CS 1301 - Homework 10

Due before October 31, 1:57 *am* on Vista (WebCT). You can resubmit your work up until the deadline.

1. **ReverseWord.java** – Write a method, reverseWord that accepts an array of characters and returns a new array with array reversed on a word-by-word basis. For example:

   ```
   input:  [I,   , a, m,   , i, r, o, n,   , m, a, n]
   output: [m, a, n,   , i, r, o, n,   , a, m,   , I]
   ```

   You can assume: (a) that there will always be exactly one space between words, (b) words are composed of only upper and lower case letters, (c) there is no space(s) before the first word or after the last word.

2. **Deviation.java** – Do problem 6.11 from the text. Note, this problem illustrates that you cannot use int[] in place of double[]. Further, there is not a straightforward way to cast int[] to double[]. Thus, you will have duplicate code in deviation(double[] x) and deviation(int[] x). As an alternative in deviation(int[] x), you could copy int[] to double[] and then call deviation(double[] x). Example, calculating standard deviation by hand:

   Find the sample standard deviation of: 4, 9, 3, 6, 5

   $$mean = \bar{x} = \frac{1}{5}\sum_{i=1}^{5} x_i = \frac{1}{5}(4+9+3+6+5) = \frac{1}{5}(27) = 5.4$$

   $$deviation = s = \sqrt{\frac{\sum_{i=1}^{n}(x_i - \bar{x})^2}{n-1}}$$

   $$= \sqrt{\frac{\left[(4-5.4)^2 + (9-5.4)^2 + (3-5.4)^2 + (6-5.4)^2 + (5-5.4)^2\right]}{4}}$$

   $$= \sqrt{\frac{\left[(-1.4)^2 + (3.6)^2 + (-2.4)^2 + (0.6)^2 + (-0.4)^2\right]}{4}}$$

   $$= \sqrt{\frac{21.2}{4}} = 2.302172$$

3. **ReverseSelection.java** – Do problem 6.17. Write as a method with an array initializer in main to test.

4. **SortStudents.java** – Do problem 6.19. The description may be a little unclear; there is one score for each student. Hint 1: you can use (copy/paste) your method from problem 2 to sort the scores, with a small modification. Hint 2: when you swap scores in the scores array, you'll also need to swap names in the names array.

5. **Hours.java** – Look at the example two-dimensional array in problem 6.23 from the text. Do **not** work the problem from the text. Work this one: write the following methods:

   a. **int[] totalHoursEmployees(int[][] hours)** – Returns an array of total hours for each employee for the week (*i.e.* sum each row).

   b. **int totalHoursEmployee(int empID, int[][] hours)** – Returns the total hours for the employee at index *empID* for the week (*i.e.* sum the row corresponding to the employee, *empID*).

c. **int[] totalHoursDays(int[][] hours)** – Returns an array of the total hours for each day of the week (*i.e.* sum each column).

d. **int totalHoursDay(int dayID, int[][] hours)** – Returns the total hours for the day at index *dayID* for the week (*i.e.* sum the column corresponding to the day, *dayID*).

e. **int totalHours(int[][] hours)** – Returns the total hours worked for all employees for the week.

f. **int totalHours(int[] empTotals)** – Accepts an array of total hours by employee (*e.g.* the result of part *a*). Returns the total hours worked for all employees for the week.

g. **int[] overtimeEmployees(int[] empTotals)** – Accepts an array of total hours by employee (*e.g.* the result of part *a*). Returns an array of indices of employees who have worked more than 40 hours in the week (overtime).

h. **void printOvertimeHours(int[] otimeEmpID, int[][] hours)** – Accepts an array of indices of employees who have worked overtime. Prints the hours, total hours, and overtime hours for all employees who worked overtime. The display should look like this:

Employee	Sun	Mon	Tues	Wed	Thurs	Fri	Sat	Total Hours	Overtime Hours
2	0	10	10	10	12	0	3	45	5

etc.

Hint: you should use the method in part *b* to help you.