Rhea Mae Edwards

CS 321: Homework #6

1. DFA to CFG

M = (Q, Σ, δ, s, F) be a DFA

Construct a CFG G with…

- Nonterminals $\{A_p \mid q \in Q\}$
- Starting Nonterminal A_s
- Production Rules $\{A_p \rightarrow cA_q \mid \delta(p, c) = q\} \cup \{A_p \rightarrow \varepsilon \mid p \in F\}$

a) Using induction on $|w|$, prove that $\delta^*(s, w) = q$ in the DFA, if and only if $A_s \overset{*}{\Rightarrow} wA_q$ in the CFG.

Then prove that $L(M) = L(G)$, hence the conversion from DFA to CFG is correct.

<u>Claim:</u>

$\delta^*(s, w) = q$ (definition of δ^*) $\leftrightarrow$ $A_s \overset{*}{\Rightarrow} wA_q$ (definition of $\overset{*}{\Rightarrow}$)

<u>Base Cases:</u> $w = \varepsilon$

$\delta^*(s, \varepsilon) = S$ (definition of δ^*)

$A_s \overset{*}{\Rightarrow} A_s = \varepsilon\, A_\varepsilon$ (definition of $\overset{*}{\Rightarrow}$)

<u>Inductive Case:</u> Assuming that the inductive hypothesis is true…

$w = xb$

$\delta^*(s, xb) = q \leftrightarrow A_s \overset{*}{\Rightarrow} x(bA_q)$?

Then applying the recursive definition of δ^*…

$\delta(\delta^*(s, x), b) = q$

$\delta^*(s, x) = p$ (can apply the inductive hypothesis to)

$\delta(p, b) = q$ (explains a rule in the grammar, from the definition of the grammar)

Through this proof, want to end up with…

$A_s \overset{*}{\Rightarrow} xbA_q$

b) Draw DFA identifies multiples of 3 in binary. Give names to states.

Then show CFG when converting this DFA using approach above.

Note: May wish to recall the recursive definition of δ^*:

$$\delta^*(q, \varepsilon) \stackrel{\text{def}}{=} q$$

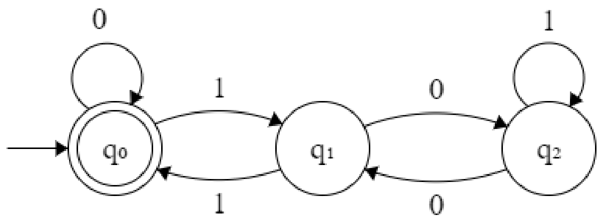
$$\delta^*(q, wb) \stackrel{\text{def}}{=} \delta(\delta^*(q, w), b) \qquad \text{when } b \text{ is a single character}$$

And the recursive definition of $\overset{*}{\Rightarrow}$:

- $\alpha \overset{*}{\Rightarrow} \alpha$, for all α
- If $\alpha \overset{*}{\Rightarrow} \beta X\gamma$, and $X \rightarrow \delta$ is a production rule, then $\alpha \overset{*}{\Rightarrow} \beta\delta\gamma$

Rhea Mae Edwards

DFA:



{w ∈ {0, 1}* | w is binary encoding of multiple of 3}

CFG:

$q_0 \rightarrow 0q_0 \mid 1q_1 \mid \varepsilon$

$q_1 \rightarrow 0q_2 \mid 1q_0$

$q_2 \rightarrow 0q_1 \mid 1q_2$

Rhea Mae Edwards

2. CFG to Chomsky normal form.

S → aSddd | T

T → bTdd | R

R → cR | ε

Step 1:

S* → S

S → aSddd | T

T → bTdd | R

R → cR | ε

Step 2:

S* → S

S → aX | T

T → Kd | R

R → cR | ε

X → Yd

Y → Zd

Z → Sd

K → Md

M → bT

Step 3:

S* → S

S → AX | T

T → KD | R

R → CR | ε

X → YD

Y → ZD

Z → SD

K → MD

M → BT

A → a

B → b

C → c

D → d

Step 4:

S* → S

S → AX | T

T → KD | CT | C

X → YD

Y → ZD

Z → SD

K → MD

M → BT

A → a

B → b

C → c

D → d

Step 5:

S* → AX | KD | CT | c

S → AX | KD | CT | c

T → KD | CT | c

X → YD

Y → ZD

Z → SD

K → MD

M → BT

A → a

B → b

C → c

D → d

Final Answer:

S* → AX | KD | CT | c

S → AX | KD | CT | c

T → KD | CT | c

X → YD

Y → ZD

Z → SD

K → MD

M → BT

A → a

B → b

C → c

D → d

3. Show languages are context-free.

a) $\{a^k b^m c^n \mid k, m, n \text{ not all equal}\}$

Consider $(\sim\{a^k b^m c^n \mid k, m, n \text{ not all equal}\}) \cap L(a^*b^*c^*) = \{a^k b^m c^n \mid k, m, n \text{ are all equal}\}$,

which is not a context-free language.

CFG describing this language is possible, meaning that this language is context-free. A representation indicating the not equal amount of a's, b's, and c's of a string can be shown with a CFG. The CFG can represent either a lesser or greater amount of number of a's than b's, a lesser or greater amount of number of b's than c's, and also a lesser or greater amount of number of c's of a's, where these values of the number of each of the three characters do not equal one another. Meaning that the language of

$\{a^k b^m c^n \mid k, m, n \text{ not all equal}\}$

is context-free.

b) $\{xcy \mid x, y \in \{a, b\}^* \text{ and rev}(x) \text{ is a substring of } y\}$

In Other Words: Strings that contain a single *c* character, where the part of the string occurring before *c* appears reversed somewhere after the *c*.

Example: <u>aaabb</u>cabababb<u>baaa</u>bba is in the language.

PDA showing where when you complete the process of adding/pushing *x* onto the idea of a stack, once the character *c* is read, then the popping off characters from the stack will begin. The popping will start, continue, and end with a valid string when the reversal of *x* has been read, also being a substring of *y*. Such a PDA is possible as just explained, meaning that this language of

$\{xcy \mid x, y \in \{a, b\}^* \text{ and rev}(x) \text{ is a substring of } y\}$

is context-free.