Semantic Error 1:

The first semantic error appears at line 271, where the function is attempting to create an instance of a pervious matrix with the same dimensions. However, the function is currently creating a matrix using the default constructor, automatically creating a 3x3 matrix, even though the previous matrix may not be 3x3.

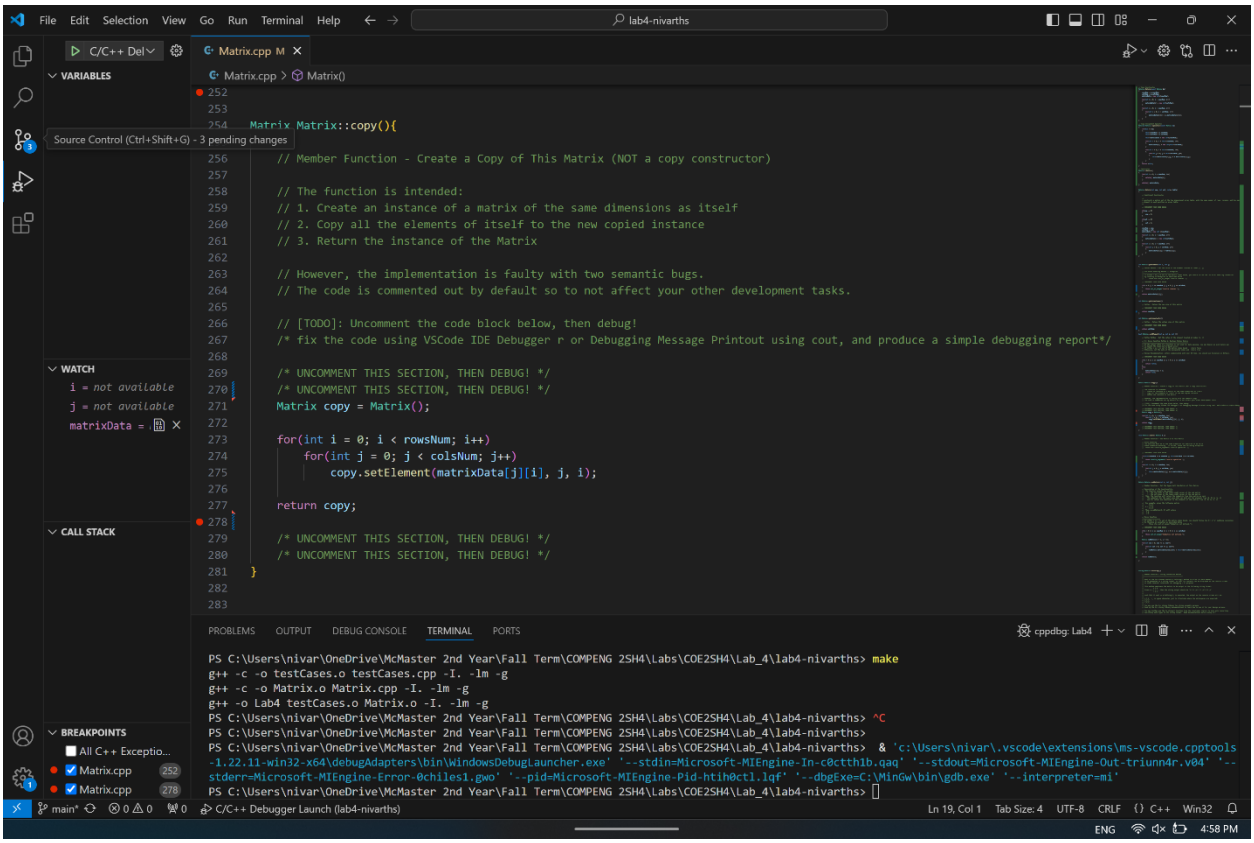

Figure 1: Semantic Error 1 before debug

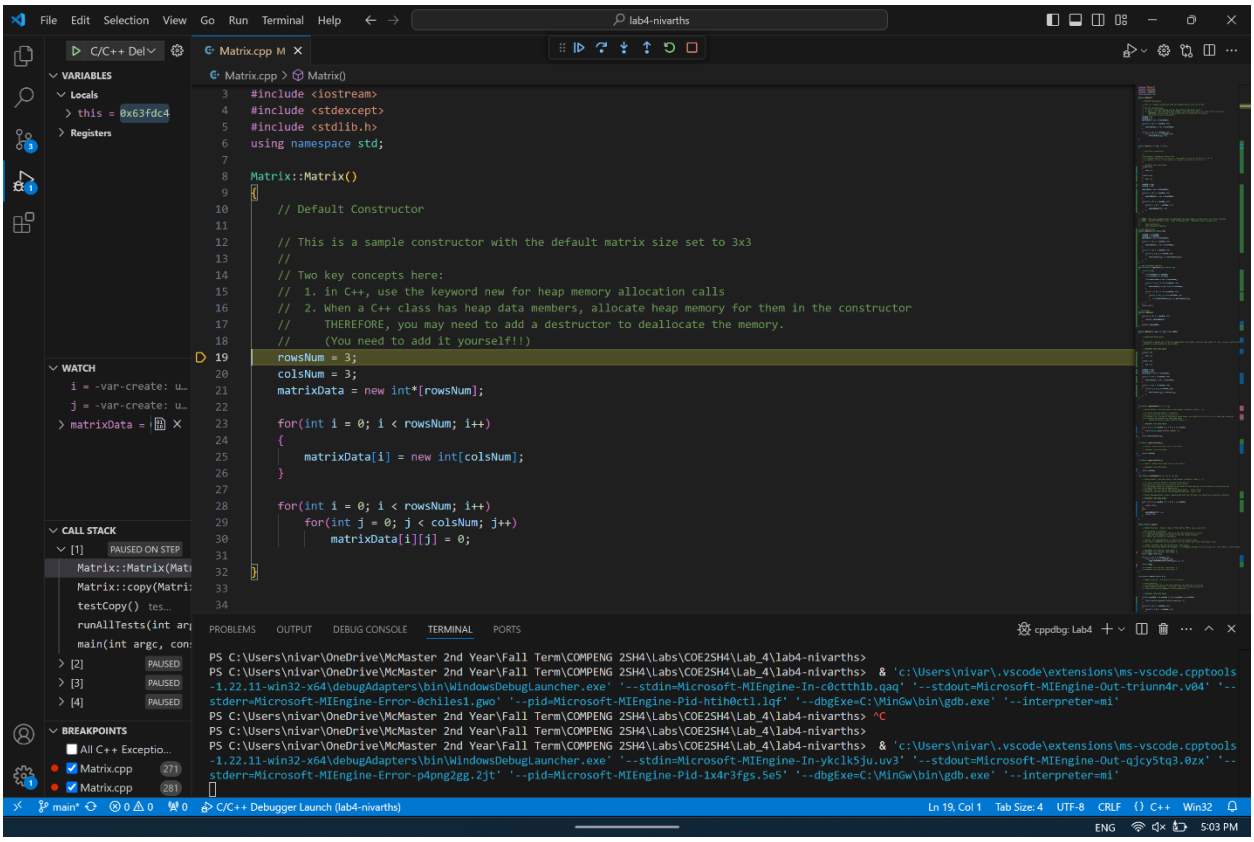

Figure 2: after step into, function goes into the default constructor rather than additional constructor.

Semantic error 1 debug:

To fix this, line 271 must be changed to *Matrix copy(this->rowsNum, this->colsNum)*. This will ensure that while the program is running, the program steps into the additional constructor instead, and an instance of a previous matrix is created with the same dimensions, rather than a default 3x3 matrix.

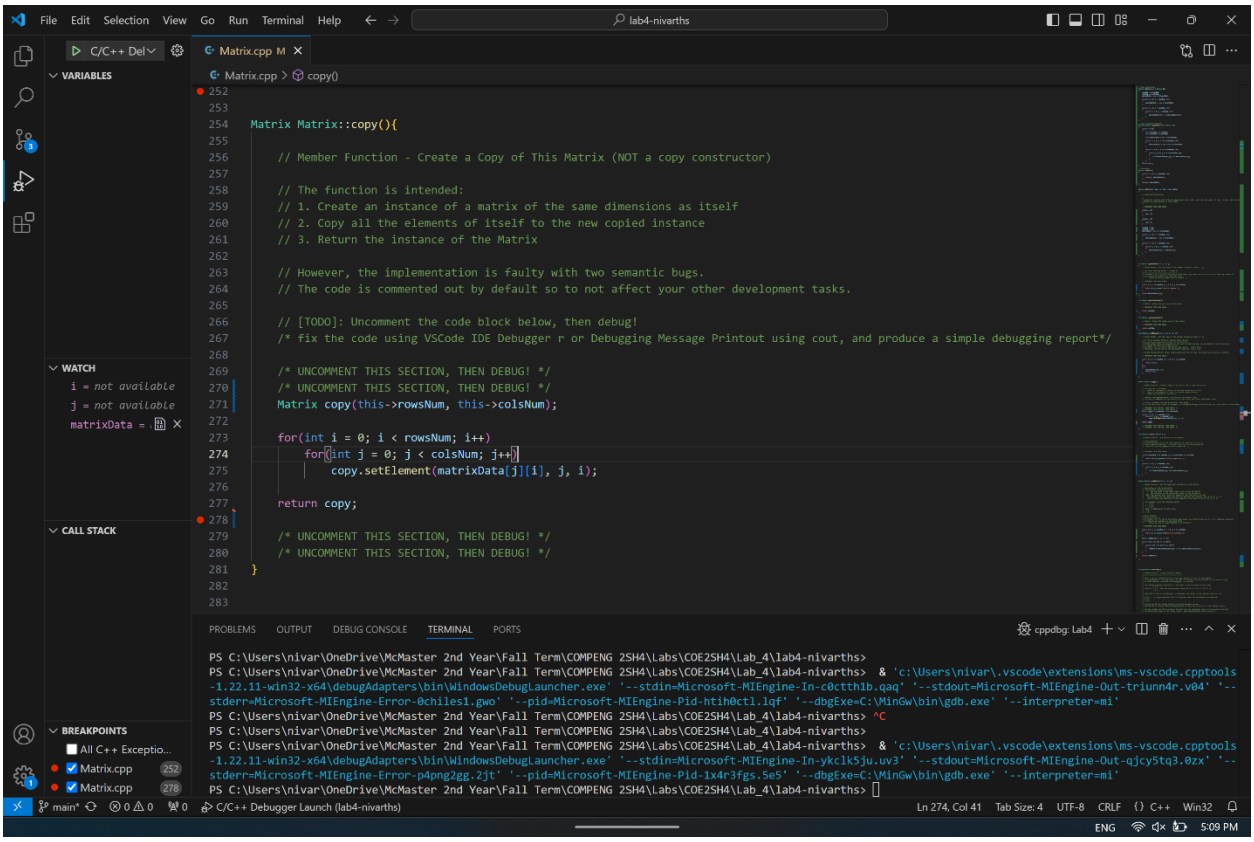

Figure 3: Screenshot showing that line 271 has been modified to now use additional constructor to copy dimensions correctly.

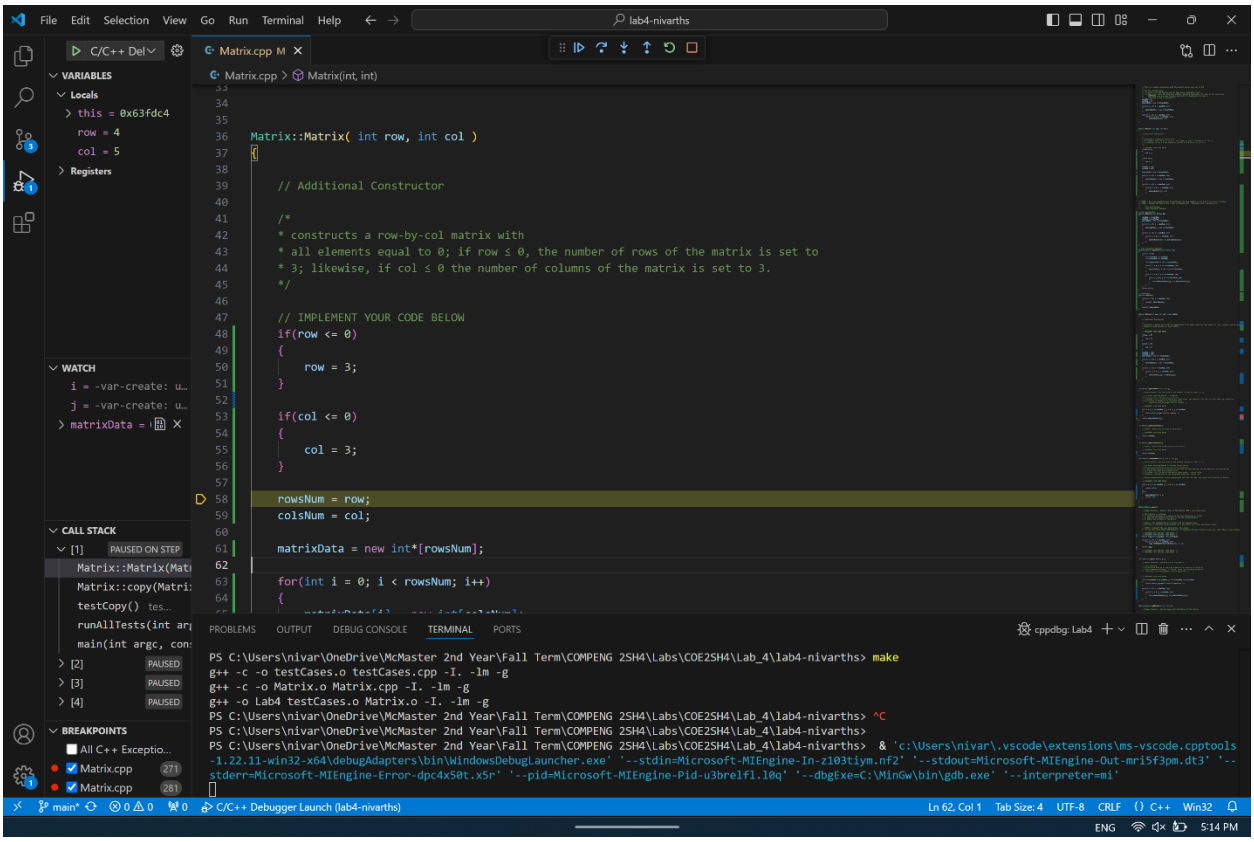

Figure 4: program now steps into and uses the additional constructor to copy dimensions correctly.

Semantic error 2:

The second semantic error present in this function is line 275, where the indices for the set element function are inverted (matrixData[j][i]). This results in segmentation fault and will not return the copied matrix.

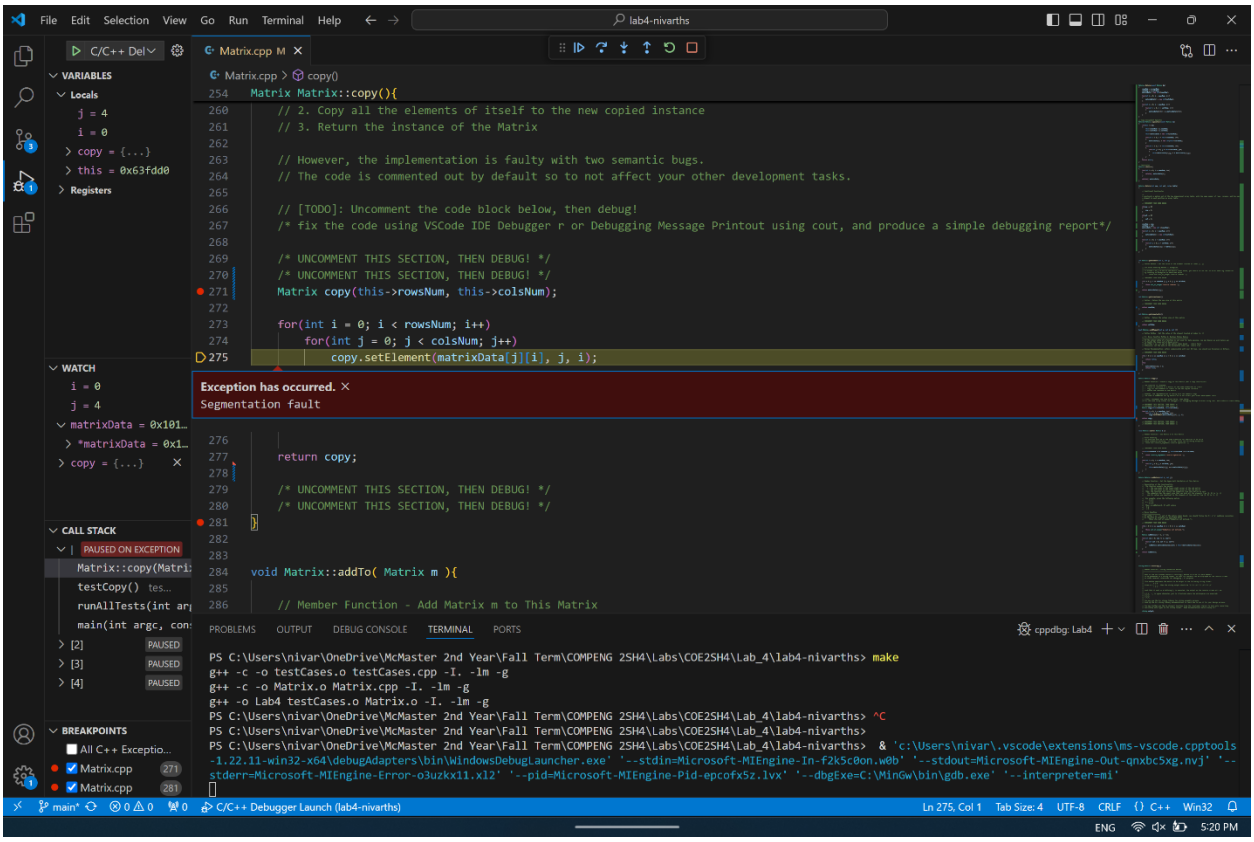

Figure 5: semantic error 2 producing a segmentation fault.

Semantic error 2 debug:

To fix this error, line 275 must be updated so the setElement function has the proper indices, so it can copy the values in the matrix correctly.

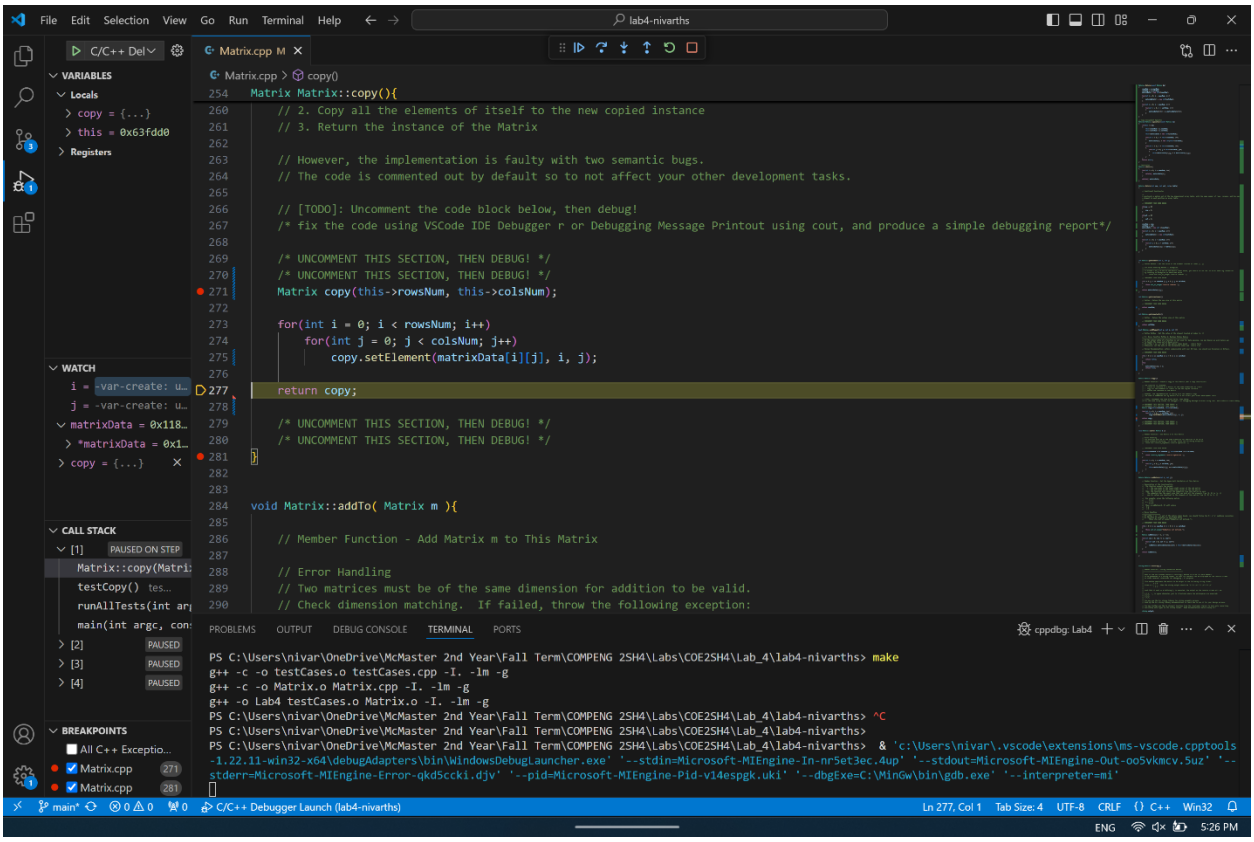

Figure 6: error has been debugged and line 275 has been updated, leading to no segmentation fault and the program moving on to returning the copied matrix.