Recall the Academic Integrity statement that you signed. Write all answers clearly on these pages, ensuring your final answers are easily recognizable. The number of points for each problem is clearly marked, for a total of 25 points. I will post my solutions on the web on Monday, off the Solutions link, after class.

1. (2 pts) Explain why in the past hardware was more expensive than software, but today the opposite is true (for full credit, think of two aspects to this answer).

2a. (2 pts) First, fill in each of the following costs as percentages of the lifetime cost of software. Now, assume that Company A builds a software system for $10 million. Compute the cost to maintain this system and fill in its total lifetime cost.

General cost to build software:   %
General cost to maintain software:   %

Lifetime cost (to both build and maintain Company A’s software): $

2b. (2 pts) The current Windows 7 release is estimated to contain about 50 million lines of code. Using the second best $/line of code estimate (the Ford Continental software, from the lecture), estimate the total cost to just build (not maintain) this operating system. Show your calculation.

3. (5 pts) Translate the following RHS of an EBNF rule into its equivalent syntax chart. Then, classify each of the examples on the right as legal or illegal according to this RHS (or its equivalent syntax chart).

\[ A(A[B|C])Z \]

1. AZ
2. ABZ
3. AAZ
4. AABCZ
5. AAAAZ
6. AABAACZ
7. AABACAAZ
8. ABACAAAAZ
4. (5 pts) In English, we often write one name by itself (Bob), two names with just an and between them (Bob and Carol), and more than two names with commas between them, with the last name also preceded by an and (Bob, Carol, Ted and Alice). Notice there is no , before the and.

Write an EBNF description named *name-sequence* that specifies only legal (according to the previous description) sequences of name (one, two, three, or more names). Note writing Bob, Bob and Bob is allowed. Your solution can comprise a single EBNF rule or a list of smaller rules. Try to write the simplest, most elegant solution possible. Use the *name* rule below in your *name-sequence* rule.

\[
\text{name} \leftarrow \text{Bob} \mid \text{Carol} \mid \text{Ted} \mid \text{Alice} \mid \text{Fred} \mid \text{Wilma} \mid \text{Barney} \mid \text{Betty}
\]

5. (6 pts) In the following Java code, identify and classify every token: Draw a vertical line between tokens, and underneath label every *identifier* with an I, every *keyword* with a K, every *separator* with an S, every *operator* with an O, every *literal* with an L, and every *comment* with a C. Write each letter below the box of the token that it classifies. For each literal, also indicate its type (int, double, boolean, char, or String).

Look at the tokens exactly as the Java compiler would. These tokens are very close together: please draw the boxes as neatly as you can.

```java
int x = Prompt.forInt("Enter",0,9); if (x<=5) {x++; System.out.print(""+x);} //close code
```

6. (3 pts) (a) Rewrite the following code, putting two spaces around every token: "ans=\+x+++y"; (b) Which of the following are decimal numerals (base 10 integer literals): 012 or -120 or 1. or 1e0 (hint: which are legal according to the stated EBNF); (c) What are the *semantics* of comments in Java i.e., after comment tokens have been recognized, how does the Java compiler process them? Be precise and accurate. Do not write anything about the *syntax* of comments, write just about their *semantics*.

(a)

(b)

(c)