Exam 1 (Feb 20, 2014) Name: ______________

Problem 1 (40 points)

(a) Enumerate the 3 major programming paradigms discussed in class.

(b) Which of the 3 major programming paradigms best matches the von Neumann computer architecture? Explain why.

(b) What is the von Neumann bottleneck? What type of memory is available in modern memory architectures that alleviates the von Neumann bottleneck?

(c) Which programming paradigm achieves computation through statements that change the state of the program?

Problem 2 (20 points)

There are multiple ways in which a programming language may be implemented.

(a) Enumerate the implementation methods discussed in class, in the order of their runtime speed, from slowest to fastest.

(b) Which implementation method can be said to perform *compilation by need* or *lazy compilation*?

Problem 3 (30 points)

Consider the following context free grammar, in which the alphabet is $\Sigma = \{\text{a, b, c}\}$ and the start symbol is S:

$$\begin{aligned} S &\rightarrow A\, c\, B \\ A &\rightarrow a\, A \mid b\, A \mid \epsilon \\ B &\rightarrow b\, B \mid b \end{aligned}$$

(a) Describe in English the language generated by this grammar.

(b) Design a regular expression that generates the **same** language.

Problem 4 (30 points)

Consider the following context free grammar, in which the alphabet is $\Sigma = \{a, b, c\}$ and the start symbol is S:

$$\begin{array}{rcl} S & \rightarrow & S\ a\ b \\ S & \rightarrow & S\ S \\ S & \rightarrow & c \end{array}$$

Which of the following strings can be generated by this grammar? For the strings that belong to the language generated by the grammar show a leftmost derivation.

(a) cabab

(b) ccacab

(c) cccabc

(d) cabcc

(e) cabcab

Problem 5 (30 points)

Consider the following context free grammar for simple Boolean expressions:

$$\begin{aligned} \langle \mathit{expr} \rangle &\rightarrow \langle \mathit{expr} \rangle \text{ and } \langle \mathit{expr} \rangle \\ &\rightarrow \langle \mathit{expr} \rangle \text{ or } \langle \mathit{expr} \rangle \\ &\rightarrow \text{A} \mid \text{B} \mid \text{C} \end{aligned}$$

(a) Prove that the grammar is ambiguous.

(b) Write an equivalent grammar that encodes the operator precedence rule for “and” and “or” in Boolean expressions.

Problem 6 (35 points)

Consider the following code snippets in Python. If evaluating the code results in errors, explain the cause. If the code evaluates without error, show the value of the variable **s** after evaluation.

(a)
```
s = [1, 2, 3, 4, 5, 6]
s[2:-1] = [’a’, ’b’]
```

(b)
```
s = “miner”
s[0] = ’d’
```

(c)
```
s = (10) * 2
```

(d)
```
s = (10, 11, 12)
s.append(13)
```

(e)
```
s = reduce(lambda x, y: x − y, range(4))
```

(f)

```
s = [x**2+y for x in [1, 2, 3] for y in [1, 2] if x + y > 2]
```

(g)

```
def mystery(a, *b, **c):
    yield a
    for item in b:
        yield item
    for item in c:
        yield item

for item in mystery(1, 2, 3, ou = 10, osu = 9)
        print item
```

Problem 7 (30 points)

Circle the statements that are **true**:

1. The syntactic rules of a programming language are specified using regular expressions.
2. The leaf nodes in a parse tree correspond to terminal symbols in the grammar.
3. Regular Grammars generate the same set of languages as Regular Expressions.
4. In a compiler, the output of the lexical analyzer is used as input for the syntactic analyzer.
5. Every function in Python must have a name.
6. In Python a function can be used as argument to another function.
7. Lists in Python are immutable.
8. Backus-Naur-Form (BNF) is a metalanguage for Context Free Grammars.
9. The precision of long integers in Python is the same as the precision of long integers in C.
10. This exam was easy.

Bonus History Question (5 points)

A man who knows four languages is worth _________? (Charles V, Holy Roman Emperor).