**Data Structure\_2071035 Lee Somin**

**Technical Report – TransposeMatrix.cpp**

*Theorical Explanation of Functions in ‘TransposeMatrix.cpp’*

**#define MAX\_TERMS**

This definition is for assigning the size of ‘element data[]’. The value should be ‘rows\*cols’ of the matrix B that is needed to be transposed, since the matrix can have maximum ‘rows\*cols’ nonzero data.

**#define SWAP(x, y, t)**

This definition is for performing swap operation after transposing the matrix, when the data should be sorted from data that has lowest row number o highest row number.

**struct element**

This structure contains the actual element value of the sparse matrix. It contains the nonzero element’s information of row, column, and value. This structure is used inside the structure ‘Sparse Matrix’.

**struct sparseMatrix**

This structure contains array of data as form of ‘element’ array, the row and column size of the actual dense matrix and the number of nonzero terms.

**Sparse\_matrix\_transpose**

Input: sparseMatrix a

Return: sparseMatrix b

This function was made to perform the transpose operation to given sparseMatrix ‘a’ for input. When the function is called, the transposed matrix ‘b’ is set to ‘a.cols\*a.rows’ matrix. Then, for all the terms in matrix ‘a’ is put into ‘b’ with the row and column switched. Lastly, the function sorts the b into the ascending order (bubble sort), to use the function ‘Show\_matrix\_dense’. When the sorting is finished, the function returns sparseMatrix ‘b’ which is the transpose of matrix ‘a’.

**Show\_matrix\_dense**

Input: sparseMatrix a

Return: non

This function prints out the input sparseMatrix ‘a’ in dense form. The inner ‘for’ statement is for printing out each element. When inner ‘for’ statement is complete, the function changes line and prints next row by increasing i by 1. The ‘if’ statement determines weather the data of ‘cnt’th place of input matrix matches the current location. Since ‘cnt’ starts from 0 to a.terms, the input matrix SHOULD be in ascending order based on row, then column.

**main()**

input: non

Return: non

The main function first assigns and initializes the original sparseMatrix B. The matrix can be changed if wanted. Then, it executes the transpose operation by calling Function ‘Sparse\_matrix\_transpose’ and saves it to BT. Finally, it prints out the original matrix B and the transposed matrix BT in dense matrix form using function ‘Show\_matrix\_dense’.