

MA 251 Data Structures

Laboratory Assignment 5

03-09-2019

Note: Upload your programs to the server (deadline: 4:30 pm)

1. In this lab, we will write functions to store and process a sparse matrix. Please download the files sparse.h and lab5_1.c. The C file contains three functions – readSparse(), transpose() and printSparse(). You need to complete these functions. You are expected to turn in the completed C file as well as a snapshot of your running program. A typical input/output session is depicted below.

Expected output: [row and col values start from 1]

\$/a.out

Enter the size of matrix (rows,columns):3 4

Enter no. of non-zero elements:4

Enter the next triple(row,column,value):2 1 5

Enter the next triple(row,column,value):2 3 3

Enter the next triple(row,column,value):3 2 1

Enter the next triple(row,column,value):3 4 2

After Transpose:

row column value

4 3 4

1 2 5

2 3 1

3 2 3

4 3 2

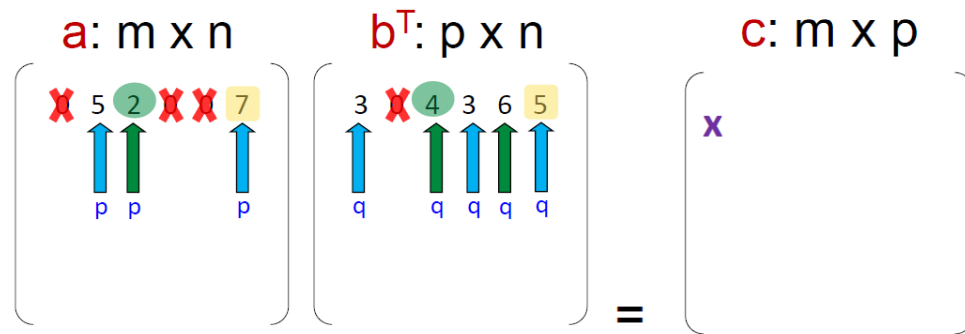
2. In this assignment, we will multiply two matrices **a** and **b**. The classical approach to multiply two matrices is shown below. as shown

$$\begin{matrix} \mathbf{a: m \times n} \\ \left(\begin{array}{cccccc} 0 & 5 & 2 & 0 & 0 & 7 \end{array} \right) \end{matrix} \begin{matrix} \mathbf{b: n \times p} \\ \left(\begin{array}{c} 3 \\ 0 \\ 4 \\ 3 \\ 6 \\ 5 \end{array} \right) \end{matrix} = \begin{matrix} \mathbf{c: m \times p} \\ \left(\begin{array}{c} \mathbf{x} \end{array} \right) \end{matrix}$$

$$X = 0 \times 3 + 5 \times 0 + 2 \times 4 + 0 \times 3 + 0 \times 6 + 7 \times 5 = 43$$

$$c(i,j) = \sum a(i,k) \times b(k,j)$$

The classical approach takes $O(\text{rows} \times \text{cols}^2)$. An alternative is to compute the transpose of matrix **b** and do a polynomial addition. Write a program to multiply two sparse matrix using the second approach. To compute matrix transpose, use the transpose function of Assignment 1.



Expected output:

Matrix 1: (4x4)
 Row Column Value
 1 2 10
 1 4 12
 3 3 5
 4 1 15
 4 2 12

Matrix 2: (4x4)
 Row Column Value
 1 3 8
 2 4 23
 3 3 9
 4 1 20
 4 2 25

Result of Multiplication: (4x4)
 Row Column Value
 1 1 240
 1 2 300
 1 4 230
 3 3 45
 4 3 120
 4 4 276

Turn in the code and snapshot of the output.