Assignment 1

Hyoungshick Kim

September 19, 2018

1 Sparse matrix

This is a programming assignment to test your understanding of sparse matrices.

- Your goal is to implement an efficient data structure to represent sparse matrices and support the matrix operations for the data structure.

- You will write a code in the C programming language to print out the sparse matrix into the output files named 'hw1_output.txt' after performing several elementary matrix operations from two given *square* integer matrices (n-by-n matrix) in the input file named 'hw1_input.txt' where a square matrix is a matrix with the same number of rows and columns. The input file consists of the order of two square matrices (A and B), two square input matrices themselves, and the matrix operations to be performed. Each part is separated from the next part by a character $. The details are as follows:

    - The first part of the input file represents the order (i.e., the number of rows) of the matrix A and B.

    - The second part of the input file represents the matrix A containing the value of each element in the matrix A in a row major format.

    - The third part of the input file represents the matrix B containing the value of each element in the matrix B in a row major format.

    - The fourth part of the input file enumerates m operations (Addition, Multiplication, and Transposition) to be performed where m is the number of operations. We use +, *, and ' to denote those operations, respectively. We do not consider operator precedence. All operations have left to right associativity.

- In the output file (hw1_output.txt), you should output the matrix after performing each operation in the input file (hw1_input.txt). Each output matrix is separated from the next output matrix by a character $. If the output matrix is a dense matrix (i.e., when the ratio of the number of non-zero elements to the number of total elements in the matrix is greater than 1/3), you should print out the output matrix in a two dimensional array; otherwise, you should print out the matrix in the sparse matrix representation. For the two dimensional array representation, you should print out the character 'D' and the matrix itself from the second line. For the

sparse matrix representation, you should print out the character 'S' and the sparse matrix itself from the second line; each line consists of the row, column and value of each non-zero element in the matrix, respectively. Please use the *column-major* order sparse matrix.

- The following is an example of input and output files:

```
[Input file: hw1_input.txt]
3
$
0 0 0
2 0 -1
0 0 0
$
0 1 -4
0 -1 0
0 0 0
$
A+B*B'

[Output file: hw1_output.txt]
D
0 1 -4
2 -1 -1
0 0 0
$
S
1 2 -1
2 2 3
2 3 -8
$
S
2 1 -1
2 2 3
3 2 -8
```

- You will be judged by (1) the correctness of the results returned by your submitted program, (2) the actual running time of the program and (3) the well-written document to explain your source code and the performance analysis of your algorithm. For test, we will use $3 \leq n \leq 1,000$, $1 \leq m \leq 500$ and $5 \leq t \leq 10,000$ where t is the number of non-zero elements. Please test your code extensively with several inputs, so you are sure it works correctly.

- You cannot use any pre-defined data structures except arrays provided by some libraries.

- Please upload your source code (c files), instructions to illustrate how your source code works, document to explain your code and the performance analysis to iCampus. Submit your assignment by midnight, Sunday September 30; **late submis-**

sions are allowed with a penalty. **Each 24 hours (or part thereof) late will cost you 20%.**

- **Your assignments must be your own original work.** We will use a tool to check for plagiarism in assignments.