Formal languages and Automata

HW 3 Solution

Mohammad Jalali

Isfahan University of Technology

2021 may 4

1. CFL to CFG
2. CFG Apps
3. Q 2
4. chomsky normal form
5. ambiguous G.
6. Ambiguity App.
7. inherently ambiguous L.

1. CFL to CFG
2. CFG Apps
3. Q 2
4. chomsky normal form
5. ambiguous G.
6. Ambiguity App.
7. inherently ambiguous L.

L1

$$\{\ a^n b^m \mid m, n \geq 0,\ 2n \leq m \leq 3n\ \}$$

$$S \to aSbb | aSbbb | \epsilon$$

$$S \to aXbb | \epsilon \quad X \to S | Sb$$

L2

$$\{\ w \mid w \in \Sigma^*,\ n_a(w) = n_b(w) + 2\ \}$$

$$S \to XaXaX \quad X \to aXb \mid bXa \mid XX \mid \epsilon$$

mistakes:

$$S \to XaaX \quad X \to aXb|bXa|\epsilon$$

$$S \to XaXaX \quad X \to \epsilon|aXbX|bXaX$$

L3

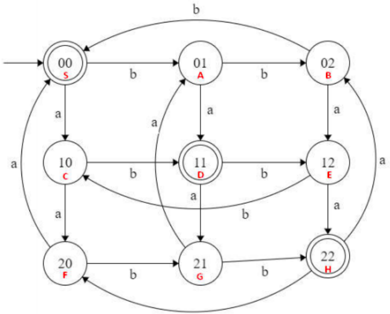
$\{\ a^i b^j c^k \mid i, j, k \geq 0,\ k = |i - j|\ \} \rightarrow \Sigma = \{a, b, c\}$
S_1: if i = k+j $(a^k a^j b^j c^k)$,
S_2: if j = k + i $(a^i a^i b^k c^k)$

$$S \rightarrow S_1 | S_2 | \epsilon$$
$$S_1 \rightarrow aSc | X \quad X \rightarrow aXb | \epsilon$$
$$S_2 \rightarrow YZ$$
$$Y \rightarrow aYb | \epsilon$$
$$Z \rightarrow bZc | \epsilon$$

L4

$$\{\ w \mid \exists\, k \geq 0 :\ n_a(w)\ +\ 2n_b(w) = 3k\ \}$$

$$\left.\begin{array}{l} n_a(w) = 3k+1 \textbf{ and } n_b(w) = 3k+1 \\ n_a(w) = 3k+2 \textbf{ and } n_b(w) = 3k+2 \\ n_a(w) = 3k+3 \textbf{ and } n_b(w) = 3k+3 \end{array}\right\} \Rightarrow \{n_a(w) + 2n_b(w) = 3k\} \equiv \{n_a(w) \bmod 3 = n_b(w) \bmod 3\}$$



L4

$$
\begin{aligned}
G(L_4): S &\to \varepsilon \mid aC \mid bA \\
A &\to aD \mid bB \\
B &\to aE \mid bS \\
C &\to aF \mid bD \\
D &\to \varepsilon \mid aG \mid bE \\
E &\to aH \mid bC \\
F &\to aS \mid bG \\
G &\to aA \mid bH \\
H &\to \varepsilon \mid aB \mid bF
\end{aligned}
$$

L5

$$\{ \; w_1 \# w_2 \mid w_1, w_2 \in \Sigma^*, \; w_1 \neq w_2^R \; \}$$

$$\begin{aligned}
S &\rightarrow aSa \mid aSb \mid bSa \mid bSb \mid T \mid Y \\
T &\rightarrow aXb \mid bXa \\
X &\rightarrow aXb \mid bXb \mid bXa \mid aXa \mid \# \\
Y &\rightarrow ZX \mid XZ \\
Z &\rightarrow aA \mid bB \mid a \mid b
\end{aligned}$$

L6

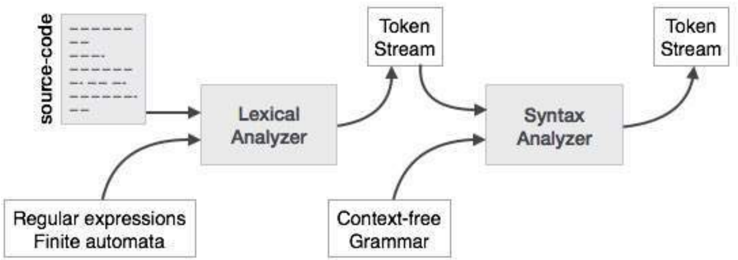
$$\{\ w \mid n_a(w) + n_b(w) = n_c(w)\ \} \rightarrow \Sigma = \{a, b, c\}$$

$$S \rightarrow \epsilon \mid aScS \mid cSaS \mid bScS \mid cSbS$$

1. CFL to CFG
2. CFG Apps
   - Syntax Analyzer
3. Q 2
4. chomsky normal form
5. ambiguous G.
6. Ambiguity App.
7. inherently ambiguous L.

1. CFL to CFG
2. CFG Apps
   - Syntax Analyzer
3. Q 2
4. chomsky normal form
5. ambiguous G.
6. Ambiguity App.
7. inherently ambiguous L.

Syntax Analyzer



Strings of balanced parantheses are not regular!

$$\{(^i)^i | i \geq 0\}$$

$$S \rightarrow (S) | \epsilon$$

1 CFL to CFG

2 CFG Apps

3 Q 2

4 chomsky normal form

5 ambiguous G.

6 Ambiguity App.

7 inherently ambiguous L.

G 1

$$G_1 : \; S \rightarrow aSb \,|\, bX \,|\, Xa, \quad X \rightarrow aX \,|bX \,|\, \epsilon$$

$$L(G_1) \; = \; \{a^i wb^i \,|\, w = b\sigma \text{ or } \sigma a \text{ where } \sigma \in \Sigma^*; \; i \geq 0\}$$

$$\bar{L}(G_1) = \left\{a^n b^n \mid n \geq 0\right\}$$

$$G(\bar{L}_1) : S \rightarrow \varepsilon \mid aSb$$

G 2

$G_2: \; S \to aSb \mid bSa \mid SS \mid \epsilon$

$$L(G_2) = \{w \mid n_a(w) = n_b(w)\}$$

$$\bar{L}(G_2) = \{w \mid n_a(w) \neq n_b(w)\}$$

$$\begin{aligned}
G(\bar{L}_2): S &\to S_A \mid S_B \\
S_A &\to XaXY \\
S_B &\to XbXZ \\
X &\to \varepsilon \mid aXb \mid bXa \mid XX \\
Y &\to \varepsilon \mid aXY \\
Z &\to \varepsilon \mid bXZ
\end{aligned}$$

1. CFL to CFG
2. CFG Apps
3. Q 2
4. chomsky normal form
5. ambiguous G.
6. Ambiguity App.
7. inherently ambiguous L.

Algorithm

Step 1 If the start symbol S occurs on some right side, create a new start symbol S' and a new production S' S.
Step 2 Remove Null productions.
Step 3 Remove unit productions.
Step 4 Replace each production A B1···Bn where n > 2 with A B1C where C B2 ···Bn. Repeat this step for all productions having two or more symbols in the right side.
Step 5 If the right side of any production is in the form A aB where a is a terminal and A, B are non-terminal, then the production is replaced by A XB and X a. Repeat this step for every production which is in the form A aB.

G 1

$$S \to AB \mid aB,$$
$$A \to abb \mid \lambda,$$
$$B \to bbA$$

$$\boxed{\begin{array}{l} S_0 \to S \\ S \to AB \mid aB \\ A \to abb \mid \varepsilon \\ B \to bbA \end{array}} \to \boxed{\begin{array}{l} S_0 \to S \\ S \to AB \mid aB \mid B \\ A \to abb \mid \varepsilon \\ B \to bbA \mid bb \end{array}} \to \boxed{\begin{array}{l} S_0 \to S \mid AB \mid aB \mid B \\ S \to AB \mid aB \mid B \\ A \to abb \\ B \to bbA \mid bb \end{array}} \to \boxed{\begin{array}{l} S_0 \to AB \mid aB \mid B \mid bbA \mid bb \\ S \to AB \mid aB \mid B \mid bbA \mid bb \\ A \to abb \\ B \to bbA \mid bb \end{array}}$$

$$\to \boxed{\begin{array}{l} S_0 \to AB \mid U_a B \mid U_{bb} A \mid U_b U_b \\ A \to U_a U_{bb} \\ B \to U_{bb} U_a \mid U_b U_b \\ U_{bb} \to U_b U_b \\ U_a \to a \\ U_b \to b \end{array}}$$

G 2

$S \to abAB,$
$A \to bAB \mid \epsilon,$
$B \to bbA$

$$
\boxed{\begin{array}{l} S_0 \to S \\ S \to abAB \\ A \to bAB \mid \varepsilon \\ B \to bbA \end{array}}
\to
\boxed{\begin{array}{l} S_0 \to S \\ S \to abAB \mid abB \\ A \to bAB \mid \varepsilon \mid bB \\ B \to bbA \mid bb \end{array}}
\to
\boxed{\begin{array}{l} S_0 \to S \mid abAB \mid abB \\ S \to abAB \mid abB \\ A \to bAB \mid bB \\ B \to bbA \mid bb \end{array}}
\to
\boxed{\begin{array}{l} S_0 \to U_{abA}B \mid U_a U_{bB} \\ A \to U_{bA}B \mid U_b B \\ B \to U_b U_{bA} \mid U_b U_b \\ U_{abA} \to U_a U_{bA} \\ U_{bA} \to U_b A \\ U_{bB} \to U_b B \\ U_a \to a \\ U_b \to b \end{array}}
$$

1. CFL to CFG
2. CFG Apps
3. Q 2
4. chomsky normal form
5. ambiguous G.
6. Ambiguity App.
7. inherently ambiguous L.

G 1

$$S \to aAB$$
$$A \to bBb$$
$$B \to A \,|\, \epsilon$$

$$S \Rightarrow aAB \Rightarrow abBbB \Rightarrow \begin{array}{l} B \to A \\ B \to \varepsilon \end{array} \begin{cases} abAbB \Rightarrow abbBbbB \Rightarrow \boxed{abbbb} \\ abbB \Rightarrow abbA \Rightarrow abbbBb \Rightarrow \boxed{abbbb} \end{cases}$$

G 2

$S \to SS \mid aSb \mid bSa \mid \epsilon$

$$S \Rightarrow SS \Rightarrow aSbS \Rightarrow abS \Rightarrow abaSb \Rightarrow \boxed{abab}$$

$$S \Rightarrow aSb \Rightarrow abSab \Rightarrow \boxed{abab}$$

G 3

$S \to AB \mid aaB$
$A \to a \mid Aa$
$B \to b$

$$S \Rightarrow aaB \Rightarrow \boxed{aab}$$
$$S \Rightarrow AB \Rightarrow AaB \Rightarrow aaB \Rightarrow \boxed{aab}$$

1. CFL to CFG
2. CFG Apps
3. Q 2
4. chomsky normal form
5. ambiguous G.
6. **Ambiguity App.**
7. inherently ambiguous L.

Consider the grammer:

$S \rightarrow if\ E\ then\ S \mid if\ E\ then\ S\ else\ S \mid a$

$E \rightarrow b$

The expression if b then if b then a else a has two parse tree:

if b then if b then a else a

if b then if b then a else a

Correct CFG

MIF: all then are matched, UIF: some then is unmatched.

$S \to MIF \mid UIF$

$MIF \to if\ E\ then\ MIF\ else\ MIF \mid a$

$UIF \to if\ E\ then\ S \mid if\ E\ then\ MIF\ else\ UID$

$E \to b$

1. CFL to CFG
2. CFG Apps
3. Q 2
4. chomsky normal form
5. ambiguous G.
6. Ambiguity App.
7. **inherently ambiguous L.**

L 1

$$\{\ a^i b^j c^k \mid i,j,k \geq 0, i = j \text{ or } j = k\}$$

- زبان L_1 ذاتا مبهم است. فرض می‌کنیم $s_1 = a^k b^p c^p$ و $s_2 = a^p b^p c^k$ دو رشته در زبان L_1 هستند که به‌ترتیب با درخت تجزیه‌های τ_1 و τ_2 به دست می‌آیند. $2p$ طول لم تزریق و $k = p!$ است. برگ‌های τ_1 با مقدار a را جدا می‌کنیم، در نتیجه زیررشته‌ای که ایجاد می‌شود دارای $2p$ (p تا b و p تا c) حرف است. با توجه به لم تزریق، در درخت تجزیه این زیررشته یک متغیر تکراری R قرار دارد. رشته $s_1 = a^k b^p c^p$ را با استفاده از متغیر R به‌شکل $uvxyz$ تقسیم می‌کنیم. هر کدام از v و y تنها می‌تواند شامل یک حرف از الفبا باشد؛ چرا که در غیر اینصورت uv^2xy^2z در زبان L_1 نخواهد بود. از طرفی v و y نمی‌توانند شامل a باشند، چرا که متغیر R را به‌نحوی انتخاب کردیم که در درخت تجزیه مربوط به زیررشته‌ای تنها متشکل از b و c قرار داشت. در نتیجه v متشکل از تعدادی b و y متشکل از تعدادی c است که طول هر دو را برابر p در نظر می‌گیریم. اکنون رشته $s = uv^d xy^d z = a^k b^k c^k$ ($d = \frac{k}{p}$ ؛ عددی صحیح است) از درخت تجزیه‌ای به دست می‌آید که b ها و c ها دارای پدر مشترک هستند. به همین ترتیب با استفاده از τ_2 می‌توان درخت تجزیه‌ای برای s ساخت که a ها و b ها پدر مشترک داشته باشند. در نتیجه رشته‌هایی به فرم $a^n b^n c^n$ با هر گرامری حداقل دارای دو درخت تجزیه متفاوت هستند که اجتناب‌ناپذیر است و این یعنی زبان L_1 **ذاتا مبهم** است.

L 2

$$\{ \ ww^R \mid w \in \Sigma^* \ \}$$

Thanks!