CS 420 Spring 2019
First Test
Solutions

Name: ____

Put all your answers on the test itself. Be sure to put your name above.

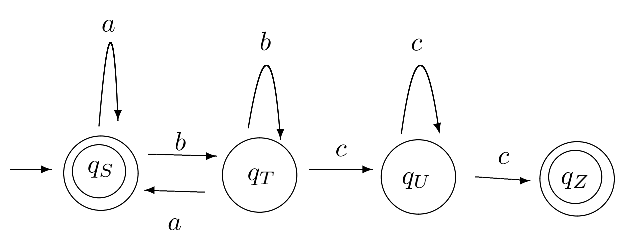
1. Using the method from class, give an NFA which is equivalent to the following right regular grammar.

$$\begin{array}{rcl} S & \to & aS|bT|\varepsilon \\ T & \to & bT|aS|cU \\ U & \to & cU|c \end{array}$$

Answer: We replace the rule $U \to c$ to obtain the following grammar:

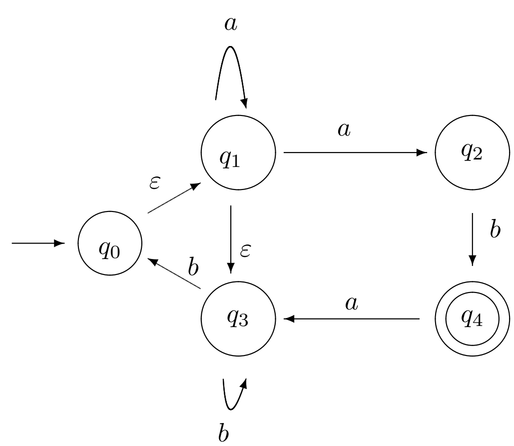
$$\begin{array}{rcl} S & \to & aS|bT|\varepsilon \\ T & \to & bT|aS|cU \\ U & \to & cU|cZ \\ Z & \to & \varepsilon \end{array}$$

We then obtain the following NFA:



[22 points]

2. Let N be the NFA given by



and let M be the DFA equivalent to N obtained by the method from class. Answer the following questions about N and M. (Put your answers in the blank spaces.)

(a) Does N accept the string ε? No

(b) Does N accept the string bba? No

(c) Does N accept the string ab? Yes

(d) Does N accept the string $abab$? Yes

(e) What is the start state of M? $\{q_0, q_1, q_3\}$

(f) Is $\{q_1, q_2, q_3\}$ an accept state of M? No

(g) Is $\{q_0, q_1, q_3, q_4\}$ an accept state of M? Yes

(h) What state does M go to from state $\{q_1, q_2, q_3\}$ reading an a?

$\{q_1, q_2, q_3\}$

(i) What state does M go to from state $\{q_1, q_2, q_3\}$ reading a b?

$\{q_0, q_1, q_3, q_4\}$

(j) What state does M go to from state $\{q_0, q_1, q_3, q_4\}$ reading an a?

$\{q_1, q_2, q_3\}$

(k) What state does M go to from state $\{q_0, q_1, q_3, q_4\}$ reading an b?

$\{q_0, q_1, q_3\}$

[22 points]

3. (a) Suppose that you are in the second step of putting a grammar into Chomsky normal form (eliminate ε-rules). You have already eliminated $Y \to \varepsilon$ and the current grammar is:

$$\begin{array}{rcl} S_0 & \to & S \\ S & \to & X|XaX|YX \\ X & \to & \varepsilon|XaX|XS \\ Y & \to & aY|a|XX \end{array}$$

Show the grammar that you get by eliminating $X \to \varepsilon$. (You are not being asked to eliminate all ε-rules, just $X \to \varepsilon$.)

$$\begin{array}{rcl} S_0 & \to & \underline{S} \\ S & \to & \underline{X|XaX|YX|\varepsilon|aX|Xa|a|Y} \\ X & \to & \underline{XaX|XS|aX|Xa|a|S} \\ Y & \to & \underline{aY|a|XX|X} \end{array}$$

(b) Suppose that you are in the third step of putting a grammar into Chomsky normal form (eliminate unit rules). You have already eliminated $A \to B$ and the current grammar is:

$$\begin{array}{rcl} S_0 & \to & S \\ S & \to & SaS|aA|a \\ A & \to & C|aA|a|CBA|SB \\ B & \to & CBA|a|SB \\ C & \to & B|aB|SaB|b|S \end{array}$$

Show the grammar that you get by eliminating $A \to C$. (You are not being asked to eliminate all unit rules, just $A \to C$.)

$$\begin{array}{rcl} S_0 & \to & \underline{S} \\ S & \to & \underline{SaS|aA|a} \\ A & \to & \underline{aA|a|CBA|SB|aB|SaB|b|S} \\ B & \to & \underline{CBA|a|SB} \\ C & \to & \underline{B|aB|SaB|b|S} \end{array}$$

(c) Suppose that in the fourth step of converting a grammar into Chomsky normal form, you have a rule $X \to YZab$. Show the collection of rules of the form $A \to BC$ and $A \to c$ that you use to replace this one rule in the fourth and fifth steps of converting the grammar.

$\underline{X \to YW,\ W \to ZV,\ V \to AB,\ A \to a,\ B \to b}$

[22 points]

4. Classify each of the following languages as being i) regular, or ii) context-free but not regular, or iii) not context-free. Justify your answers.

   (a) $\{u\#w | u, w \in \{a, b\}^* \text{ and } n_a(u) = n_a(w)\}$
   [Here, $n_c(y)$ means the number of occurrences of the symbol c in the string y.]

   Classification: ii

   Explanation:

   The language is context-free because it is generated by the following context-free grammar:

   $$S \rightarrow aSa|bS|Sb|\#$$

   We use the Pumping Lemma to show that the language is not regular. Given $p \geq 1$, let $s = a^p\#a^p$. Then, s is in the language and $|s| = 2p + 1 \geq p$. Given x, y, z with $s = xyz$, $|xy| \leq p$ and $|y| > 0$, we must have $y = a^k$ for some $k > 0$. We choose $i = 2$. Then, $xy^1z = a^{p+k}\#a^p$ which is not in the language.

(b) $\{u\#w | u, w \in \{a, b\}^*$ and $n_a(u) = n_a(w)$ and $n_b(u) = n_b(w)\}$

Classification: iii

Explanation:

We use the Pumping Lemma to show that the language is not context-free. Given $p \geq 1$, choose $s = a^p b^p \# a^p b^p$. Then, s is in the language and $|s| = 4p+1 \geq p$. Given u, v, x, y, z with $s = uvxyz$, $|vxy| \leq p$ and $|vy| > 0$, we choose $i = 2$ and consider cases to show that $uvvxyyz$ is not in the language.

Case 1: Either v or y contains #. Then $uvvxyyz$ contains two #'s so is not in the language.

Case 2: Both v and y are on the same side of the #. Then, in $uvvxyyz$, the string to the left of the # does not have the same length as the string to the right of the #, so $uvvxyyz$ is not in the language.

Case 3: v is to the left of the # and y is to the right of the #. Then, since $|vxy| \leq p$, v must consist of only b's and y consists of only a's. We have $uvvxyyz = a^p b^{p+|v|} \# a^{p+|y|} b^p$. Since $|v| + |y| > 0$, $uvvxyyz$ either has more b's to the left of the # than to the right of the #, or has more a's to the right of the # than to the left of the # (or both), so $uvvxyyz$ is not in the language.

(c) $\{u\#w | u, w \in \{a,b\}^*$ and $|u| + |w|$ is even $\}$

Classification: $\underline{\quad i \quad}$
Explanation:
The language is given by the regular expression

$$(\Sigma\Sigma)^*\#(\Sigma\Sigma)^* \cup \Sigma(\Sigma\Sigma)^*\#\Sigma(\Sigma\Sigma)^*$$

where Σ stands for the regular expression $a \cup b$.

[24 points]

5. A context-free grammar G is called *left regular* if every rule in G has one of the forms:

$$\begin{array}{lcl} A & \to & Ba \\ A & \to & a \\ A & \to & \varepsilon \end{array}$$

where A and B are variables and a is a terrminal.

Prove that if G is left regular, then $L(G)$ is regular.
[Hint: We proved in a homework solution that the reversal of a regular language is regular, and you can use this fact without reproving it. Also,think about what happens to the language generated by a grammar if you reverse the righthand sides of all the rules in the grammar.]

Solution: Let G be a left regular grammar. We must prove that $L(G)$ is regular. Define a grammar G' by reversing the right sides of all the rules in G. Then, G' is right regular and $L(G') = L(G)^R$. Since we know that right regular grammars generate regular languages, $L(G') = L(G)^R$ is regular. Finally, since the reversal of a regular language is regular, $(L(G)^R)^R = L(G)$ is regular.

[10 points]