**CS 100J Prelim 2 Fall 2006 Answers**

**Question 1.**
(a) Local variable: A variable declared in the body of a method. It is created when the frame for a call is created, before execution of the method body.

(b) // Store in c the number of positions k in Strings s1 and s2 such that s1[k] == s2[k]
   int n= Math.min(s1.length(), s2.length());
   int c= 0;
   // invariant: c = no. of positions i in s1[0..k-1] and s2[0..k-1] such that s1[i] == s2[i]
   for (int k= 0; k < n; k= k+1) {
      if (s1.charAt(k) == s2.charAt(k))
         c= c+1;
   }
   // c = no. of positions i in s1[0..n-1] and s2[0..n-1] such that s1[i] == s2[i]

**Question 2.**
(a) c.Bigger(a) : true,
(b) a.getJob(): ERROR. Apparent class is Student.
(c) c.getJob(): "work-study",
(d) d.getJob(): "Cabbie",
(e) c.getSound(): "I'm new here.",
(f) d.getSound(): "",
(g) ((Frosh)a).getSound(): "I'm new here.",
(h) ((Frosh)d).getSound(): ERROR,
(i) ((Senior)d).getSound(): "",
(j) ((Senior)a).getSound(): E RR.

**Question 3.**

```java
/** Constructor: a Senior with major "CS", name m, and gpa of 3.8 */
public Senior(String m) {
   super(m, 3.8);  major= "CS";
}
/** = "ob is a non-null Senior with the same fields as this Senior" */
public boolean equals(Object ob) {
   if (ob == null) return false;
   if (!(ob instanceof Senior)) return false;
   Senior obs= (Senior) ob;
   return this.major.equals(obs.major) &&
   this.getName().equals(obs.getName()) &&
   this.getGpa() == obs.getGpa();
}
```

**Question 4.**

```java
/** Edit string s as described in q. 4 */
public static String edit(String s) {
   String res= "";
   // inv: res contains the processed s[0..k-1]
   for (int k= 0; k != s.length(); k= k+1) {
      // If res is empty or ends in ". " (but not "i.e."),
      // append capitalized s[k] to res;
      // otherwise append s[k] to res.
      if ((res.length() == 0 && res.endsWith(" ")) ||
       res.endsWith("i.e.") ) {
         res= res +
         Character.toUpperCase(s.charAt(k));
      }
      else
         res= res + s.charAt(k);
   }
   return res;
}
```

**Question 5.**

```java
/** = i as a string but with leading 0's, if necessary, so that it is 3 digits
   Precondition: 0 <= i < 1000 */
public static String digit3(int i) {
   if (i < 10) return "00" + i;
   if (i < 100) return "0" + i;
   return "" + i;
}
/** = A string that contains a representation of i, but
   with a comma every three digits.
   Precondition: i > 0.
   Example: toString(5243642) is "5,243,642"
*/
public static String toString(int i) {
   if (i < 1000) return "" + i;
   return toString(i/1000) + "," + digit3(i%1000);
}
/** = no. elephants in this Elephant’s family tree */
public int treeSize() {
   int size= 1;  // for this elephant.
   if (father != null) size= size + father.treeSize();
   if (mother != null) size= size + mother.treeSize();
   return size;
}
```