

Домашнее задание от 12.10.
Теоретическая информатика. 2 курс.
Решения.

Глеб Минаев @ 204 (20.Б04-мкн)

20 октября 2021 г.

Содержание

Задача 4 1

Задача 4. Пусть дана грамматика (Σ, N, R, S) , задающая язык L . Рассмотрим грамматику (Σ, N', R', S') , где:

- N' состоит из символов A , $A_{a \rightarrow b}$ и $A_{a \leftrightarrow b}$ (копии A) для всяких $A \in N$ и $a, b \in \Sigma$, а также ещё одного символа S' .
- R' состоит из следующих правил. Для всякого правила $A \rightarrow u_0 B_1 \dots B_n u_n$ ($u_i \in \Sigma^*$, $B_i \in N$) из R мы добавим в R' правила
 1. то же правило $A \rightarrow u_0 B_1 \dots B_n u_n$,
 2. правило $A_{a \rightarrow b} \rightarrow u_0 B_1 \dots u_{i-1} B_i v B_{i+1} u_{i+1} \dots B_n u_n$, где v — это строка u_i (для некоторого i) с некоторым a , подменённым на b , (u_i должна содержать a ; иначе правило не добавляется),
 3. правило $A_{a \rightarrow b} \rightarrow u_0 B_1 \dots B_{i-1} u_{i-1} (B_i)_{a \rightarrow b} u_i B_{i+1} \dots B_n u_n$, где (n должно быть > 0 ; иначе правило не добавляется),
 4. правило $A_{a \leftrightarrow b} \rightarrow u_0 B_1 \dots B_i v B_{i+1} \dots B_j w B_{j+1} \dots B_n u_n$, где v — это строка u_i (для некоторого i) с некоторым a , подменённым на b , (u_i должна содержать a ; иначе правило не добавляется), а w — это строка u_j (для некоторого j) с некоторым b , подменённым на a , (u_j должна содержать b ; иначе правило не добавляется) (i и j никак не зависят друг от друга, даже порядком; если $i = j$ подмены считаются проведёнными одновременно, т.е. в u_i некоторая пара символов a и b поменялась местами),
 5. правило $A_{a \leftrightarrow b} \rightarrow u_0 B_1 \dots B_i v B_{i+1} \dots u_{j-1} (B_j)_{b \rightarrow a} u_j \dots B_n u_n$, где v — это строка u_i (для некоторого i) с некоторым a , подменённым на b , (u_i должна содержать a ; иначе правило не добавляется) (n должно быть > 0 ; иначе правило не добавляется) (i и j никак не зависят друг от друга),
 6. правило $A_{a \leftrightarrow b} \rightarrow u_0 B_1 \dots B_i v B_{i+1} \dots u_{j-1} (B_j)_{a \rightarrow b} u_j \dots B_n u_n$, где v — это строка u_i (для некоторого i) с некоторым b , подменённым на a , (u_i должна содержать a ; иначе правило не добавляется) (n должно быть > 0 ; иначе правило не добавляется) (i и j никак не зависят друг от друга),

7. правило $A_{a \leftrightarrow b} \rightarrow u_0 B_1 \dots u_{i-1} (B_i)_{a \rightarrow b} u_i \dots u_{j-1} (B_j)_{b \rightarrow a} u_j \dots B_n u_n$ (n должно быть > 1 ; иначе правило не добавляется) ($i \neq j$; других зависимостей между i и j нет),
8. правило $A_{a \leftrightarrow b} \rightarrow u_0 B_1 \dots u_{i-1} (B_i)_{a \leftrightarrow b} u_i \dots B_n u_n$ (n должно быть > 0 ; иначе правило не добавляется).

Также добавим правила вида $S' \rightarrow S_{a \rightarrow b}$ для любых $a, b \in \Sigma$.

Несложно видеть, что для каждого $A \in N$ правила для A в новой грамматике те же, что и раньше, а значит язык каждого A не изменился при переходе от старой грамматики к новой.

Покажем, что всякое $w \in L(A_{a \rightarrow b})$ это слово из $L(A)$, в котором некоторый a заменили на b , по индукции по размеру дерева w .

Рассмотрим первую подстановку в дереве разбора w .

- Если это замена 2, то можно заменить v заменить обратно на u_i , а $A_{a \rightarrow b}$ на B и получить дерево разбора некоторого $w' \in L(A)$. Тогда при возвращении обратно мы меняем в w' некоторый символ a на b . Что и требовалось показать.
- Если это замена 3, то поддерево $(B_i)_{a \rightarrow b}$ образует некоторое слово u , получаемое из $u' \in L(B)$ заменой a на b . Тогда заменяя $A_{a \rightarrow b}$ на A , $(B_i)_{a \rightarrow b}$ на B_i и конструируя поддерево B_i так, чтобы оно порождало u' , мы получаем дерево разбора $w' \in L(A)$. Возвращая всё назад мы заменяем u' на u , что равносильно замене некоторого a в u' на b . Т.е. w получается из $w' \in L(A)$ заменой некоторого a на b .

По аналогии можно сконструировать всякое такое слово w , заменяя либо какую-то правильную строку u_i на u_i с проделанной заменой, либо делегируя эту работу B_i , заменяя его на $(B_i)_{a \rightarrow b}$.

Таким образом $L(A_{a \rightarrow b})$ — это язык $L(A)$, где во всяком слове подменили a на b (если в слове нет a , то оно просто игнорируется). Аналогично, $L(A_{a \leftrightarrow b})$ — это язык $L(A)$, где во всяком слове поменяли местами какие-то a и b (если хотя бы одного из a и b в слове нет, слово игнорируется). Это видно из правил, где разбираются случаи, где могут происходить замены и как их обрабатывать. Следовательно, первым действием S' выбирает, какую замену хочет произвести, и запускает её, т.е.

$$\text{swap}(L) = \bigcup_{\substack{a, b \in \Sigma \\ a \neq b}} L(S_{a \leftrightarrow b}) = L(S).$$