COP-4020
PROGRAMMING LANGUAGE CONCEPTS
Instructor: Manuel Bermúdez
Spring 2018

MIDTERM EXAM I
Closed Books, Closed Notes, 50 minutes

Problem 1 ______ (20p.)
Problem 2 ______ (80p.)

SCORE ______ (100p.)

Name: ____________________________
UFID: ____________________________

Note: Turn in your work on this exam only
Problem 1 (20 pts):

Below are the rules for transforming a regular expression into an NFA, using the top-down algorithm. Complete the pictures below to expand the set of rules, adding three regular expression operators:

- The unary ‘+’ operator (one or more instances of the operand), e.g. \( a^+ \)
- The unary ‘?’ operator (zero or one instance of the operand), e.g. \( a? \)
- The binary ‘list’ operator (list of instances of the left operand, separated by instances of the right operand). Thus, \( a \text{ list } b \) is a list of a’s, separated by b’s, e.g. \( abababa \). Hint: \( a \text{ list } b = a(ba)^* \)

RE to NFA Algorithm 2 (top-down) Rules:
Problem 2 (80 pts):

Consider the following regular expression: \((a+b*c)^*a*b\)

a) (30 pts.) Transform this regular expression to an NFA, from there to a right-linear regular grammar, and from there back to the original regular expression. Show your work.

b) (25 pts.) Transform the NFA from Part a), to a DFA.

c) (25 pts.) Minimize de DFA obtained in Part b).

---

**Grammar:**

- \(A \rightarrow B\)
- \(B \rightarrow aB\)
- \(C \rightarrow aD\)
- \(D \rightarrow cB\)
- \(E \rightarrow \epsilon\)

**Equations:**

- \(A = B\)
- \(B = aB + cD = aB + b * cB + D = (a+b*c)^*D\)
- \(C = bC + cB = b * cB = (a+b*c)*a*b\)
- \(D = aD + bE = aD + b = a * b\)
- \(E = \epsilon\)