



Course: DSA-5005

Computing Structures

Fall 2024

Paulina Castillo

paulina.castillo@ou.edu

1. The code file is attached to the homework. However, can be found in the following Git repository https://github.com/paulinacastillov/computer_structures
2. I used chat GPT 4 to help me with different features of my code. For example when I don't understand an error or when I need ideas about how to implement something. Examples of the prompts and suggestions that I ask are the following:
I got an error over and over and I didn't understand why, so I asked chat GPT

```
why do I get expected ; at the end of declaration here
void displayMatrix const () {
    for (int i=0; i < myMatrix.size(); i++) {
        myMatrix[i].display();
    }
};
```

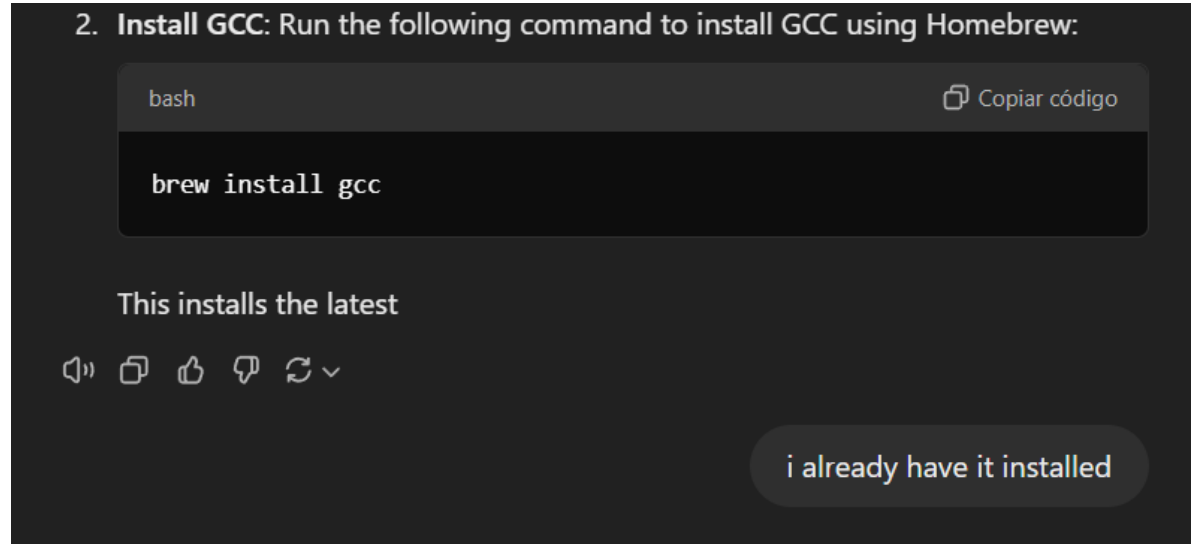
Based also in the class sparseRow I asked

```
generate an inputMatrix method that inserts sparseRow
into myMatrix vector
```

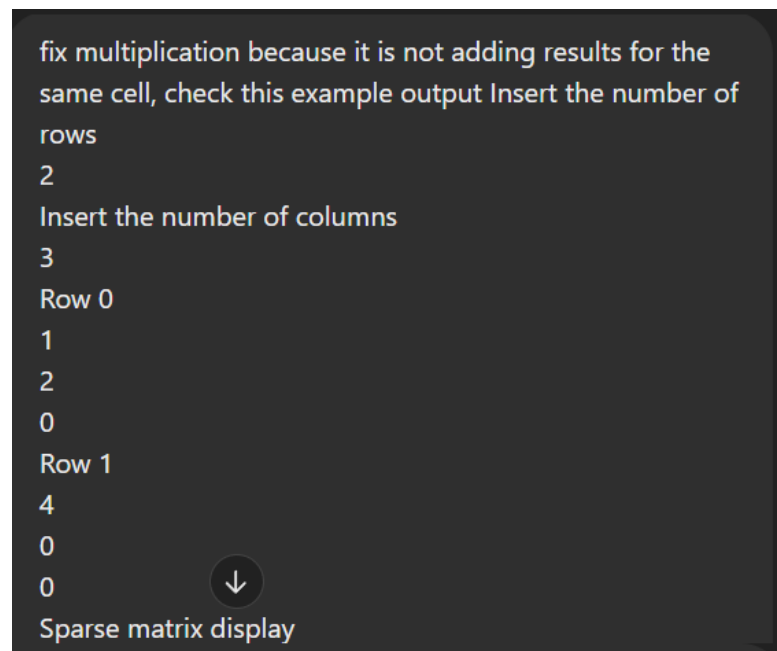
And I asked most errors that I couldn't fix by myself

```
got no matching member function for push_back error
```

Sometimes it suggested something that I've already done before so my suggestions were explanations



When I had the functions into my classes I started to try small examples and the trying to fix if there were some errors.



And with the code that I got I started to implement with the code that I already had. Sometimes Was useless because Chat GPT changed from a pointer to a variable and I need to change it manually (it was faster that explain what was grong and the check if the changes were ok or not).


An answer :

The issue you're facing with the `Multiply` method is that it's inserting multiple values for the same cell without adding them together. When multiplying two matrices, if multiple products contribute to the same cell in the result matrix, they need to be summed together.

To fix this, we can maintain a map or a 2D array to accumulate the sums for each cell during the multiplication process. Here's how we can adjust the `Multiply` method to correctly add values to the same cell:

Revised `Multiply` Method:

cpp

 Copiar código

// Multiply two SparseMatrix objects

3. I create a main class just to avoid having to write manually each time two matrix

```
void inputMatrixManually2() {
    addValue(0, 1, 25);
    addValue(0, (int)3, 10);
    addValue(1, 3, 36);
    addValue(2, 7, 101);
    addValue(3, 0, 67);
    addValue(3, 5, 72);
    addValue(4, 3, 44);
    addValue(4, 5, 93);
    addValue(5, 1, 55);
    addValue(6, 3, 76);
    addValue(7, 3, 85);
    addValue(7, 5, 23);
};
```

```
int main() {
    SparseMatrix matrix1 = SparseMatrix(5, 8, 0);
    matrix1.inputMatrixManually1();
    matrix1.displaySparse();

    SparseMatrix* tr = matrix1.Transpose();
    cout << "Transpose sparse matrix" << endl;
    tr->displaySparse();

    SparseMatrix matrix2 = SparseMatrix(8, 8, 0);
    matrix2.inputMatrixManually2();
    matrix2.displaySparse();

    SparseMatrix* result1 = matrix1.Add(matrix2);

    if (result1) {
        cout << "Sum:" << endl;
        result1->displayMatrix();
    }
}
```

I checked that the output of my code had the same results as the output text file provided. After that, I just commented on the “fix” `main()`. In general, everything was fine except for the sum of matrices. I didn’t understand why so I asked chat GPT and he solved the part of the code that I have it. I don’t understand exactly what changed, The two codes looked alike, so probably was just tipping. I also used Chat GPT to create some matrices random and with Symbolab made the operations and compared them with my code.

2. Matriz C (3x2) y Matriz D (2x3):

$$\bullet C = \begin{pmatrix} 2 & 3 \\ 4 & 5 \\ 6 & 7 \end{pmatrix}$$

$$\bullet D = \begin{pmatrix} 1 & 0 & 2 \\ 3 & 4 & 5 \end{pmatrix}$$

$$\begin{pmatrix} 2 & 3 \\ 4 & 5 \\ 6 & 7 \end{pmatrix} \begin{pmatrix} 1 & 0 & 2 \\ 3 & 4 & 5 \end{pmatrix}$$

Steps

Examples

$$\begin{pmatrix} 2 & 3 \\ 4 & 5 \\ 6 & 7 \end{pmatrix} \begin{pmatrix} 1 & 0 & 2 \\ 3 & 4 & 5 \end{pmatrix}$$

Solution

$$\begin{pmatrix} 11 & 12 & 19 \\ 19 & 20 & 33 \\ 27 & 28 & 47 \end{pmatrix}$$

First one in matrix format

```
2      3
4      5
6      7
```

First one in sparse matrix format

Sparse matrix display

Row #	Col #	Value
-------	-------	-------

0	0	2
---	---	---

0	1	3
---	---	---

1	0	4
---	---	---

1	1	5
---	---	---

2	0	6
---	---	---

2	1	7
---	---	---

Second one in matrix format

1	0	2
---	---	---

3	4	5
---	---	---