

Assignment: Data Structure and STL

Sparse matrix is a matrix of which most of the elements are zeros. To save the storage, we can use several format to store sparse matrices. Those formats include Compressed Row Storage, Compressed Column Storage, etc. See here: http://netlib.org/linalg/html_templates/node90.html.

Now we are going to use a simpler format. In this format, we only store the non-zero elements. Each element is characterized by <row, col, value>.

The matrix is characterized by a class SparseMatrix, with the usage of vector in STL.

The declaration code should be:

```
#include<vector>
using namespace std;

struct term {
    int col;
    int row;
    double value;
};

class SparseMatrix{
private:
    vector<term> data;
    int ncol;
    int nrow;
};
```

Here we set SparseMatrix has all the non-zero values stored in vector<term> data. ncol and nrow are the numbers of columns and rows of the matrix.

Following that, your work is listed as following:

1. Create default constructor, copy constructor;
2. Implement the assignment operator;
3. Implement the member functions
 double get(int i, int j); //get the i-th row, j-th column element
 void set(int i, int j, double v); //change the i-th row, j-th column element by value v
4. Complete the SparseMatrix by adding the operator+ function, add two sparse matrix together.
5. Add the operator* function, SparseMatrix times SparseMatrix, return another SparseMatrix.
6. Create main function to test the two operators.

Zip your code, output and any documents and submit into the Blackboard.