CS200 Fall 2014 WA1: Grammars
due 9/16/14 by **9:30AM (Hand-in before start of lecture)**

1. [10 points] A grammar is *ambiguous* if there is a sentence in the language that can produce two different derivation trees. Show that the following grammar is ambiguous by a) identifying a sentence that can produce more than one derivation tree and b) providing two different derivation trees for the sentence.

    <S> = <A>
    <A> = <A> + <A> | <B>
    <B> = a | b | c

2. [10 points] Given the following grammars with start symbol <S>, specify the type (0, 1, 2 or 3) of each one and give a reason that relates part of the grammar to a requirement for the grammar type. Note: while technically any grammar is Type 0, the answer should be the highest number that is correct.

    a. <S> = aaa<T>bb
        <T> = a<T><T>b | λ

    b. <S> = a<Q>bc | λ
        <Q>= a<Q>b<T> | λ
        b<T>bc = <S> | bbcc
        <T>b = b<T>

3. [24 points] For each of the following languages on T={a, b, c}, construct the corresponding regular expression and regular grammar.
    a. All strings containing exactly three a's.
    b. All strings containing at most three b's.

4. [10 points] Find a regular grammar that produces the language $aa^*(ab \cup a)^*$

5. [10 points] For the following expressions of binary operations, a) state whether it is prefix, infix, postfix or none, b) if it is none, modify it to make it legal prefix and if it is one of {prefix, infix or postfix}, state what it evaluates to. (All numbers are single digits.)
    a. 5 4 * 3 +
    b. 8 * 2 - 4
    c. * + - 8 2
    d. 3 9 2 + -
    e. / + 6 4 - 5 3

6. [36 points] For programming assignment 1, you were required to scan a file and find the words, while disregarding anything within <>. The following questions relate to this problem. You can use some special symbols: \> and \< can represent the terminals < and >. You may also use symbols from Java's Pattern class, e.g., \p{Lower}, \p{Upper}, \p{Alpha}, \p{Digit} and \p{Punct}.
    a. Give a grammar and a regular expression for the language consisting of words (which consist of just letters) separated by punctuation or whitespace (use the special symbols from Java's Pattern class for letters, numbers, punctuation and whitespace).
    b. Assuming we decided to identify numbers, give a grammar and a regular expression for identifying numbers both integers and floats (e.g., 1.2 365.492).
    c. Assuming we decided to recognize headings in HTML (e.g., <h1>…</h1>), give a grammar for recognizing headings h1 through h3 where the heading is "h" followed by a digit such that the beginning and end digits match (your grammar does not need to include nested headings).