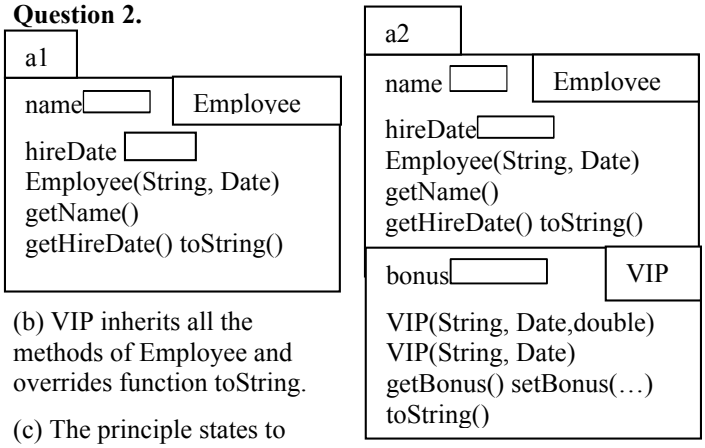
Prelim 1 Solutions, CS100J Spring 2008

Question 1. (a) = is used in an assignment statement, b == c tests whether two expressions have the same value, and .equals(…) (when defined in a class) tests whether the contents of two objects are the same.

(b) k is created during the first step of executing the call: when drawing the frame for the call.

(c) ans 2.65, ans 2.5, ans is 3.0, ans 2.0

Question 2.



(b) VIP inherits all the methods of Employee and overrides function toString.

(c) The principle states to initialize inherited fields first.

(d) Create a new object of class VIP, execute call VIP("David Gries", **new** Date(1979 – 1900, 2, 1), 10000), and yield the name of the object as the value of the **new** expression. Note that step 2 involves evaluating another new expression, but it need not be mentioned.

```
/** An instance represents an employee */
public class Employee {
  private String name; // employee's name
  private Date hireDate; // date hired

  /** Constructor: employee named n hired
                   on date d */
  public Employee (String n, Date d)
      { name= n;  hireDate= d; }

  /** = name of the employee */
  public String getName() { return name;}

  /** = a repr. of this employee */
  public String toString() {
      return "Name: " + name +
      ", Date of Hire: " + hireDate;
  }
}

/** Instance represents a VIP employee*/
public class VIP extends Employee{
  private double bonus; // VIP's bonus

  /** Constructor: a VIP with name n,
      hire date d, and bonus b */
  public VIP (String n, Date d, double b)
      { super(n, d);  bonus= b; }

  /** Constructor: VIP with name n,
      hire date d, bonus 0 */
  public VIP(String n, Date d)
      { this(n, d, 0); }

  /** = the bonus for this employee*/
  public double getBonus()
      { return bonus; }

  /** set the bonus to b*/
  public void setBonus(double b)
      { bonus= b; }

  /** = a description of this VIP */
  public String toString() {
      return super.toString() +
            " Bonus: " + bonus;
}}
```

Question 3.

```
public class ThreeNumbers {
  private int x, y, z;

  public ThreeNumbers(int x,int y,int z)
    { this.x= x; this.y= y; this.z=z; }

/** Swap fields x and y */
  public void swapXY()
    { int temp= x; x= y; y= temp; }

  /** Swap fields y and z */
  public void swapYZ() { … }

  /** Swap fields y and z */
  public void swapXZ() { … }

  /** Sort fields so that x <= y <= z */
  public void sort() {
    if (y < x){ swapXY(); }
    if (z < x){ swapXZ(); }
    if (z < y){ swapYZ(); }
}}
```

3g. We give perhaps more test cases than are necessary. (3, 4, 5), (3, 5, 4), (4, 3, 5), (4, 5, 3), (5, 4, 3), (4, 5, 3), (4, 4, 5), (4, 5, 4), (5, 4, 4), (4, 4, 4).

Question 4.

```
/** = s, prepended with '0' if nec., so that it has >= 2 chars.
       Precondition. s.length() >= 1. */
public static String prepend0(String s) {
  if (s.length() < 2) { s= '0' + s; }
  return s;
}

/** = s, but in form F1. Pre: s is in form F1 or F2 */
public static String reformat(String s) {
  int k= s.indexOf("/");
  if (k == -1) { return s; }

  // { s has the form month/day/year }
  String month= s.substring(0,k);
  s= s.substring(k+1);
  k= s.indexOf("/");
  String day= s.substring(0,k);
  String year= s.substring(k+1);
  return year + "." + prepend0(month) + "." + prepend0(day);
}

/** = "Date s1 occurs before s2. Pre: s is in form F1 or F2 */
public static boolean comesBefore(String s1, String s2) {
  return reformat(s1).compareTo(reformat(s2)) < 0;
}
```