CS 4114
Solutions to Midterm Exam
March 1, 2001

[40] 1. Consider the following language:

\[ L_1 = \{ w \in \{a, b, c\}^* \mid \text{any substring } bb \text{ in } w \text{ is immediately followed by a } c \} \].

1. Give examples of 5 strings that are in \( L_1 \) and of 5 strings that are not in \( L_1 \).
2. Give a regular expression that represents \( L_1 \).
3. Give a regular grammar that generates \( L_1 \).

1. Here are 5 strings that are in \( L_1 \): \( baac, \lambda, abbcacbce, cababcba, \) and \( bbeccabbe \). Here are 5 strings that are not in \( L_1 \): \( bb, bbbe, ababc, bbbcba, \) and \( babababbceabbb \).

2. Here is a regular expression representing \( L_1 \):

\[ r_1 = ( (a \cup c \cup ba \cup bc)^* bbe)^* (b \cup (a \cup c \cup ba \cup bc)^*) \]

3. Here is a regular grammar that generates \( L_1 \):

\[
\begin{align*}
S & \rightarrow aS \mid cS \mid bA \mid b \mid \lambda \\
A & \rightarrow aS \mid cS \mid bB \\
B & \rightarrow cS \mid cC \\
C & \rightarrow aC \mid cC \mid aD \mid cD \\
D & \rightarrow bC \mid b \mid \lambda
\end{align*}
\]

[30] 2. Consider the following language:

\[ L_2 = \{ uab^ib^iR \mid u \in \{a, b, c\}^*, 1 \leq i \} \].

1. Give examples of 5 strings that are in \( L_2 \) and of 5 strings that are not in \( L_2 \).
2. Give a context-free grammar that generates \( L_2 \).
1. Here are 5 strings that are in $L_2$: $ab$, $cababbc$, $aaabbabbbabaaa$, $aaabbb$, and $cabbabbacc$. Here are 5 strings that are not in $L_2$: $\lambda$, $cc$, $aabc$, $abecca$, and $b$.

2. Here is a context-free grammar that generates $L_2$:

$$
S \rightarrow aSa \mid bSb \mid cSc \mid A \\
A \rightarrow aAb \mid ab.
$$

[40] 3. The regular grammar $G_3$ is given by

$$
S \rightarrow aS \mid aA \\
A \rightarrow aA \mid bB \\
B \rightarrow aS \mid b.
$$

1. Identify a string $w \in L(G_3)$ that has at least two different derivations. Also, give two different derivations for $w$.

2. Find a regular expression that represents $L(G_3)$.

1. The string $w = aabb$ has exactly two different derivations, namely:

$$
S \Rightarrow aS \\
\Rightarrow aaA \\
\Rightarrow aabB \\
\Rightarrow aabb
$$

and

$$
S \Rightarrow aA \\
\Rightarrow aaA \\
\Rightarrow aabB \\
\Rightarrow aabb.
$$

2. As was done in class and in homework 4, we set up a system of three equations with variables $r_S$, $r_A$, and $r_B$ as regular expressions representing the yield of $S$, $A$, and $B$, respectively.

$$
\begin{align*}
    r_S &= ar_S \cup ar_A \\
    r_A &= ar_A \cup br_B \\
    r_B &= ar_S \cup b.
\end{align*}
$$
After substituting the value of \( r_B \), we obtain a system of two equations.

\[
\begin{align*}
  r_S &= ar_S \cup ar_A \\
  r_A &= ar_A \cup b( ar_S \cup b ) \\
       &= a^* b( ar_S \cup b ).
\end{align*}
\]

The justification for this last step was given in class. After substituting the value of \( r_A \), we obtain the regular expression we want.

\[
\begin{align*}
  r_S &= ar_S \cup aa^* b( ar_S \cup b ) \\
       &= (a \cup aa^* ba) r_S \cup aa^* bb \\
       &= (a \cup aa^* ba)^* aa^* bb.
\end{align*}
\]

[40] 4. Context-free grammar \( G_4 \) is the following:

\[
\begin{align*}
  S &\rightarrow abbA \mid \lambda \\
  A &\rightarrow abaA \mid bbaA \mid \lambda.
\end{align*}
\]

1. Give a definition of \( L(G_4) \) in set theoretic form.

2. Give a recursive definition of \( L(G_4) \) that does not mention \( G_4 \).

3. Give a regular grammar \( G'_4 \) that is equivalent to \( G_4 \). (That is, you should be able to argue that \( L(G'_4) = L(G_4) \).)

---

1. The following is a set theoretic definition of \( L(G_4) \):

\[
\{ \lambda \} \cup \{ aab \}^+ \{ aba, bba \}^*.
\]

2. The following is a recursive definition of \( L(G_4) \):

   • **Basis:** \( \lambda \in L(G_4) \) and \( aab \in L(G_4) \).
   
   • **Recursive step:** If \( x \in L(G_4) \) and \( |x| > 0 \), then \( aabx \in L(G_4) \), \( xaba \in L(G_4) \), and \( xbba \in L(G_4) \).
   
   • **Closure:** Any element of \( L(G_4) \) can be obtained by a finite number of applications of the recursive step to the basis.

The closure is optional.
3. Here is a regular grammar $G'_4$ that is equivalent to $G_4$:

$$
S \rightarrow aA \mid \lambda \\
A \rightarrow aB \\
B \rightarrow bS \mid bC \\
C \rightarrow aD \mid bD \\
D \rightarrow bE \\
E \rightarrow aC \mid \lambda 
$$

To see that $G'_4$ represents the language we are talking about, we make these observations:

- $G'_4$ generates $\lambda$, and every other string generated by $G'_4$ has the prefix $aab$.
- For any nonempty string $x$ generated by $G'_4$, a prefix of one or more $aab$'s is generated before either the $S \rightarrow \lambda$ or the $B \rightarrow bC$ rule is used.
- Once the $C, D, E$ group of nonterminals is reached (via the use of the $B \rightarrow bC$ rule), then the $S, A, B$ group is never returned to.
- The $C, D, E$ group generates $(aba \cup bba)^+$.

Putting all these observations together, $G'_4$ generates exactly the strings in $L(G_4)$, as desired.