CIS (4|5)61
Winter 2017 Midterm Exam

Your name: _____

Read through the whole exam once before you start working on answers, so that you see how the questions are related.

Some answers require diagrams. Work them out on scratch paper before writing in the exam. I may deduct points or not give any credit for writing or diagrams that are too difficult to read.

Think, then draft, then write. No other order works.

Total: _____

1. [5 points] Consider the following patterns for lex, flex, of jflex, with the usual maximum munch rule with ties broken by preferring the first among matching patterns.

```
[0-9]+          { return DEC; }
x?[0-9a-f]+     { return HEX; }
[a-z][a-z0-9]* { return ID; }
```

Which pattern is matched by each of these strings? (Write the token name in the provided space.)

- _____ 0
- _____ xa9fe
- _____ xa9fezk
- _____ x32
- _____ y31

(score)

2. [5 points]] Consider the same lex/flex/jflex patterns again, with the usual maximum munch and priority rules.

```
[0-9]+          { return DEC; }
x?[0-9a-f]+     { return HEX; }
[a-z][a-z0-9]* { return ID; }
```

Draw an NFA that matches all three patterns and distinguishes their accepting states (i.e., it should have at least one accepting state for DEC, at least one accepting state for HEX, and at least one accepting state for ID).

3. [10 points] Consider the same lex/flex/jflex patterns again, with the usual maximum munch and priority rules.

```
[0-9]+          { return DEC; }
x?[0-9a-f]+     { return HEX; }
[a-z][a-z0-9]* { return ID; }
```

Draw the DFA that lex, flex, or jflex would consruct to accept all three patterns, labeling the accepting states by the pattern they would match (i.e., each accepting state should be labeled with exactly one of DEC, HEX, or ID). You may use the label '*' on a transition to indicate a default, i.e., "anything in [a-z0-9] but not matching one of the other transitions from the same state."

Please work this out on scratch paper before writing the answer. I will not give credit for diagrams I can't read.

(score)

4. [5 points] In the following grammar I use λ to indicate an empty right-hand side. Otherwise the notational conventions are as in the Quack grammar manual.

$\langle Pgm \rangle$ ::= $\langle L \rangle$ `$`

$\langle L \rangle$ ::= $\langle L \rangle$ $\langle optSep \rangle$ $\langle I \rangle$

$\langle L \rangle$ ::= λ

$\langle optSep \rangle$ ::= `,`

$\langle optSep \rangle$ ::= λ

$\langle I \rangle$ ::= `i`

Which of the following are syntactically correct sequences of symbols in this grammar. (Circle the strings that are in the language.)

- `, , ,`
- `, i`
- `i i ,`
- `i i , i i`
- `i , i`

(score)

5. [15 points] Show that the following grammar is not LR(0) but is LALR(1). It is sufficient to show one state with an LR(0) conflict that is resolved by LALR(1) lookahead.

$\langle Pgm \rangle$ ::= $\langle L \rangle$ `$`

$\langle L \rangle$ ::= $\langle L \rangle$ $\langle optSep \rangle$ $\langle I \rangle$

$\langle L \rangle$ ::= λ

$\langle optSep \rangle$ ::= `,`

$\langle optSep \rangle$ ::= λ

$\langle I \rangle$ ::= `i`

(score)