Mathematical Methods for Computer Science 1 Spring 2017

Series 11

Sylvain Julmy

1

The grammar $G = (V, T, P, S)$ where

$$
\begin{aligned}
V &= \{P\} \\
T &= \{a, b\} \\
P &= P \to aPa \mid bPb \mid \epsilon \\
S &= P
\end{aligned}
$$

2

a)

This grammar generate the set of all possible balanced parenthesis string.
There are three possibilities :

1. (S) : we pick an opened parenthesis, S and a closed parenthesis, this is the only pick that consume the symbol of a string. So the opened parenthesis would always be closed when "returning" from S.
2. SS : do not generate any symbols, but allow the grammar to accept sequence of balanced parenthesis terms.
3. ϵ : the empty string is balanced and allow to terminate the recursion.

If we only pick $P = S \to (S) \mid \epsilon$, it is the set of any well balanced term, by adding SS, we allow to reapeat this indefinitely.

b)

This grammar generate the set of all possible operation with the operator $*$ and the symbol a.
Terms of such an operator are always inside parenthesis.
The proof is by induction on the structure :

Base case : in the base case, the accepted string is just the single symbol a or the product $(S * S)$ where S is either a terminal or a product $(S * S)$.

Induction : at each step of the construction, we have to pick $S = a$ or $S = (S * S)$. So the construction look likes an abstract syntax tree (AST) for product expression.
A string accepted by this grammar is always well balanced, because only the rules $S \to (S * S)$ generate parenthesis and an opened parenthesis is always closed.

3

$$
\begin{aligned}
&V = \{A, S, B\}\\
&T = \{a, b\}\\
&P = \{\\
&S \to ASB \mid \epsilon,\\
&A \to aAS \mid a,\\
&B \to SbS \mid A \mid bb\\
&\}\\
&S = S
\end{aligned}
$$

a)

S is a nullable variable and A,B are not nullable. Therefore, we obtain

$$
\begin{aligned}
&P = \{\\
&S \to ASB \mid AB,\\
&A \to aAS \mid aA \mid a,\\
&B \to SbS \mid bS \mid Sb \mid b \mid A \mid bb\\
&\}
\end{aligned}
$$

b)

Unit pair : (S, S), (A, A) and (B, B) are trivial unit pairs. Then (B, A), from $B \to A$, is a unit pair to. Therefore, we obtain

$$
\begin{aligned}
&P = \{\\
&S \to ASB \mid AB,\\
&A \to aAS \mid aA \mid a,\\
&B \to SbS \mid bS \mid Sb \mid b \mid aAS \mid aA \mid a \mid bb\\
&\}
\end{aligned}
$$

c)

Each terminals is generating, therefore a and b are generating. There are no production of P that does not contains a non-generating symbol, therefore all production are usefull.

4

Assume $X \in V'' \cup T''$, we know that $X \underset{G}{\overset{*}{\Rightarrow}} w$ for some $w \in T^*$ and every symbol used in order to generate w are generating, therefore $X \underset{G'}{\overset{*}{\Rightarrow}}$.
Hence X is not eliminated from G'', we know that $\exists \alpha, \beta : S \underset{G'}{\overset{*}{\Rightarrow}} \alpha X \beta$ and every symbol is reachable, therefore $S \underset{G''}{\overset{*}{\Rightarrow}} \alpha X \beta$.
We know that every symbol in $\alpha X \beta$ is reachable and in $V' \cup T'$, so they are generating in G'.
The derivation of a string xwy use only symbols that are reachable from S because they are reached by symbols in $\alpha X \beta$.
Finally, X can be any symbol in G'' and then G'' does not contains any useless symbol.

5

a)

$$
\begin{aligned}
&G = V, T, P, S \\
&V = \{S\} \\
&T = \{(,), \neg, \rightarrow, p, q\} \\
&P = \{ \\
&\quad S \rightarrow \neg S \mid (S \rightarrow S) \mid p \mid q \\
&\ \} \\
&S = S
\end{aligned}
$$

S occurs on the RHS of a production, therefore we create $S' \rightarrow S \in P$:

$$
\begin{aligned}
&P = \{ \\
&\quad S' \rightarrow S, \\
&\quad S \rightarrow \neg S \mid (S \rightarrow S) \mid p \mid q \\
&\ \}
\end{aligned}
$$

We don't have to remove any ϵ-production or unit productions, because none of these are present.
Then, we transform RHS of productions which owns more than 2 symbols :

$$
\begin{aligned}
S &\to \neg S \mid (S \to S) \mid p \mid q \\
\longrightarrow \\
S &\to \neg S \mid (C_1 \mid p \mid q \\
C_1 &\to S \to S) \\
\longrightarrow \\
S &\to \neg S \mid (C_1 \mid p \mid q \\
C_1 &\to SC_2 \\
C_2 &\to \to S) \\
\longrightarrow \\
S &\to \neg S \mid (C_1 \mid p \mid q \\
C_1 &\to SC_2 \\
C_2 &\to \to C_3 \\
C_3 &\to S)
\end{aligned}
$$

Finally

$$
\begin{aligned}
S' &\to S \\
S &\to \neg S \mid (C_1 \mid p \mid q \\
C_1 &\to SC_2 \\
C_2 &\to \to C_3 \\
C_3 &\to S) \\
\longrightarrow \\
S' &\to S \\
S &\to XS \mid YC_1 \mid p \mid q \\
C_1 &\to SC_2 \\
C_2 &\to ZC_3 \\
C_3 &\to SW \\
W &\to) \\
X &\to \neg \\
Y &\to (\\
Z &\to \to
\end{aligned}
$$

b)

We have a string w of length n, then we are going to take $n-1$ times a rule of the form $A \to BC$ where A,B and C are non-terminal symbols, such that we can construct a string with n non-terminal symbols.
On each non-terminal symbol of said string of length n, we apply a rule of the form $A \to a$ where A is non-terminal and a is terminal i.e. we apply n rules. In total we have applied $n-1+=2n-1$.