Computing *Unassessed* Tutorial 2

Susan Eisenbach

20 November 2002

1. Write a Java program that reads in exactly 10 real numbers from the keyboard and prints them out in ascending (sorted) order. You should write a method that takes an unsorted list of numbers and returns a sorted list. You should call this method from your main program.

   There are many ways of sorting numbers. Numerous books have been written describing hundreds of algorithms. For this question you may use whatever algorithm you would like. Here is an algorithm if you would like an idea:

   > Read in all the numbers, sort them and then print them all out. To sort the numbers find the smallest and swap it with the element at the front of the list, repeat this process on the sublist that excludes the first element of the list. Continue repeating this process until you have used up the list.

2. Declare a variable `student` which consists of a person's name, mark for Programming, for Logic and a grade for Lab. A mark is a number (between 0 and100) and a grade is a letter (between A and F).

3. Write a predicate (a boolean method) `isStronger`, which takes two `student`s and returns `true` if and only if the first `student` has done better than the second in the ordering below.

   1. the Programming mark is most important,
   2. numerical order of Maths marks is the determining factor when two students have the same Programming mark,
   3. alphabetical order of Lab grades is the determining factor when two students have the same Programming and Maths marks.

1.

```
void main(){
double[] vector = new double[10];
     print("type in exactly 10 numbers-->");
     for (int i = 0; i<vector.length; i++){
          vector[i] = readDouble();
     }
     vector = sort(vector);
     for (int i = 0; i<vector.length; i++){
          println((i+1) + ": " + vector[i]);
     }
}
double[] sort(double [] v){
//post: forall i,j i<j ==> v[i] <= v[j]
double temp;
     for (int i = 0; i<v.length; i++){
      for (int j = i+1; j<v.length; j++){
          if (v[i] > v[j]){
               temp = v[i];
               v[i] = v[j];
               v[j] = temp;
          }
      }
     }
     return v;
}
```

2.

```
class Student{
     String name;
     int programming;
     int maths;
     char lab;
}
```

3.

```
boolean isStronger(Student s1, Student s2){
     return s1.programming > s2.programming ||
          s1.programming == s2.programming &&
          s1.maths > s2.maths ||
          s1.programming == s2.programming &&
          s1.maths == s2.maths && s1.lab < s2.lab;
     }
```