CSCE 222: Discrete Structures for Computing
Section 503
Fall 2016

Joseph Martinsen

October 15, 2016

Problem Set 7

Due: 16 October 2016 (Sunday) before 11:59 p.m. on eCampus (`ecampus.tamu.edu`).
You must show your work in order to recieve credit.

Problem 1. (25 points)

1. A **palindrome** is a string that reads the same backward as it does forward, i.e. a string w, where $w = w^R$, where w^R is the reversal of the string w. Give a context-free grammar, expressed in Backus-Naur form, that generates the set of all palindromes over the alphabet $\Sigma = \{a, b, c\}$.
2. Use bottom-up parsing determine whether the following strings belong to $L(G)$, where $G = (\Sigma, N, S, P)$, where $\Sigma = \{a, b, c\}$, $N = \{S, A, B, C\}$, and $P = \{S \to AB, A \to aC, B \to aB, B \to bC, B \to b, C \to cb, C \to b\}$.
   (a) abbb
   (b) ababb
   (c) acbaacb
   (d) acbaaabcb

Solution.

1. $\langle palindrome\rangle ::= \langle char\rangle \Big| \langle charPchar\rangle \Big| \lambda$
   $\langle charPchar\rangle ::= a\langle palindrome\rangle a \Big| b\langle palindrom\rangle b$
   $\langle char\rangle ::= a \Big| b$
2. (a) abbb
   Using bottom up, $C \to b, A \to aC$ abbb can be generated as Abb. In combination with the productions, $C \to B, B \to bC$, abbb is generated □
   (b) ababb
   Using bottom up, $C \to b, B \to bC, B \to aB$ generates abb so ababb can be converted to abB. Using $C \to b, A \to aC$ ababb generates ab. Combining both, it is shown that ababb can be generated by AB □
   (c) acbaacb
   Using bottom up, there is no way a string ending in acb can be generated in this language. Only ending with c that ends as cb comes from $C \to cb$. Going one step up, only possibilites end with bcb from $B \to bC$ ∴ acbaacb is not in this language □

(d)

$acbaaabcb$	Given	(1)
$acbaaabC$	from$C \to cb$	(2)
$acbaaaB$	from $B \to bC$	(3)
$acbaaB$	from $B \to aB$	(4)
$acbaB$	from $B \to aB$	(5)
$acbB$	from $B \to aB$	(6)
aCB	from $C \to cb$	(7)
AB	from $A \to aC$	(8)
	$\square$	(9)

Problem 2. (25 points)
In **extended Backus-Naur form (EBNF)**, the symbol ? indicates that the preceding symbol, or group of symbols inside parentheses, is optional (can appear zero or once); the symbol $*$ indicates the the preceding symbol or group can be repeated zero or more times; the symbol $+$ indicates that the preceding symbol or group can appear one or more times.

1. Describe the language generated by each of these grammars expressed in EBNF.

   (a)

   $$\begin{aligned} S &::= L+D?L+ \\ L &::= a \mid b \mid c \\ D &::= 0 \mid 1 \end{aligned}$$

   (b)

   $$\begin{aligned} S &::= PD+ \mid D+ \\ P &::= + \mid - \\ D &::= 0 \mid 1 \mid 2 \mid 3 \mid 4 \mid 5 \mid 6 \mid 7 \mid 8 \mid 9 \end{aligned}$$

   (c)

   $$\begin{aligned} S &::= L^*(D+)?L^* \\ L &::= x \mid y \\ D &::= 0 \mid 1 \end{aligned}$$

2. Show that EBNF and BNF can generate the same languages by describing how productions for a grammar in EBNF can be translated into a set of productions for the grammar in BNF.

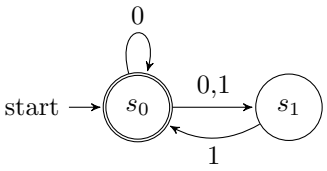
Solution.

Problem 3. (25 points)
Construct deterministic finite-state automata that recognize each of these languages.

1. the set consisting of the bitstrings 00, 11, and 010.
2. the set of bitstrings that start with 10 and end with one or more 1s.
3. the set of bitstrings consisting of an odd number of 0s followed by a final 1.
4. the set of bitstrings that have neither two consecutive 0s nor two consecutive 1s.

Solution.

Problem 4. (25 points)
Consider this nondeterministic finite-state automaton:



1. Construct a deterministic finite-state automaton that recognizes the same language.
2. What is the language that the automaton recognizes?

Solution.

Aggie Honor Statement: On my honor as an Aggie, I have neither given nor received any unauthorized aid on any portion of the academic work included in this assignment.

Checklist: Did you...
1. abide by the Aggie Honor Code?
2. solve all problems?
3. start a new page for each problem?
4. show your work clearly?
5. type your solution?
6. submit a PDF to eCampus?