

SparseMatrix Add(a,b)::= if the dimensions of a and b are the same, return the matrix produced by adding corresponding items, namely those with identical row and column values. else return error

SparseMatrix Multiply(a,b)::= if number of columns in a equals number of rows in b, return the matrix d produced by multiplying a by b according to the formula:

$$d[i][j] = \sum (a[i][k] \cdot b[k][j])$$

where d(i,j) is the (i,j)-th element else return error.

2.5.2 Sparse Matrix Representation

- We know that we can characterize uniquely any element within a matrix by using the triple $\langle \text{row}, \text{col}, \text{value} \rangle$. We use an array of triples to represent a sparse matrix.
- Since we want our transpose operation to work efficiently, we should organize the triples so that the row indices are in ascending order. For all the triples in any row, the column indices should also be in ascending order.

SparseMatrix Create(maxRow, maxCol)::=

#define MAX_TERMS 101 /* maximum no. of terms + 1 */

typedef struct {

int col;

int row;

int value;

} term;

term a[MAX_TERMS];

a[0].row contains the number of rows; a[0].col contains the number of cols; a[0].value contains the total no. of nonzero entries.

	row	col	value
a[0]	6	6	8
a[1]	0	0	15
a[2]	0	3	22
a[3]	0	5	-15
a[4]	1	1	11
a[5]	1	2	3
a[6]	2	3	-6
a[7]	4	0	9
a[8]	5	2	28