**COMP 412, Fall 2017**

**Practice Mid-Term Exam**

**Disclaimer:** THIS IS A PRACTICE EXAM. These questions are taken from past exams. The questions on the actual exam may or may not be similar.

1. **Comparisons**

In each part of this question, you are given two terms. Briefly define each term, and then explain the distinction or difference between them. *“Briefly” means a couple of sentences, not a couple of paragraphs.*

   a. Deterministic finite automaton (DFA) versus non-deterministic finite automaton (NFA)

   b. Hopcroft’s algorithm for DFA minimization versus Brzozowski’s algorithm for DFA minimization.

   c. Clean value versus dirty value in a local register allocator

   d. Regular expression versus context-free grammar

2. **Cycle of Constructions:**

Given an alphabet \{0,1\}, \(L\) is the set of all strings of 0’s and 1’s that have length \(\geq 1\) and begin and end with the string "11".

   a. Write a regular expression for the language \(L\).

   b. Diagram the canonical nondeterministic finite automaton produced by Thompson's construction for the regular expression you produced in subpart (a) of this question

   c. Use the subset construction to convert the NFA produced in part (b) into a deterministic finite automaton. State minimization is not necessary. (When you are done, this DFA should recognize the language \(L\).)

   Diagram this automaton by drawing each node of the DFA labeled with the list of original NFA states corresponding to that node.
3. **LR Parsing:**

The LR(1) table construction algorithm starts by building the Canonical Collection of Sets of LR(1) Items. An LR(1) item is a pair \([\alpha, \beta]\) where \(\alpha\) is a production in the grammar with a placeholder, \(\bullet\), somewhere in it, and \(\beta\) is the "lookahead" symbol, a single word (or terminal symbol) from the grammar.

a. If the grammar contains the production \(A \rightarrow \beta \, \delta \, \gamma\), what LR(1) items can that production generate. Assume a lookahead symbol of "w".

b. Define the term “handle” in an LR(1) parser?

c. Which of the lookahead symbols, if any, in your answer to part (a) represents a handle?

Assume that one of the sets of items built in the canonical collection contains the two items \([A \rightarrow \alpha \, \beta \, \bullet, \gamma]\) and \([B \rightarrow \alpha \, \beta \, \bullet, \delta]\).

d. What happens with these two items during the table construction if \(\gamma \neq \delta\)?

What happens with these two items during the table construction if \(\gamma = \delta\)?

4. **Context Sensitive Analysis**

Consider the classic expression grammar, given below. Write a set of ad-hoc syntax-directed translation rules that will construct an abstract syntax tree as an expression is parsed.

\[
\begin{align*}
\text{Expr} & \rightarrow \text{Expr} + \text{Term} \\
& \mid \text{Expr} - \text{Term} \\
& \mid \text{Term} \\
\text{Term} & \rightarrow \text{Term} \ast \text{Factor} \\
& \mid \text{Term} \div \text{Factor} \\
& \mid \text{Factor} \\
\text{Factor} & \rightarrow (\text{Expr}) \\
& \mid \text{identifier} \\
& \mid \text{number}
\end{align*}
\]

5. **Intermediate Representations**

Consider the simple assignment statement \(a \leftarrow b \ast c + d \ast e + f \ast g\).

Write down a representation of the statement as:

a. An abstract syntax tree (AST)

b. One address code

c. Three address code