【问题描述与分析】

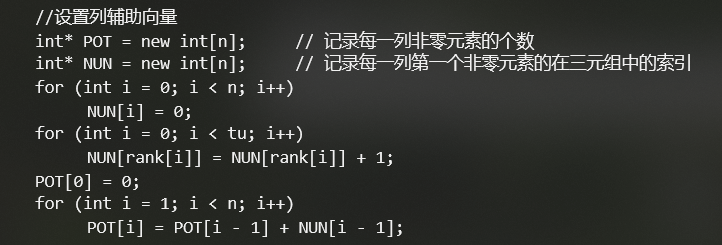
该问题要求实现一个稀疏矩阵的转置算法。问题的难点在于两点：一是稀疏矩阵中存在大量零元素，如何对其进行更高效的存储；二是如何设计算法使得转置后的矩阵仍能够以行序号由小到大排序，方便查看。

【数据结构的选择】

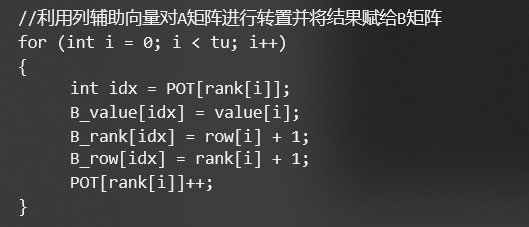
采用二维数组直接存储稀疏矩阵会造成大量的空间浪费，因此采用三元组的方式分别存储非零元素的值、行号和列号，可以最大程度节省空间。其次为了方便转置后的矩阵能够以行序号由小到大排序，设计列辅助向量POT和NUN来实现转置算法，其中NUN存储每一列非零元素的个数，POT存储每一列第一个非零元素在三元组中的序号。

【算法设计】

首先设计程序引导用户输入稀疏矩阵三元组A，根据非零元素个数tu构建三个静态向量value、row和rank，在其基础上通过遍历所有非零元素设置列辅助NUN和POT，具体算法实现如下：

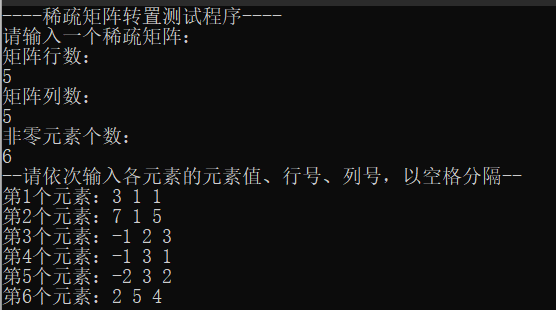


通过列辅助向量实现矩阵的转置，首先对所有非零元素进行遍历，根据列辅助向量得到各元素在转置矩阵B三元组中的位置，然后对B中相应位置的元素的值、行号和列号进行赋值。算法实现如下：

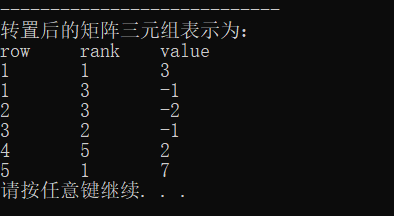


【测试用例】

首先输入矩阵的行数、列数及非零元素的个数进行初始化，然后依次输入各非零元素的值、行号和列号，以空格划分。



结果输出为：



【评价与心得】

该方法使用三元组存储稀疏矩阵并巧妙地采用列辅助向量的方式实现了稀疏矩阵的转置以及以行号对各非零元素进行排序，其时间复杂度为O（n+tu）。

其算法实现相对容易，不过设计辅助变量的思想值得反复玩味与学习。