rightarrow$ **small rats often** *verb adjective noun*
$\Rightarrow$ **small rats often eat** *adjective noun*
$\Rightarrow$ **small rats often eat big** *noun*

Формальные грамматики

Пример (продолжение).

Правила вывода:

$$\textit{noun} ::= \textbf{cats} \mid \textbf{bats} \mid \textbf{rats}$$

Грамматический вывод:

$\Rightarrow$ **small** *noun adverb verb adjective noun*
$\Rightarrow$ **small rats** *adverb verb adjective noun*
$\Rightarrow$ **small rats often** *verb adjective noun*
$\Rightarrow$ **small rats often eat** *adjective noun*
$\Rightarrow$ **small rats often eat big** *noun*
$\Rightarrow$ **small rats often eat big cats**

Пример (продолжение).

Правила вывода:

Грамматический вывод:

$\Rightarrow$ **small** *noun adverb verb adjective noun*
$\Rightarrow$ **small rats** *adverb verb adjective noun*
$\Rightarrow$ **small rats often** *verb adjective noun*
$\Rightarrow$ **small rats often eat** *adjective noun*
$\Rightarrow$ **small rats often eat big** *noun*
$\Rightarrow$ **small rats often eat big cats**

Итак,

$$\mathit{sentence} \Rightarrow_* \textbf{small rats often eat big cats}$$

ФОРМАЛЬНЫЕ ГРАММАТИКИ

Порождающая грамматика — это система $G = (\Sigma, \mathcal{N}, P, S)$, состоящая из

- конечного алфавита Σ терминальных букв (терминалов),
- конечного алфавита $\mathcal{N}$ нетерминальных букв (нетерминалов), $\Sigma \cap \mathcal{N} = \varnothing$,
- конечного множества P грамматических правил (продукций) вида $\alpha \to \beta$, где $\alpha \in (\Sigma \cup \mathcal{N})^* \mathcal{N} (\Sigma \cup \mathcal{N})^*, \beta \in (\Sigma \cup \mathcal{N})^*$,
- начального нетерминала S, $S \in \mathcal{N}$.

ФОРМАЛЬНЫЕ ГРАММАТИКИ

В приведенном примере описана грамматика, в которой

$\Sigma = \{$ a, b, c, . . . , x, y, z, . $\}$

$\mathcal{N} = \{$ *sentence*, *noun-group*, *verb-group*, *noun*, *verb*, *adjective*, *adverb* $\}$

ФОРМАЛЬНЫЕ ГРАММАТИКИ

В приведенном примере описана грамматика, в которой

$\Sigma = \{$ a, b, c, ..., x, y, z, . $\}$

$\mathcal{N} = \{$ *sentence*, *noun-group*, *verb-group*, *noun*, *verb*, *adjective*, *adverb* $\}$

P состоит из правил

sentence $\to$ *noun-group verb-group noun-group* .

noun-group $\to$ *noun*

noun-group $\to$ *adjective noun-group*

...

adverb $\to$ **rare**

ФОРМАЛЬНЫЕ ГРАММАТИКИ

В приведенном примере описана грамматика, в которой

$\Sigma = \{$ a, b, c, ..., x, y, z, . $\}$

$\mathcal{N} = \{$ *sentence*, *noun-group*, *verb-group*, *noun*, *verb*, *adjective*, *adverb* $\}$

P состоит из правил

sentence $\to$ *noun-group verb-group noun-group* .

noun-group $\to$ *noun*

noun-group $\to$ *adjective noun-group*

...

adverb $\to$ **rare**

Начальный нетерминал S — это *sentence*

ФОРМАЛЬНЫЕ ГРАММАТИКИ

Пример (более абстрактный).

Грамматика $G = (\{a, b, c\}, \{S, B, C, H\}, P, S)$, где множество P состоит из правил

$$S \to aSBC$$
$$S \to aBC$$
$$CB \to HB$$
$$HB \to HC$$
$$HC \to BC$$
$$aB \to ab$$
$$bB \to bb$$
$$bC \to bc$$
$$cC \to cc$$

Формальные грамматики

Грамматическое правило $r : \alpha \to \beta$ определяет отношение непосредственной выводимости слов $\xrightarrow{r}$: для любой пары слов $\gamma', \gamma'' \in (\Sigma \cup \mathcal{N})^*$

$$\gamma' \xrightarrow{r} \gamma'' \iff \gamma' = \theta\alpha\eta \ \wedge \ \gamma'' = \theta\beta\eta.$$

ФОРМАЛЬНЫЕ ГРАММАТИКИ

Грамматическое правило $r : \alpha \to \beta$ определяет отношение непосредственной выводимости слов $\xrightarrow{r}$: для любой пары слов $\gamma', \gamma'' \in (\Sigma \cup \mathcal{N})^*$

$$\gamma' \xrightarrow{r} \gamma'' \iff \gamma' = \theta\alpha\eta \ \wedge \ \gamma'' = \theta\beta\eta.$$

Для грамматики $G = (\Sigma, \mathcal{N}, P, S)$ отношение непосредственной выводимости $\xrightarrow{G}$ — это объединение отношений непосредственной выводимости для правил грамматики:

$$\xrightarrow{G} = \bigcup_{r \in P} \xrightarrow{r} .$$

Формальные грамматики

Грамматическое правило $r : \alpha \to \beta$ определяет отношение непосредственной выводимости слов $\xrightarrow{r}$: для любой пары слов $\gamma', \gamma'' \in (\Sigma \cup \mathcal{N})^*$

$$\gamma' \xrightarrow{r} \gamma'' \iff \gamma' = \theta\alpha\eta \ \wedge \ \gamma'' = \theta\beta\eta.$$

Для грамматики $G = (\Sigma, \mathcal{N}, P, S)$ отношение непосредственной выводимости $\xrightarrow{G}$ — это объединение отношений непосредственной выводимости для правил грамматики:

$$\xrightarrow{G} = \bigcup_{r \in P} \xrightarrow{r} .$$

Рефлексивно-транзитивное замыкание отношения $\xrightarrow{G}$ обозначим записью $\xrightarrow{G}_* .$

ФОРМАЛЬНЫЕ ГРАММАТИКИ

Грамматическим выводом в грамматике G называется всякая конечная последовательность слов в алфавите $\Sigma \cup \mathcal{N}$, связанных отношением непосредственной выводимости:

$$\gamma_1 \xrightarrow{G} \gamma_2 \xrightarrow{G} \gamma_3 \xrightarrow{G} \cdots \xrightarrow{G} \gamma_{k-1} \xrightarrow{G} \gamma_k,$$

Слово γ'' выводимо в грамматике G из слова γ', если $\gamma' \xrightarrow{G}_* \gamma''$.

ФОРМАЛЬНЫЕ ГРАММАТИКИ

Грамматическим выводом в грамматике G называется всякая конечная последовательность слов в алфавите $\Sigma \cup \mathcal{N}$, связанных отношением непосредственной выводимости:

$$\gamma_1 \xrightarrow{G} \gamma_2 \xrightarrow{G} \gamma_3 \xrightarrow{G} \cdots \xrightarrow{G} \gamma_{k-1} \xrightarrow{G} \gamma_k,$$

Слово γ'' выводимо в грамматике G из слова γ', если $\gamma' \xrightarrow{G}_* \gamma''$.

Язык, порождаемый грамматикой G — это множество $L(G) = \{w : w \in \Sigma^*, S \xrightarrow{G}_* w\}$ терминальных слов, выводимых из стартового нетерминала S.

ФОРМАЛЬНЫЕ ГРАММАТИКИ

Пример.

Рассмотрим грамматику $G = (\{a, b\}, \{S\}, P, S)$, где множество P состоит из правил

$S \to aSb$

$S \to ba$

$S \to SS$

$S \to \varepsilon$

ФОРМАЛЬНЫЕ ГРАММАТИКИ

Пример.

Рассмотрим грамматику $G = (\{a, b\}, \{S\}, P, S)$, где множество P состоит из правил

$$S \to aSb$$

$$S \to ba$$

$$S \to SS$$

$$S \to \varepsilon$$

Тогда грамматическим выводом является последовательность

S

ФОРМАЛЬНЫЕ ГРАММАТИКИ

Пример.

Рассмотрим грамматику $G = (\{a, b\}, \{S\}, P, S)$, где множество P состоит из правил

$$S \to aSb$$

$$S \to ba$$

$$S \to SS$$

$$S \to \varepsilon$$

Тогда грамматическим выводом является последовательность

$$S \xrightarrow{G} SS$$

Пример.

Рассмотрим грамматику $G = (\{a, b\}, \{S\}, P, S)$, где множество P состоит из правил

$$S \to aSb$$

$$S \to ba$$

$$S \to SS$$

$$S \to \varepsilon$$

Тогда грамматическим выводом является последовательность

$$S \xrightarrow{G} SS \xrightarrow{G} aSbS$$

Пример.

Рассмотрим грамматику $G = (\{a, b\}, \{S\}, P, S)$, где множество P состоит из правил

$$S \to aSb$$

$$S \to ba$$

$$S \to SS$$

$$S \to \varepsilon$$

Тогда грамматическим выводом является последовательность

$$S \xrightarrow{G} SS \xrightarrow{G} aSbS \xrightarrow{G} aaSbbS$$

Формальные грамматики

Пример.

Рассмотрим грамматику $G = (\{a, b\}, \{S\}, P, S)$, где множество P состоит из правил

$$S \to aSb$$

$$S \to ba$$

$$S \to SS$$

$$S \to \varepsilon$$

Тогда грамматическим выводом является последовательность

$$S \xrightarrow{G} SS \xrightarrow{G} aSbS \xrightarrow{G} aaSbbS \xrightarrow{G} aabbS$$

Пример.

Рассмотрим грамматику $G = (\{a, b\}, \{S\}, P, S)$, где множество P состоит из правил

$$S \to aSb$$

$$S \to ba$$

$$S \to SS$$

$$S \to \varepsilon$$

Тогда грамматическим выводом является последовательность

$$S \xrightarrow{G} SS \xrightarrow{G} aSbS \xrightarrow{G} aaSbbS \xrightarrow{G} aabbS \xrightarrow{G} aabbba$$

Пример.

Рассмотрим грамматику $G = (\{a, b\}, \{S\}, P, S)$, где множество P состоит из правил

$$S \to aSb$$
$$S \to ba$$
$$S \to SS$$
$$S \to \varepsilon$$

Тогда грамматическим выводом является последовательность

$$S \xrightarrow{G} SS \xrightarrow{G} aSbS \xrightarrow{G} aaSbbS \xrightarrow{G} aabbS \xrightarrow{G} aabbba$$

Таким образом, $aabbba \in L(G)$.

ФОРМАЛЬНЫЕ ГРАММАТИКИ

Пример.

Грамматика $G = (\{a, b, c\}, \{S, B, C, H\}, P, S)$, где множество P состоит из правил

$$S \to aSBC$$

$$S \to aBC$$

$$CB \to HB$$

$$HB \to HC$$

$$HC \to BC$$

$$aB \to ab$$

$$bB \to bb$$

$$bC \to bc$$

$$cC \to cc$$

Каков язык $L(G)$ этой грамматики?

Пример.

Грамматика $G = (\{a, b, c\}, \{S, B, C, H\}, P, S)$, где множество P состоит из правил

$$S \to aSBC$$
$$S \to aBC$$
$$CB \to HB$$
$$HB \to HC$$
$$HC \to BC$$
$$aB \to ab$$
$$bB \to bb$$
$$bC \to bc$$
$$cC \to cc$$

Каков язык $L(G)$ этой грамматики?

$L(G) = \{a^n b^n c^n : n \geq 1\}$.

Хомский предложил следующую классификацию формальных грамматик в зависимости от того, какие правила разрешается использовать для грамматического вывода. Эта классификация формальных грамматик и языков получила название

Иерархия Хомского

Иерархия Хомского

Грамматика $G = (\Sigma, \mathcal{N}, P, S)$ называется

- неограниченной (типа 0), если в ней разрешены любые грамматические правила;

Иерархия Хомского

Грамматика $G = (\Sigma, \mathcal{N}, P, S)$ называется

- неограниченной (типа 0), если в ней разрешены любые грамматические правила;
- контекстно-зависимой (типа 1), если в ней разрешены только грамматические правила вида $\theta A \eta \to \theta \alpha \eta$, где $A \in \mathcal{N}$, $\alpha \in (\Sigma \cup \mathcal{N})^+$, $\theta, \eta \in (\Sigma \cup \mathcal{N})^*$;

Иерархия Хомского

Грамматика $G = (\Sigma, \mathcal{N}, P, S)$ называется

- неограниченной (типа 0), если в ней разрешены любые грамматические правила;
- контекстно-зависимой (типа 1), если в ней разрешены только грамматические правила вида $\theta A \eta \to \theta \alpha \eta$, где $A \in \mathcal{N}$, $\alpha \in (\Sigma \cup \mathcal{N})^+$, $\theta, \eta \in (\Sigma \cup \mathcal{N})^*$;
- контекстно-свободной (типа 2), если в ней разрешены только грамматические правила вида $A \to \alpha$, где $A \in \mathcal{N}$, $\alpha \in (\Sigma \cup \mathcal{N})^*$;

Иерархия Хомского

Грамматика $G = (\Sigma, \mathcal{N}, P, S)$ называется

- неограниченной (типа 0), если в ней разрешены любые грамматические правила;
- контекстно-зависимой (типа 1), если в ней разрешены только грамматические правила вида $\theta A\eta \to \theta\alpha\eta$, где $A \in \mathcal{N}$, $\alpha \in (\Sigma \cup \mathcal{N})^+$, $\theta, \eta \in (\Sigma \cup \mathcal{N})^*$;
- контекстно-свободной (типа 2), если в ней разрешены только грамматические правила вида $A \to \alpha$, где $A \in \mathcal{N}$, $\alpha \in (\Sigma \cup \mathcal{N})^*$;
- регулярные (типа 3), если в ней разрешены только грамматические правила вида $A \to wB$ или $A \to w$, где $A, B \in \mathcal{N}$, $w \in \Sigma^*$.

Иерархия Хомского

Пример. Неограниченная грамматика, моделирующая вычисление машины Тьюринга:

$\Sigma = \{0, 1, \$, \#\}$, $\mathcal{N} = \{S, Q_1, Q_2\}$.

$S \to \$Q_1 0\#$,

$0Q_1 0 \to Q_2 01$, $\quad 1Q_1 0 \to Q_2 11$,

$Q_1 10 \to 0Q_1 0$, $\quad Q_1 11 \to 0Q_1 1$,

$Q_2 00 \to 0Q_1 0$, $\quad Q_2 01 \to 1Q_1 1$, $\quad Q_2 1 \to 0$,

$\$Q_1 \to \$0Q_1$ $\quad \$Q_2 \to \$0Q_2$

$Q_1\# \to Q_1 0\#$ $\quad Q_2\# \to Q_2 0\#$

Иерархия Хомского

Пример. Неограниченная грамматика, моделирующая вычисление машины Тьюринга:

$\Sigma = \{0, 1, \$, \#\}$, $\mathcal{N} = \{S, Q_1, Q_2\}$.

$S \to \$Q_1 0\#$,

$0Q_1 0 \to Q_2 01$, $\quad 1Q_1 0 \to Q_2 11$,

$Q_1 10 \to 0Q_1 0$, $\quad Q_1 11 \to 0Q_1 1$,

$Q_2 00 \to 0Q_1 0$, $\quad Q_2 01 \to 1Q_1 1$, $\quad Q_2 1 \to 0$,

$\$Q_1 \to \$0Q_1$ $\quad \$Q_2 \to \$0Q_2$

$Q_1\# \to Q_1 0\#$ $\quad Q_2\# \to Q_2 0\#$

S

ИЕРАРХИЯ ХОМСКОГО

Пример. Неограниченная грамматика, моделирующая вычисление машины Тьюринга:

$\Sigma = \{0, 1, \$, \#\}$, $\mathcal{N} = \{S, Q_1, Q_2\}$.

$S \to \$Q_1 0\#,$

$0Q_1 0 \to Q_2 01,$ $\quad 1Q_1 0 \to Q_2 11,$

$Q_1 10 \to 0Q_1 0,$ $\quad Q_1 11 \to 0Q_1 1,$

$Q_2 00 \to 0Q_1 0,$ $\quad Q_2 01 \to 1Q_1 1,$ $\quad Q_2 1 \to 0,$

$\$Q_1 \to \$0Q_1$ $\quad \$Q_2 \to \$0Q_2$

$Q_1\# \to Q_1 0\#$ $\quad Q_2\# \to Q_2 0\#$

$S \longrightarrow \$Q_1 0\#$

Иерархия Хомского

Пример. Неограниченная грамматика, моделирующая вычисление машины Тьюринга:

$\Sigma = \{0, 1, \$, \#\}$, $\mathcal{N} = \{S, Q_1, Q_2\}$.

$S \to \$Q_1 0\#,$

$0Q_1 0 \to Q_2 01,$ $\quad$ $1Q_1 0 \to Q_2 11,$

$Q_1 10 \to 0Q_1 0,$ $\quad$ $Q_1 11 \to 0Q_1 1,$

$Q_2 00 \to 0Q_1 0,$ $\quad$ $Q_2 01 \to 1Q_1 1,$ $\quad$ $Q_2 1 \to 0,$

$\$Q_1 \to \$0Q_1$ $\quad$ $\$Q_2 \to \$0Q_2$

$Q_1\# \to Q_1 0\#$ $\quad$ $Q_2\# \to Q_2 0\#$

$S \longrightarrow \$Q_1 0\# \longrightarrow \$0Q_1 0\#$

Иерархия Хомского

Пример. Неограниченная грамматика, моделирующая вычисление машины Тьюринга:

$\Sigma = \{0, 1, \$, \#\}$, $\mathcal{N} = \{S, Q_1, Q_2\}$.

$S \to \$Q_1 0\#,$

$0Q_1 0 \to Q_2 01,$ $\quad 1Q_1 0 \to Q_2 11,$

$Q_1 10 \to 0Q_1 0,$ $\quad Q_1 11 \to 0Q_1 1,$

$Q_2 00 \to 0Q_1 0,$ $\quad Q_2 01 \to 1Q_1 1,$ $\quad Q_2 1 \to 0,$

$\$Q_1 \to \$0Q_1$ $\quad \$Q_2 \to \$0Q_2$

$Q_1\# \to Q_1 0\#$ $\quad Q_2\# \to Q_2 0\#$

$S \longrightarrow \$Q_1 0\# \longrightarrow \$0Q_1 0\# \longrightarrow \$Q_2 01\#$

Иерархия Хомского

Пример. Неограниченная грамматика, моделирующая вычисление машины Тьюринга:

$\Sigma = \{0, 1, \$, \#\}$, $\mathcal{N} = \{S, Q_1, Q_2\}$.

$S \to \$Q_1 0\#$,

$0Q_1 0 \to Q_2 01$, $\quad 1Q_1 0 \to Q_2 11$,

$Q_1 10 \to 0Q_1 0$, $\quad Q_1 11 \to 0Q_1 1$,

$Q_2 00 \to 0Q_1 0$, $\quad Q_2 01 \to 1Q_1 1$, $\quad Q_2 1 \to 0$,

$\$Q_1 \to \$0Q_1$ $\quad \$Q_2 \to \$0Q_2$

$Q_1\# \to Q_1 0\#$ $\quad Q_2\# \to Q_2 0\#$

$S \longrightarrow \$Q_1 0\# \longrightarrow \$0Q_1 0\# \longrightarrow \$Q_2 01\# \longrightarrow \$1Q_1 1\#$

Иерархия Хомского

Пример. Контекстно-зависимая грамматика.

$\Sigma = \{a, b, c\}$, $\mathcal{N} = \{S, B, C, H\}$,

$$S \to aSBC$$
$$S \to aBC$$
$$CB \to HB$$
$$HB \to HC$$
$$HC \to BC$$
$$aB \to ab$$
$$bB \to bb$$
$$bC \to bc$$
$$cC \to cc$$

Иерархия Хомского

Пример. Контекстно-свободная грамматика.

$\Sigma = \{x, y, 0, 1, +, \times, (,)\}$, $\mathcal{N} = \{S, V, C, A\}$,

$$S \to V$$
$$S \to C$$
$$S \to (SAS)$$
$$V \to x$$
$$V \to y$$
$$C \to 0$$
$$C \to 1$$
$$A \to +$$
$$A \to \times$$

Иерархия Хомского

Пример. Регулярная грамматика, моделирующая конечный автомат.

$\Sigma = \{a, b\}$, $\mathcal{N} = \{S, Q_1, Q_2, Q_3\}$,

$$
\begin{aligned}
S &\to aQ_1 \\
S &\to aS \\
S &\to bQ_2 \\
Q_1 &\to aQ_2 \\
Q_1 &\to aS \\
Q_2 &\to bQ_1 \\
Q_2 &\to \varepsilon
\end{aligned}
$$

Иерархия Хомского

Утверждение 7.1.

1. Каждая регулярная грамматика является контекстно-свободной,
2. Каждая контекстно-свободная грамматика без правил вида $A \to \varepsilon$ является контекстно-зависимой,
3. Каждая контекстно-зависимая грамматика является неограниченной.

Иерархия Хомского

Утверждение 7.1.

1. Каждая регулярная грамматика является контекстно-свободной,
2. Каждая контекстно-свободная грамматика без правил вида $A \to \varepsilon$ является контекстно-зависимой,
3. Каждая контекстно-зависимая грамматика является неограниченной.

А теперь разберемся с грамматиками каждого типа по отдельности.

РЕГУЛЯРНЫЕ ГРАММАТИКИ

Регулярные грамматики состоят из правил вида $A \to wB$ или $A \to w$, где $A, B \in \mathcal{N}$, $w \in \Sigma^*$.

Поскольку в правой части каждого правила присутствует не более одного нетерминала, который может содержаться только в правом конце слова, такие грамматики также называются праволинейными.

Зададимся вопросом:
какие языки порождаются регулярными (праволинейными) грамматиками?

Регулярные грамматики

Пусть задана грамматика $G = (\Sigma, \mathcal{N}, P, S)$ и нетерминал N, $N \in \mathcal{N}$.

Языком нетерминала N, $N \in \mathcal{N}$, в грамматике G называется множество терминальных слов $L_G(N) = \{w : w \in \Sigma^*, N \xrightarrow{G}_* w\}$, выводимых из этого нетерминала. В частности, $L(G) = L_G(S)$.

Регулярные грамматики

Пусть задана грамматика $G = (\Sigma, \mathcal{N}, P, S)$ и нетерминал N, $N \in \mathcal{N}$.

Языком нетерминала N, $N \in \mathcal{N}$, в грамматике G называется множество терминальных слов $L_G(N) = \{w : w \in \Sigma^*, N \xrightarrow{G}_* w\}$, выводимых из этого нетерминала. В частности, $L(G) = L_G(S)$.

Утверждение 7.2. Каждый язык, порождаемый регулярной грамматикой, является регулярным.

Доказательство. Проведем для случая, когда в грамматике нет правил $N \to \varepsilon$, в правой части которых стоит пустое слово.

РЕГУЛЯРНЫЕ ГРАММАТИКИ

Рассмотрим произвольный нетерминал $N, \ N \in \mathcal{N}$, и все грамматические правила $N \to w_i A_i, \ 1 \le i \le k$, и $N \to u_j, \ 1 \le j \le m$.

РЕГУЛЯРНЫЕ ГРАММАТИКИ

Рассмотрим произвольный нетерминал N, $N \in \mathcal{N}$, и все грамматические правила $N \to w_i A_i$, $1 \le i \le k$, и $N \to u_j$, $1 \le j \le m$.

Легко видеть, что справедливо равенство

$$L_G(N) = \bigcup_{i=1}^{k} w_i L_G(A_i) \cup \{u_1, \ldots, u_m\}.$$

РЕГУЛЯРНЫЕ ГРАММАТИКИ

Рассмотрим произвольный нетерминал N, $N \in \mathcal{N}$, и все грамматические правила $N \to w_i A_i$, $1 \le i \le k$, и $N \to u_j$, $1 \le j \le m$.

Легко видеть, что справедливо равенство

$$L_G(N) = \bigcup_{i=1}^{k} w_i L_G(A_i) \cup \{u_1, \ldots, u_m\}.$$

Таким образом семейство языков $L_G(N)$, $N \in \mathcal{N}$, является решением системы линейных уравнений над языками, в которой все уравнения имеют вид:

$$X_N = \sum_{j=1}^{k} w_i X_{A_i} + u_1 + \cdots + u_m.$$

Но, как следует из утверждения 3.3 (Лекция 3), эта система уравнений имеет единственное решение, и этим решением является набор регулярных языков. QED

Регулярные грамматики

Задача 1. Какое значение для предложенного доказательства имеет допущение об отсутствии в грамматике правил с пустой правой частью?

Проведите доказательство утверждения 7.2 для регулярных грамматик произвольного вида.

Регулярные грамматики

Задача 1. Какое значение для предложенного доказательства имеет допущение об отсутствии в грамматике правил с пустой правой частью?

Проведите доказательство утверждения 7.2 для регулярных грамматик произвольного вида.

Утверждение 7.3. Каждый регулярный язык порождается регулярной грамматикой.

Доказательство. Поскольку класс регулярных языков совпадает с классом автоматных языков, достаточно показать, что каждый автоматный язык порождается регулярной грамматикой.

Регулярные грамматики

Пусть $L = L(\mathcal{A})$, где $\mathcal{A} = (\Sigma, Q, q_0, F, T)$ — детеминированный конечный автомат.

Рассмотрим регулярную грамматику $G = (\Sigma, Q, P, q_0)$, в которой в роли нетерминалов выступают все состояния $q_0, q_1, \ldots, q_n$ автомата $\mathcal{A}$, а множество грамматических правил для каждого перехода $(q', a, q'') \in T$ автомата $\mathcal{A}$ содержит правило $q' \to aq''$, и для каждого финального состояния $q,\ q \in F$, содержит правило $q \to \varepsilon$.

РЕГУЛЯРНЫЕ ГРАММАТИКИ

Пусть $L = L(\mathcal{A})$, где $\mathcal{A} = (\Sigma, Q, q_0, F, T)$ — детеминированный конечный автомат.

Рассмотрим регулярную грамматику $G = (\Sigma, Q, P, q_0)$, в которой в роли нетерминалов выступают все состояния $q_0, q_1, \ldots, q_n$ автомата $\mathcal{A}$, а множество грамматических правил для каждого перехода $(q', a, q'') \in T$ автомата $\mathcal{A}$ содержит правило $q' \to aq''$, и для каждого финального состояния $q,\ q \in F$, содержит правило $q \to \varepsilon$.

Далее индукцией по длине слова $w = a_1 a_2 \ldots a_n$ нетрудно показать, что автомат $\mathcal{A}$ имеет успешное вычисление

$$q_0 \xrightarrow{a_1} q_1 \xrightarrow{a_2} \cdots \xrightarrow{a_n} q_n$$

тогда и только тогда в грамматике $G = (\Sigma, Q, P, q_0)$ существует вывод

$$q_0 \xrightarrow{G} a_1 q_1 \xrightarrow{G} \cdots \xrightarrow{G_n} a_1 a_2 \ldots a_n q_n \xrightarrow{G_n} a_1 a_2 \ldots a_n.$$

QED

РЕГУЛЯРНЫЕ ГРАММАТИКИ

Таким образом, мы доказали, что справедлива

Теорема 7.4. Класс регулярных языков совпадает с классом языков, порождаемых регулярными грамматиками.

Этим и объясняется такое название рассмотренных грамматик.

НЕОГРАНИЧЕННЫЕ ГРАММАТИКИ

Неограниченные грамматики по своим порождающим возможностям равносильны машинам Тьюринга.

Утверждение 7.5. Для любой неограниченной грамматики G существует недетерминированная двухленточная МТ $\mathcal{M}$, порождающая все слова языка $L(G)$ и только эти слова.

Доказательство. Нужная МТ $\mathcal{M}$ работает так: записывает на ленте начальный нетерминал S и моделирует вывод по правилам грамматики G, чередуя ленты для записи слов в выводе. Как только будет получено терминальное слово, МТ $\mathcal{M}$ останавливается и записывает его как результат на выходной ленте. QED

НЕОГРАНИЧЕННЫЕ ГРАММАТИКИ

Неограниченные грамматики по своим порождающим возможностям равносильны машинам Тьюринга.

Утверждение 7.5. Для любой неограниченной грамматики G существует недетерминированная двухленточная МТ $\mathcal{M}$, порождающая все слова языка $L(G)$ и только эти слова.

Доказательство. Нужная МТ $\mathcal{M}$ работает так: записывает на ленте начальный нетерминал S и моделирует вывод по правилам грамматики G, чередуя ленты для записи слов в выводе. Как только будет получено терминальное слово, МТ $\mathcal{M}$ останавливается и записывает его как результат на выходной ленте. QED

Можно доказать и большее.

Задача 2. Докажите, что для любой пары алфавитов Σ, Γ существует такая МТ $\mathcal{U}$, что

$$L(\mathcal{U}) = \{w\#P : G = (\Sigma, \Gamma, P, S) \text{ — грамматика}, \ w \in L(G)\}.$$

Неограниченные грамматики

Утверждение 7.6. Для любого рекурсивно перечислимого языка L существует такая грамматика G, для которой верно $L = L(G)$.

Доказательство. Самостоятельно.

Подсказка: см задачу 4.6 и приведенный ранее пример неограниченной грамматики.

Неограниченные грамматики

Утверждение 7.6. Для любого рекурсивно перечислимого языка L существует такая грамматика G, для которой верно $L = L(G)$.

Доказательство. Самостоятельно.

Подсказка: см задачу 4.6 и приведенный ранее пример неограниченной грамматики.

Теорема 7.7. Язык является рекурсивно перечислимым тогда и только тогда, когда он порождается некоторой формальной грамматикой.

КОНТЕКСТНО-СВОБОДНЫЕ ГРАММАТИКИ

Грамматика $G = (\Sigma, \mathcal{N}, P, S)$ называется контекстно-свободной, если в ней разрешены только грамматические правила вида $A \to \alpha$, где $A \in \mathcal{N}$, $\alpha \in (\Sigma \cup \mathcal{N})^*$;

Пример. $G = (\{a, b\}, \{S\}, P, S)$, где

$$P = \{S \to aSb \mid bSa \mid SS \mid \varepsilon\}.$$

Пример. $G = (\{a, b\}, \{S, A, B\}, P, S)$, где

$$\begin{aligned} P = \{ & S \to aSB \mid bSA \mid aSBS \mid bSAS \mid \varepsilon, \\ & A \to a, \\ & B \to b\} \end{aligned}$$

Контекстно-свободные грамматики

Именно контекстно-свободные грамматики удобно использовать для описания синтаксиса многих искусственнных языков.

$\langle\!\langle$оператор$\rangle\!\rangle \to \langle\!\langle$простой оператор$\rangle\!\rangle \mid$
begin $\langle\!\langle$составной оператор$\rangle\!\rangle$ **end**
$\langle\!\langle$составной оператор$\rangle\!\rangle \to \varepsilon \mid \langle\!\langle$оператор$\rangle\!\rangle$; $\langle\!\langle$составной оператор$\rangle\!\rangle$
$\langle\!\langle$простой оператор$\rangle\!\rangle \to \langle\!\langle$левая часть$\rangle\!\rangle := \langle\!\langle$правая часть$\rangle\!\rangle$
$\langle\!\langle$левая часть$\rangle\!\rangle \to \langle\!\langle$переменная$\rangle\!\rangle$
$\langle\!\langle$правая часть$\rangle\!\rangle \to \langle\!\langle$терм$\rangle\!\rangle$
$\langle\!\langle$переменная$\rangle\!\rangle \to \langle\!\langle$буква$\rangle\!\rangle \langle\!\langle$строка$\rangle\!\rangle$
$\langle\!\langle$буква$\rangle\!\rangle \to$ a $\mid$ b $\mid$ c $\mid$ d $\mid$ e
$\langle\!\langle$строка$\rangle\!\rangle \to \varepsilon \mid \langle\!\langle$буква$\rangle\!\rangle \langle\!\langle$строка$\rangle\!\rangle \mid \langle\!\langle$цифра$\rangle\!\rangle \langle\!\langle$строка$\rangle\!\rangle$
$\langle\!\langle$цифра$\rangle\!\rangle \to 0 \mid 1 \mid 2 \mid 3 \mid 4 \mid 5 \mid 6 \mid 7 \mid 8 \mid 9$

КОНТЕКСТНО-СВОБОДНЫЕ ГРАММАТИКИ

Для КС-грамматик можно ограничиться грамматическими выводами специального вида. Вывод

$$S \xrightarrow{G} \alpha_1 \xrightarrow{G} \cdots \xrightarrow{G} w'N\beta \xrightarrow{G} w'\gamma\beta \xrightarrow{G} \cdots$$

называется левосторонним, если на каждом шаге очередное правило применяется к самому левому нетерминалу N в строке $w'N\beta$.

Утверждение 7.8. Если слово w выводимо в КС-грамматике, то существует левосторонний вывод этого слова

Доказательство. Самостоятельно.

КОНТЕКСТНО-СВОБОДНЫЕ ГРАММАТИКИ

Основная задача, возникающая при работе с искусственными языками, порожденными КС-грамматиками — это задача синтаксического анализа :

для заданного слова $w, w \in \Sigma^*$ **, и заданной КС-грамматики** $G = (\Sigma, \mathcal{N}, P, S)$ **проверить включение** $w \in L(G)$ **.**

Именно эта задача будет находиться в центре нашего внимания.

КОНТЕКСТНО-СВОБОДНЫЕ ГРАММАТИКИ

Вначале научимся упрощать КС-грамматики, избавляясь от тех конструкций, которые затрудняют анализ грамматик. Грамматики G и G' называются эквивалентными, если $L(G) = L(G')$.

КОНТЕКСТНО-СВОБОДНЫЕ ГРАММАТИКИ

Вначале научимся упрощать КС-грамматики, избавляясь от тех конструкций, которые затрудняют анализ грамматик. Грамматики G и G' называются эквивалентными, если $L(G) = L(G')$.

Утверждение 7.9. Для любой КС-грамматики $G = (\Sigma, \mathcal{N}, P, S)$ существует эквивалентная КС-грамматика, у которой правые части правил не содержат начального нетерминала.

КОНТЕКСТНО-СВОБОДНЫЕ ГРАММАТИКИ

Вначале научимся упрощать КС-грамматики, избавляясь от тех конструкций, которые затрудняют анализ грамматик. Грамматики G и G' называются эквивалентными, если $L(G) = L(G')$.

Утверждение 7.9. Для любой КС-грамматики $G = (\Sigma, \mathcal{N}, P, S)$ существует эквивалентная КС-грамматика, у которой правые части правил не содержат начального нетерминала.

Доказательство. Введем новый нетерминал S_0, объявим его начальным и добавим в грамматику правило $S_0 \to S$. QED

КОНТЕКСТНО-СВОБОДНЫЕ ГРАММАТИКИ

Пример

$G:$	
$S \to \varepsilon$ $S \to aSBb$ $B \to A$ $B \to aaSS$ $A \to B$ $A \to \varepsilon$ $A \to bbSA$	

КОНТЕКСТНО-СВОБОДНЫЕ ГРАММАТИКИ

Пример

$G:$	$G':$
	$S_0 \to S$
$S \to \varepsilon$	$S \to \varepsilon$
$S \to aSBb$	$S \to aBb$
$B \to A$	$B \to A$
$B \to aaSS$	$B \to aaSS$
$A \to B$	$A \to B$
$A \to \varepsilon$	$A \to \varepsilon$
$A \to bbSA$	$A \to bbSA$

Контекстно-свободные грамматики

Грамматические правила вида $N' \to N''$, где $N', N'' \in \mathcal{N}$, называются переименованиями.

Утверждение 7.10. Для любой КС-грамматики $G = (\Sigma, \mathcal{N}, P, S)$ существует эквивалентная КС-грамматика, не содержащая правил-переименований.

Доказательство. Для каждого переименования $N' \to N''$ и правила $N'' \to \alpha$ добавим к множеству P новое правило $N' \to \alpha$ (если оно там отсутствовало). Ясно, что последовательное применение правил $N' \to N''$ и $N'' \to \alpha$ равносильно применению правила $N' \to \alpha$.

Как только обнаружится, что новых правил добавить невозможно, удалим все переименования. Поскольку в каждом успешном выводе переименование работает в паре с каким-то правилом, такое удаление не изменяет порождаемого грамматикой языка.

QED

КОНТЕКСТНО-СВОБОДНЫЕ ГРАММАТИКИ

Пример

G_0 :				
$S_0 \to S$				
$S \to \varepsilon$				
$S \to aSBb$				
$B \to A$				
$B \to aaSS$				
$A \to B$				
$A \to \varepsilon$				
$A \to bbSA$				

КОНТЕКСТНО-СВОБОДНЫЕ ГРАММАТИКИ

Пример

G_0 :				
$S_0 \to S$				
$S \to \varepsilon$				
$S \to aSBb$				
$B \to A$				
$B \to aaSS$				
$A \to B$				
$A \to \varepsilon$				
$A \to bbSA$				

КОНТЕКСТНО-СВОБОДНЫЕ ГРАММАТИКИ

Пример

G_0 :	G_1 :			
$S_0 \to S$	$S_0 \to S$			
$S \to \varepsilon$	$S \to \varepsilon$			
$S \to aSBb$	$S \to aSBb$			
$B \to A$	$B \to A$			
$B \to aaSS$	$B \to aaSS$			
$A \to B$	$A \to B$			
$A \to \varepsilon$	$A \to \varepsilon$			
$A \to bbSA$	$A \to bbSA$			
	$S_0 \to \varepsilon$			
	$S_0 \to aSBb$			

КОНТЕКСТНО-СВОБОДНЫЕ ГРАММАТИКИ

Пример

G_0 :	G_1 :			
$S_0 \to S$	$S_0 \to S$			
$S \to \varepsilon$	$S \to \varepsilon$			
$S \to aSBb$	$S \to aSBb$			
$B \to A$	$B \to A$			
$B \to aaSS$	$B \to aaSS$			
$A \to B$	$A \to B$			
$A \to \varepsilon$	$A \to \varepsilon$			
$A \to bbSA$	$A \to bbSA$			
	$S_0 \to \varepsilon$			
	$S_0 \to aSBb$			

КОНТЕКСТНО-СВОБОДНЫЕ ГРАММАТИКИ

Пример

G_0 :	G_1 :	G_2 :		
$S_0 \to S$	$S_0 \to S$	$S_0 \to S$		
$S \to \varepsilon$	$S \to \varepsilon$	$S \to \varepsilon$		
$S \to aSBb$	$S \to aSBb$	$S \to aSBb$		
$B \to A$	$B \to A$	$B \to A$		
$B \to aaSS$	$B \to aaSS$	$B \to aaSS$		
$A \to B$	$A \to B$	$A \to B$		
$A \to \varepsilon$	$A \to \varepsilon$	$A \to \varepsilon$		
$A \to bbSA$	$A \to bbSA$	$A \to bbSA$		
	$S_0 \to \varepsilon$	$S_0 \to \varepsilon$		
	$S_0 \to aSBb$	$S_0 \to aSBb$		
		$B \to B$		
		$B \to \varepsilon$		
		$B \to bbSA$		

КОНТЕКСТНО-СВОБОДНЫЕ ГРАММАТИКИ

Пример

G_0 :	G_1 :	G_2 :		
$S_0 \to S$	$S_0 \to S$	$S_0 \to S$		
$S \to \varepsilon$	$S \to \varepsilon$	$S \to \varepsilon$		
$S \to aSBb$	$S \to aSBb$	$S \to aSBb$		
$B \to A$	$B \to A$	$B \to A$		
$B \to aaSS$	$B \to aaSS$	$B \to aaSS$		
$A \to B$	$A \to B$	$A \to B$		
$A \to \varepsilon$	$A \to \varepsilon$	$A \to \varepsilon$		
$A \to bbSA$	$A \to bbSA$	$A \to bbSA$		
	$S_0 \to \varepsilon$	$S_0 \to \varepsilon$		
	$S_0 \to aSBb$	$S_0 \to aSBb$		
		$B \to B$		
		$B \to \varepsilon$		
		$B \to bbSA$		

КОНТЕКСТНО-СВОБОДНЫЕ ГРАММАТИКИ

Пример

G_0 :	G_1 :	G_2 :	G_3 :
$S_0 \to S$	$S_0 \to S$	$S_0 \to S$	$S_0 \to S$
$S \to \varepsilon$	$S \to \varepsilon$	$S \to \varepsilon$	$S \to \varepsilon$
$S \to aSBb$	$S \to aSBb$	$S \to aSBb$	$S \to aSBb$
$B \to A$	$B \to A$	$B \to A$	$B \to A$
$B \to aaSS$	$B \to aaSS$	$B \to aaSS$	$B \to aaSS$
$A \to B$	$A \to B$	$A \to B$	$A \to B$
$A \to \varepsilon$	$A \to \varepsilon$	$A \to \varepsilon$	$A \to \varepsilon$
$A \to bbSA$	$A \to bbSA$	$A \to bbSA$	$A \to bbSA$
	$S_0 \to \varepsilon$	$S_0 \to \varepsilon$	$S_0 \to \varepsilon$
	$S_0 \to aSBb$	$S_0 \to aSBb$	$S_0 \to aSBb$
		$B \to B$	$B \to B$
		$B \to \varepsilon$	$B \to \varepsilon$
		$B \to bbSA$	$B \to bbSA$
			$A \to A$
			$A \to aaSS$

КОНТЕКСТНО-СВОБОДНЫЕ ГРАММАТИКИ

Пример

G_0 :	G_1 :	G_2 :	G_3 :
$S_0 \to S$	$S_0 \to S$	$S_0 \to S$	$S_0 \to S$
$S \to \varepsilon$	$S \to \varepsilon$	$S \to \varepsilon$	$S \to \varepsilon$
$S \to aSBb$	$S \to aSBb$	$S \to aSBb$	$S \to aSBb$
$B \to A$	$B \to A$	$B \to A$	$B \to A$
$B \to aaSS$	$B \to aaSS$	$B \to aaSS$	$B \to aaSS$
$A \to B$	$A \to B$	$A \to B$	$A \to B$
$A \to \varepsilon$	$A \to \varepsilon$	$A \to \varepsilon$	$A \to \varepsilon$
$A \to bbSA$	$A \to bbSA$	$A \to bbSA$	$A \to bbSA$
	$S_0 \to \varepsilon$	$S_0 \to \varepsilon$	$S_0 \to \varepsilon$
	$S_0 \to aSBb$	$S_0 \to aSBb$	$S_0 \to aSBb$
		$B \to B$	$B \to B$
		$B \to \varepsilon$	$B \to \varepsilon$
		$B \to bbSA$	$B \to bbSA$
			$A \to A$
			$A \to aaSS$

КОНТЕКСТНО-СВОБОДНЫЕ ГРАММАТИКИ

Пример

G_0 :	G_1 :	G_2 :	G_3 :	G_4
$S_0 \to S$	$S_0 \to S$	$S_0 \to S$	$S_0 \to S$	
$S \to \varepsilon$	$S \to \varepsilon$	$S \to \varepsilon$	$S \to \varepsilon$	$S \to \varepsilon$
$S \to aSBb$	$S \to aSBb$	$S \to aSBb$	$S \to aSBb$	$S \to aSBb$
$B \to A$	$B \to A$	$B \to A$	$B \to A$	
$B \to aaSS$	$B \to aaSS$	$B \to aaSS$	$B \to aaSS$	$B \to aaSS$
$A \to B$	$A \to B$	$A \to B$	$A \to B$	
$A \to \varepsilon$	$A \to \varepsilon$	$A \to \varepsilon$	$A \to \varepsilon$	$A \to \varepsilon$
$A \to bbSA$	$A \to bbSA$	$A \to bbSA$	$A \to bbSA$	$A \to bbSA$
	$S_0 \to \varepsilon$	$S_0 \to \varepsilon$	$S_0 \to \varepsilon$	$S_0 \to \varepsilon$
	$S_0 \to aSBb$	$S_0 \to aSBb$	$S_0 \to aSBb$	$S_0 \to aSBb$
		$B \to B$	$B \to B$	
		$B \to \varepsilon$	$B \to \varepsilon$	$B \to \varepsilon$
		$B \to bbSA$	$B \to bbSA$	$B \to bbSA$
				$A \to aaSS$

КОНТЕКСТНО-СВОБОДНЫЕ ГРАММАТИКИ

Грамматические правила вида $N \to \varepsilon$ называются ε-правилами .

Утверждение 7.11. Для любой КС-грамматики $G = (\Sigma, \mathcal{N}, P, S)$ существует эквивалентная КС-грамматика $G' = (\Sigma, \mathcal{N}, P', S)$, которая не содержит правил $N \to \varepsilon$ для всех нетерминалов $N, N \neq S$, отличных от начального нетерминала.

КОНТЕКСТНО-СВОБОДНЫЕ ГРАММАТИКИ

Грамматические правила вида $N \to \varepsilon$ называются ε-правилами.

Утверждение 7.11. Для любой КС-грамматики $G = (\Sigma, \mathcal{N}, P, S)$ существует эквивалентная КС-грамматика $G' = (\Sigma, \mathcal{N}, P', S)$, которая не содержит правил $N \to \varepsilon$ для всех нетерминалов N, $N \neq S$, отличных от начального нетерминала.

Доказательство. Проводится аналогично доказательству теоремы об устранении ε-правил для автоматов: если в множестве P есть пара правил $N \to \varepsilon$ и $N' \to uNv$, то добавим правило $N' \to uv$.

Очевидно, последовательное применение правил $N' \to uNv$ и $N \to \varepsilon$ равносильно применению правила $N' \to uv$. Поэтому как только возможности для добавления новых правил исчерпаны, можно удалить все ε-правила $N \to \varepsilon$, где $N \neq S$. QED

Контекстно-свободные грамматики

Пример

$$\begin{array}{lll} S_0 & \to & \varepsilon \mid aSBb \\ S & \to & \varepsilon \mid aSBb \\ A & \to & \varepsilon \mid bbSA \mid aaSS \\ B & \to & \varepsilon \mid bbSA \mid aaSS \end{array}$$

Контекстно-свободные грамматики

Пример

$$
\begin{array}{lcl}
S_0 & \to & \varepsilon \mid aSBb \\
S & \to & \varepsilon \mid aSBb \\
A & \to & \varepsilon \mid bbSA \mid aaSS \\
B & \to & \varepsilon \mid bbSA \mid aaSS
\end{array}
$$

$$
\begin{array}{lcl}
S_0 & \to & \varepsilon \mid aSBb \mid aBb \mid aSb \mid ab \\
S & \to & aSBb \mid aBb \mid aSb \mid ab \\
A & \to & bbSA \mid aaSS \mid bbA \mid bbS \mid bb \mid aaS \mid aa \\
B & \to & bbSA \mid aaSS \mid bbA \mid bbS \mid bb \mid aaS \mid aa
\end{array}
$$

КОНТЕКСТНО-СВОБОДНЫЕ ГРАММАТИКИ

Таким образом, справедлива

Теорема 7.12. Для любой КС-грамматики G существует эквивалентная КС-грамматика G', которая не содержит

- начального нетерминала в правых частях правил,
- правил-переименований,
- ε-правил для всех нетерминалов, отличных от начального.

Контекстно-свободные грамматики

Таким образом, справедлива

Теорема 7.12. Для любой КС-грамматики G существует эквивалентная КС-грамматика G', которая не содержит

- начального нетерминала в правых частях правил,
- правил-переименований,
- ε-правил для всех нетерминалов, отличных от начального.

Вопрос: в каком порядке нужно применять указанные эквивалентные преобразования?

Контекстно-свободные грамматики

Из теоремы 7.12 следует

Теорема 7.13. Алгоритмически разрешима проблема включения слова в КС-язык: $w \in L(G)$.

КОНТЕКСТНО-СВОБОДНЫЕ ГРАММАТИКИ

Из теоремы 7.12 следует

Теорема 7.13. Алгоритмически разрешима проблема включения слова в КС-язык: $w \in L(G)$.

Доказательство. Как показано в утверждениях 7.9-7.11, любую КС-грамматику можно привести к виду, указанному в Теореме 7.11. Рассмотрим какой-нибудь успешный вывод в этой КС-грамматике

$$\alpha_0 = S \xrightarrow{G} \alpha_1 \xrightarrow{G} \alpha_2 \xrightarrow{G} \cdots \xrightarrow{G} \alpha_{n-1} \xrightarrow{G} \alpha_n = w.$$

Этот вывод либо проводится за один шаг $S \to \varepsilon$, либо монотонно «возрастает», т.е. для любого $i, 1 \leq n$, строка α_i либо имеет большую длину, чем строка α_{i-1}, либо имеет ту же длину, но содержит большее число терминалов, чем строка α_{i-1}. Почему?
Значит, для проверки включения $w \in L(G)$ достаточно рассмотреть все выводы длины не превосходящей $2|w|$. QED

Контекстно-свободные грамматики

Чтобы решить проблему пустоты $L(G)=\varnothing?$ для КС-грамматик, введем отношение зависимости нетерминалов: нетерминал N' зависит от нетерминала N'' в грамматике G (обозначение $N'' \preceq_G N'$), если существует такое грамматическое правило $N' \to \theta N'' \eta$, в правой части которого содержится нетерминал N''. Рефлексивно-транзитивное замыкание отношения зависимости нетерминалов будем обозначать записью $\preceq_G^*$.

Контекстно-свободные грамматики

Чтобы решить проблему пустоты $L(G) = \varnothing$? для КС-грамматик, введем отношение зависимости нетерминалов: нетерминал N' зависит от нетерминала N'' в грамматике G (обозначение $N'' \preceq_G N'$), если существует такое грамматическое правило $N' \to \theta N'' \eta$, в правой части которого содержится нетерминал N'' . Рефлексивно-транзитивное замыкание отношения зависимости нетерминалов будем обозначать записью $\preceq_G^*$.

Нетерминал N в грамматике G будем называть

- завершаемым , если $N \xrightarrow{G}_* w$ для некоторого терминального слова w ;
- достижимым , если $S \xrightarrow{G}_* \theta N \eta$;
- полезным , если N присутствует в грамматическом выводе некоторого терминального слова из S .

КОНТЕКСТНО-СВОБОДНЫЕ ГРАММАТИКИ

Утверждение 7.14. Если нетерминал N не является полезным в грамматике G, то, удалив из нее все правила, содержащие N, получим грамматику, эквивалентную G.

Утверждение 7.14. Если нетерминал N не является полезным в грамматике G, то, удалив из нее все правила, содержащие N, получим грамматику, эквивалентную G.

Утверждение 7.15. Нетерминал N является достижимым в грамматике G тогда и только тогда, когда $N \preceq_G^* S$.

КОНТЕКСТНО-СВОБОДНЫЕ ГРАММАТИКИ

Утверждение 7.16. Множество завершаемых нетерминалов грамматики G — это наименьшее множество $\widehat{\mathcal{N}}$, удовлетворяющее следующим двум условиям:

1. если грамматика G содержит правило $N \to w$, где $w \in \Sigma^*$, то $N \in \widehat{\mathcal{N}}$,
2. если грамматика G содержит правило $N \to \alpha$ и все нетерминалы строки α содержатся в множестве $\widehat{\mathcal{N}}$, то $N \in \widehat{\mathcal{N}}$.

КОНТЕКСТНО-СВОБОДНЫЕ ГРАММАТИКИ

Утверждение 7.16. Множество завершаемых нетерминалов грамматики G — это наименьшее множество $\widehat{\mathcal{N}}$, удовлетворяющее следующим двум условиям:

1. если грамматика G содержит правило $N \to w$, где $w \in \Sigma^*$, то $N \in \widehat{\mathcal{N}}$,
2. если грамматика G содержит правило $N \to \alpha$ и все нетерминалы строки α содержатся в множестве $\widehat{\mathcal{N}}$, то $N \in \widehat{\mathcal{N}}$.

Доказательство. Самостоятельно.
(с применением индукции по длине грамматического вывода)

КОНТЕКСТНО-СВОБОДНЫЕ ГРАММАТИКИ

Пример

Грамматика G:

$$\begin{array}{lcl} S & \to & UX \mid VZ \\ T & \to & aav \mid bb \\ U & \to & aUa \mid bUb \\ V & \to & aTb \mid bTa \\ W & \to & YZY \mid aab \\ X & \to & \varepsilon \mid Xa \mid Xb \\ Y & \to & \varepsilon \mid YY \mid aU \\ Z & \to & W \mid b \end{array}$$

КОНТЕКСТНО-СВОБОДНЫЕ ГРАММАТИКИ

Пример

Грамматика G:

$$\begin{array}{lcl}
S & \to & UX \mid VZ \\
T & \to & aav \mid bb \\
U & \to & aUa \mid bUb \\
V & \to & aTb \mid bTa \\
W & \to & YZY \mid aab \\
X & \to & \varepsilon \mid Xa \mid Xb \\
Y & \to & \varepsilon \mid YY \mid aU \\
Z & \to & W \mid b
\end{array}$$

Достижимые нетерминалы: S, U, X, V, Z, T, W, Y .

Завершаемые нетерминалы: T, W, X, Y, Z, V, S .

Значит, U — бесполезный нетерминал, и его можно удалить.

КОНТЕКСТНО-СВОБОДНЫЕ ГРАММАТИКИ

Пример

Грамматика G':

$$\begin{array}{lcl} S & \to & VZ \\ T & \to & aa \mid bb \\ V & \to & aTb \mid bTa \\ W & \to & YZY \mid aab \\ X & \to & \varepsilon \mid Xa \mid Xb \\ Y & \to & \varepsilon \mid YY \\ Z & \to & W \mid b \end{array}$$

Достижимые нетерминалы: S, V, Z, T, W, Y .

Значит, X — бесполезный нетерминал, и его можно удалить.

КОНТЕКСТНО-СВОБОДНЫЕ ГРАММАТИКИ

Пример

Грамматика G'':

$$
\begin{array}{lcl}
S & \to & VZ \\
T & \to & aa \mid bb \\
V & \to & aTb \mid bTa \\
W & \to & YZY \mid aab \\
Y & \to & \varepsilon \mid YY \\
Z & \to & W \mid b
\end{array}
$$

Достижимые нетерминалы: S, V, Z, T, W, Y .

Завершаемые нетерминалы: T, W, Y, Z, V, S .

Все они полезные, и при этом $L(G'') = L(G)$.

Задача 7.4 Сформулируйте необходимое и достаточное условие полезности нетерминала в терминах свойств завершаемости и достижимости.

Задача 7.5 Опишите алгоритм вычисления всех полезных терминалов, обоснуйте его корректность и оцените его сложность.

Теорема 7.17. Алгоритмически разрешима проблема пустоты КС-язков: $L(G) = \varnothing$

Контекстно-свободные грамматики

Теорема 7.17. Алгоритмически разрешима проблема пустоты КС-язков: $L(G) = \varnothing$

Доказательство. Достаточно убедиться, что начальный нетерминал грамматики является завершаемым.

Нормальная форма Хомского

А нельзя ли унифицировать форму правил КС-грамматик?

Нормальная форма Хомского

А нельзя ли унифицировать форму правил КС-грамматик?

Грамматика G в нормальной форме Хомского — это КС-грамматика, все правила которой имеют вид $S \to \varepsilon$, $N \to a$ или $N \to N'N''$, где $a \in \Sigma,\ N', N'' \in \mathcal{N} \setminus \{S\}$.

Пример. G: в нормальной форме Хомского.

$$
\begin{aligned}
S &\to \varepsilon \mid VZ \\
Z &\to AA \mid BB \\
V &\to a \mid ZV \\
A &\to a \\
B &\to b
\end{aligned}
$$

НОРМАЛЬНАЯ ФОРМА ХОМСКОГО

Теорема 7.18. Каждая КС-грамматика эквивалентная некоторой грамматике в нормальной форме Хомского

Доказательство. Опишем последовательность эквивалентных преобразований, приводящую произвольную КС-грамматику к нормальной форме Хомского.

Нормальная форма Хомского

Теорема 7.18. Каждая КС-грамматика эквивалентная некоторой грамматике в нормальной форме Хомского

Доказательство. Опишем последовательность эквивалентных преобразований, приводящую произвольную КС-грамматику к нормальной форме Хомского.

1. Устраним начальный нетерминал S из правых частей правил (см. Утверждение 7.8);

Нормальная форма Хомского

Теорема 7.18. Каждая КС-грамматика эквивалентная некоторой грамматике в нормальной форме Хомского

Доказательство. Опишем последовательность эквивалентных преобразований, приводящую произвольную КС-грамматику к нормальной форме Хомского.

1. Устраним начальный нетерминал S из правых частей правил (см. Утверждение 7.8);
2. Для каждого терминала a заменим его во всех правилах новым нетерминалом T_a и добавим новое правило $T_a \to a$;

Нормальная форма Хомского

3. Устраним ε-правила (см. Утверждение 7.11);

Нормальная форма Хомского

3. Устраним ε-правила (см. Утверждение 7.11);
4. Устраним правила-переименования (см. Утверждение 7.10);

Нормальная форма Хомского

3. Устраним ε-правила (см. Утверждение 7.11);
4. Устраним правила-переименования (см. Утверждение 7.10);
5. Вводя новые вспомогательные нетерминалы, приведем все нетерминальные строки в правых частях правил, к двучленному виду:

$$A \to BCDE \quad \Rightarrow \quad \begin{cases} A \to BA', \\ A' \to CA'', \\ A'' \to DE, \end{cases}$$

НОРМАЛЬНАЯ ФОРМА ХОМСКОГО

3. Устраним ε-правила (см. Утверждение 7.11);
4. Устраним правила-переименования (см. Утверждение 7.10);
5. Вводя новые вспомогательные нетерминалы, приведем все нетерминальные строки в правых частях правил, к двучленному виду:

$$A \to BCDE \quad \Rightarrow \quad \begin{cases} A \to BA', \\ A' \to CA'', \\ A'' \to DE, \end{cases}$$

Каждое из преобразований — эквивалентное. После их применения образуется грамматика в нормальной форме Хомского. QED

КОНТЕКСТНО-СВОБОДНЫЕ ГРАММАТИКИ

Пример.

Пусть дана грамматика G:

$$\begin{array}{lcl} S & \to & VZ \\ T & \to & aa \mid bb \\ V & \to & aTb \mid bTa \\ W & \to & YZY \mid aab \\ Y & \to & \varepsilon \mid YY \\ Z & \to & W \mid b \end{array}$$

Очевидно, начального нетерминала нет в правых частях правил.

КОНТЕКСТНО-СВОБОДНЫЕ ГРАММАТИКИ

Пример.

Введем нетерминалы T_a и добавим новые правила $T_a \to a$

$$
\begin{array}{lcl}
S & \to & VZ \\
T & \to & T_a T_a \mid T_b T_b \\
V & \to & T_a T T_b \mid T_b T T_a \\
W & \to & YZY \mid T_a T_a T_b \\
Y & \to & \varepsilon \mid YY \\
Z & \to & W \mid T_b \\
T_a & \to & a \\
T_b & \to & b
\end{array}
$$

КОНТЕКСТНО-СВОБОДНЫЕ ГРАММАТИКИ

Пример.

Избавимся от ε-правил

$$
\begin{array}{lcl}
S & \to & VZ \\
T & \to & T_aT_a \mid T_bT_b \\
V & \to & T_aTT_b \mid T_bTT_a \\
W & \to & YZY \mid T_aT_aT_b \mid YZ \mid ZY \mid Z \\
Y & \to & YY \mid Y \\
Z & \to & W \mid T_b \\
T_a & \to & a \\
T_b & \to & b
\end{array}
$$

Нетерминал Y становится незавершаемым.

Контекстно-свободные грамматики

Пример.

Грамматика упрощается

$$
\begin{aligned}
S &\to VZ \\
T &\to T_aT_a \mid T_bT_b \\
V &\to T_aTT_b \mid T_bTT_a \\
W &\to T_aT_aT_b \mid Z \\
Z &\to W \mid T_b \\
T_a &\to a \\
T_b &\to b
\end{aligned}
$$

КОНТЕКСТНО-СВОБОДНЫЕ ГРАММАТИКИ

Пример.

Избавимся от правил-переименований

$$
\begin{array}{lcl}
S & \to & VZ \\
T & \to & T_a T_a \mid T_b T_b \\
V & \to & T_a T T_b \mid T_b T T_a \\
W & \to & T_a T_a T_b \mid b \\
Z & \to & T_a T_a T_b \mid b \\
T_a & \to & a \\
T_b & \to & b
\end{array}
$$

Нетерминал W становится недостижимым.

КОНТЕКСТНО-СВОБОДНЫЕ ГРАММАТИКИ

Пример.

Приведем правила к стандартному виду

$$
\begin{array}{lcl}
S & \to & VZ \\
T & \to & T_aT_a \mid T_bT_b \\
V & \to & T_aV' \mid T_bV'' \\
V' & \to & TT_b \\
V'' & \to & TT_a \\
Z & \to & T_aZ' \mid b \\
Z' & \to & T_aT_b \\
T_a & \to & a \\
T_b & \to & b
\end{array}
$$

Получаем грамматику в нормальной форме Хомского.

Контекстно-свободные грамматики

Задача 7.5. Предложите еще какие-нибудь правила эквивалентных преобразований, при помощи которых можно упрощать КС-грамматики. Воспользуйтесь ими для еще более полного упрощения грамматики, рассмотренной в предыдущем примере.

Задача 7.6. Является ли алгоритмически разрешимой следующая проблема: выяснить, является ли контекстно-свободным язык, порождаемый заданной неограниченной грамматикой.

Контекстно-свободные грамматики

Задача 7.7. [Трудная] Является ли алгоритмически разрешимой следующая проблема: выяснить, является ли контекстно-свободным язык, порождаемый заданной контекстно-зависимой грамматикой.

КОНТЕКСТНО-СВОБОДНЫЕ ГРАММАТИКИ

Задача 7.8. Грамматика в нормальной форме Грейбах — это контекстно-свободная грамматика, в которой каждое правило имеет один из следующих четырех видов:

$$A \to \varepsilon, \qquad A \to a, \qquad A \to aB, \qquad A \to aBC.$$

Доказать, что любая КС-грамматика эквивалентна некоторой грамматике в нормальной форме Грейбах.

КОНЕЦ ЛЕКЦИИ 7