Homework No. 1

Due date: Sept. 27 (Tuesday)

In this homework, you are required to complete the one-dimentioanl Sparse Matrix ADT for two dimensional array of doubles.

```
public class SparseMatrix {
        public static SparseMatrix create(String filename);
        public SparseMatrix add(SparseMatrix other);
        public SparseMatrix transpose();
        public void print();
        ...
}
```

● static SparseMatrix create (String filename) reads data from a specified file and constructs an internal representation of a sparse matrix. The input data file format should look like the following:

```
4 6 5
0 0 1
1 2 3
1 5 3
2 2 5
3 4 18.
```

Each line must contain a triple {row, col, value} delimited by either blank or tab. The first line shows the dimension and the number of non-zero entries in the original two dimensional matrix. (The number of non-zero entries can be shown in integer.)

● SparseMatrix add(SparseMatrix other) adds the target matrix object with the one given as a parameter. As a result, the returned matrix object will eventually represent the sum of two matrices. (C = A.add(B) implies C <-- A + B). Note that A should not be modified as a result of this operation.

● [Optional] SparseMatrix transpose() performs a matrix transposition on the target matrix object. I.e., if A is a given sparse matrix, A.transpose() will return transformed matrix that is a transpose of A. Note that A should not be modified as a result of this operation.

You need to make use of "SparseMatrix.java".

For SparseMatrix.java, you can freely modify the file to complete the code. But, do NOT modify the "main()" method. You may add additional fields or private methods, if necessary. However, no public methods are allowed to be changed.

Program Testing:

```
$ java –cp . SparseMatrix p matrixA
$ java –cp . SparseMatrix t matrixA
$ java –cp . SparseMatrix a matrixA matrixB
```

Implementation Guidelines:

- Use an array implementation.
- Neglect error conditions. (Assume that everything is all right.)

What to submit?

- Hardcopy of your source code. (Submit in class.)
- Your source files archived in .zip or .jar format. (Email to your TA (홍승호, seunghonice@gmail.com))