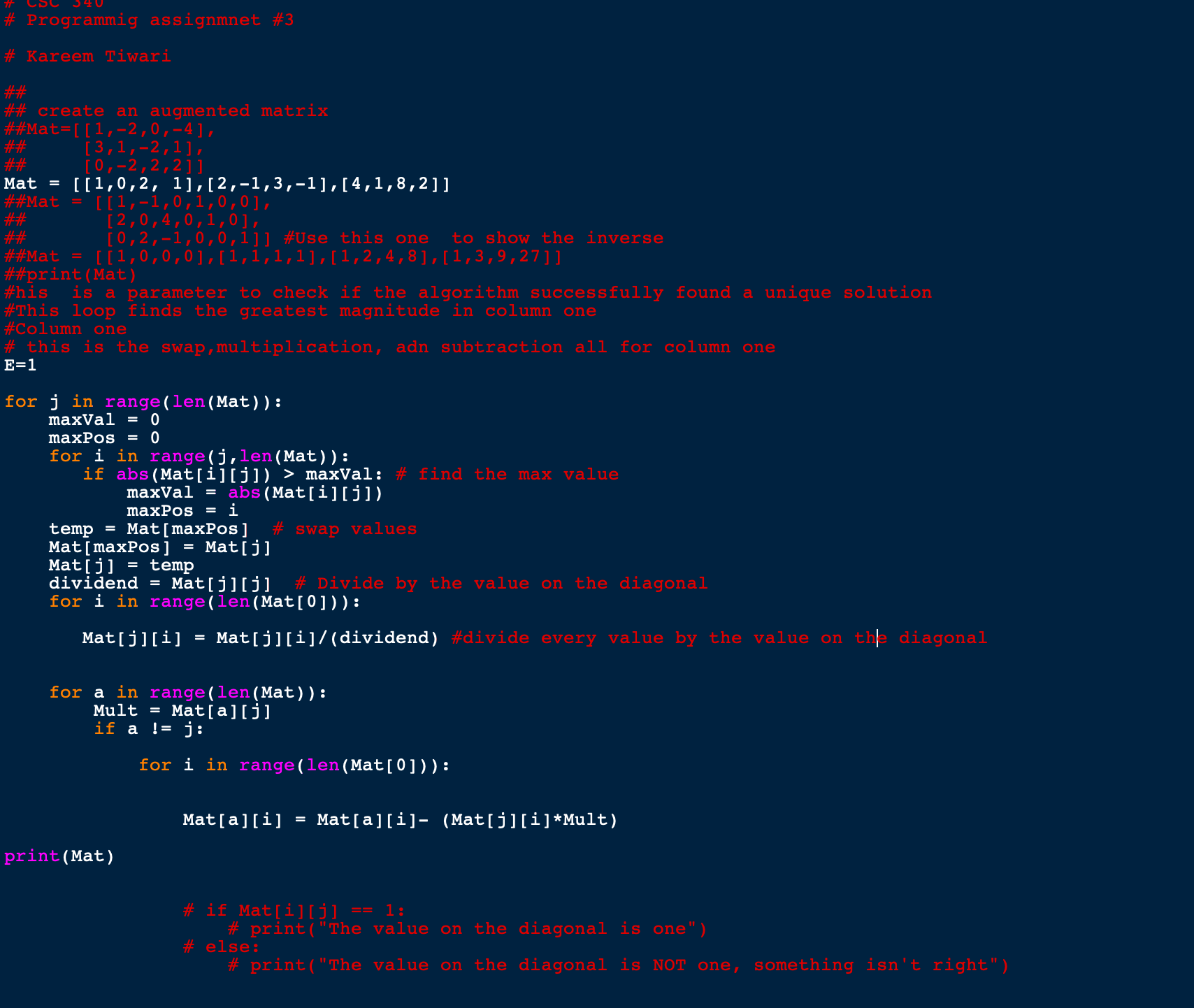
Programming Assignment #2

Gauss-Jordan Elimination

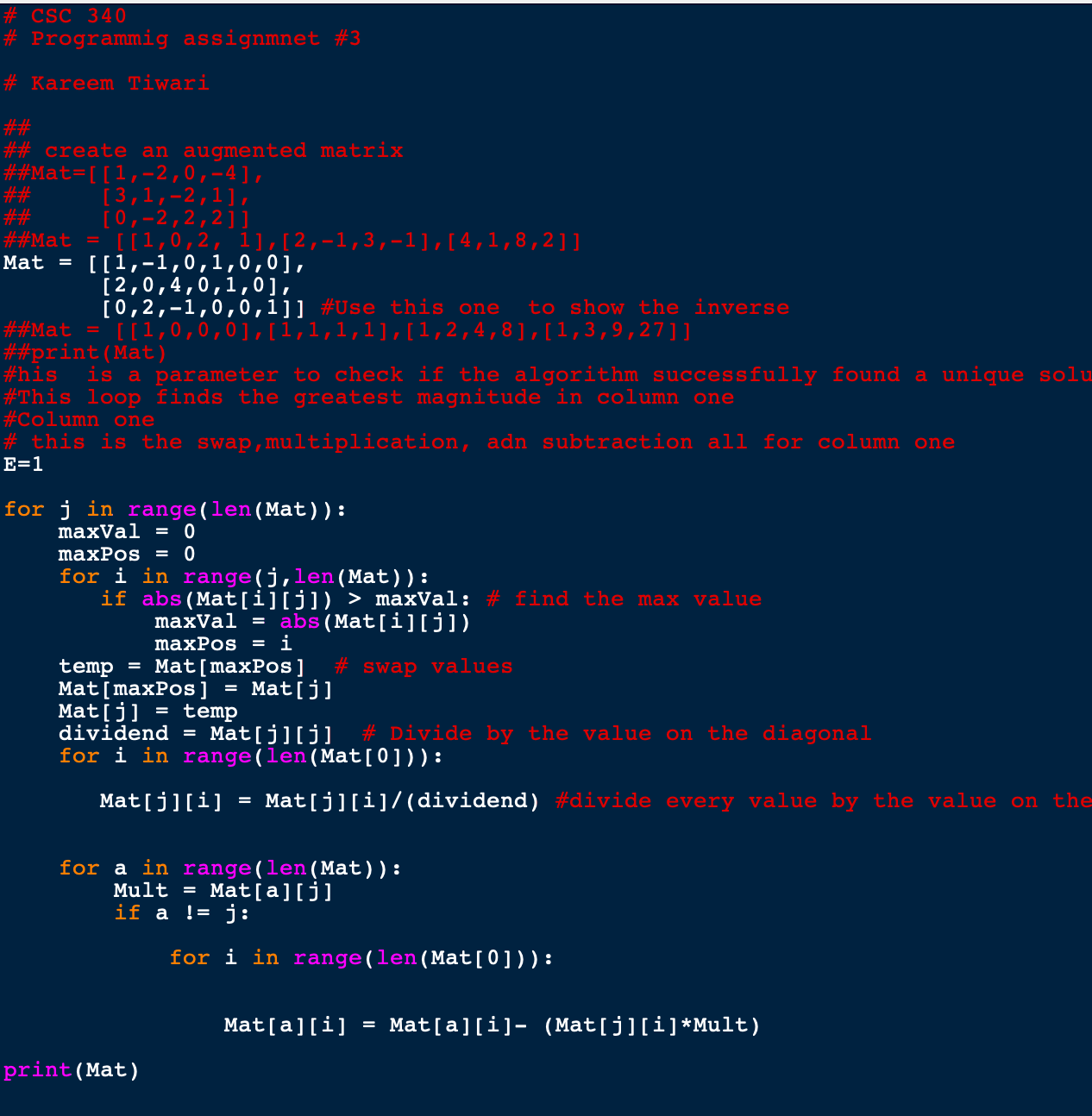
The purpose of this program was to be able to take a system of equations which would be entered as a matrix of values.

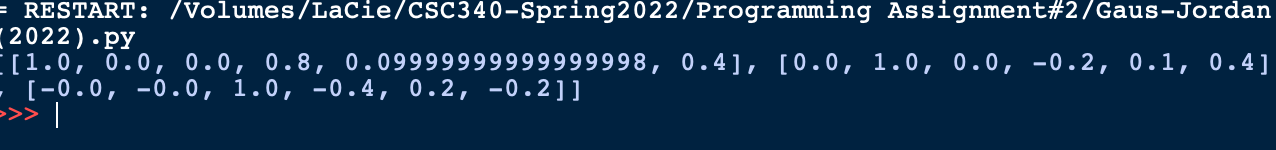
For the Gauss-Jordan program we follow the steps as seen in the code below to get the desired result. The desired result is to check through the matrix and find the max value in the rows and then make swaps for the largest absolute value to be on the diagonal. Then we divide the entire row to get the value on the diagonal to a value of one. Then go through the matrix rows and subtract every value by the values in the row that we made the diagonal value one and then multiply the diagonal value before the subtraction occurs on each row. That gets repeated until the values on the diagonal are all equal to the value of 1. This is all shown in the code below:



As well as creating the Gauss-Jordan program find the identity matrix of system of equations given. We were also challenged to create a Gauss-Jordan program that id dynamic enough to find the inverse of the matrix that is input into the program. The code that I had shown previously is dynamic enough to have the matrix input to find the matrix inverse.

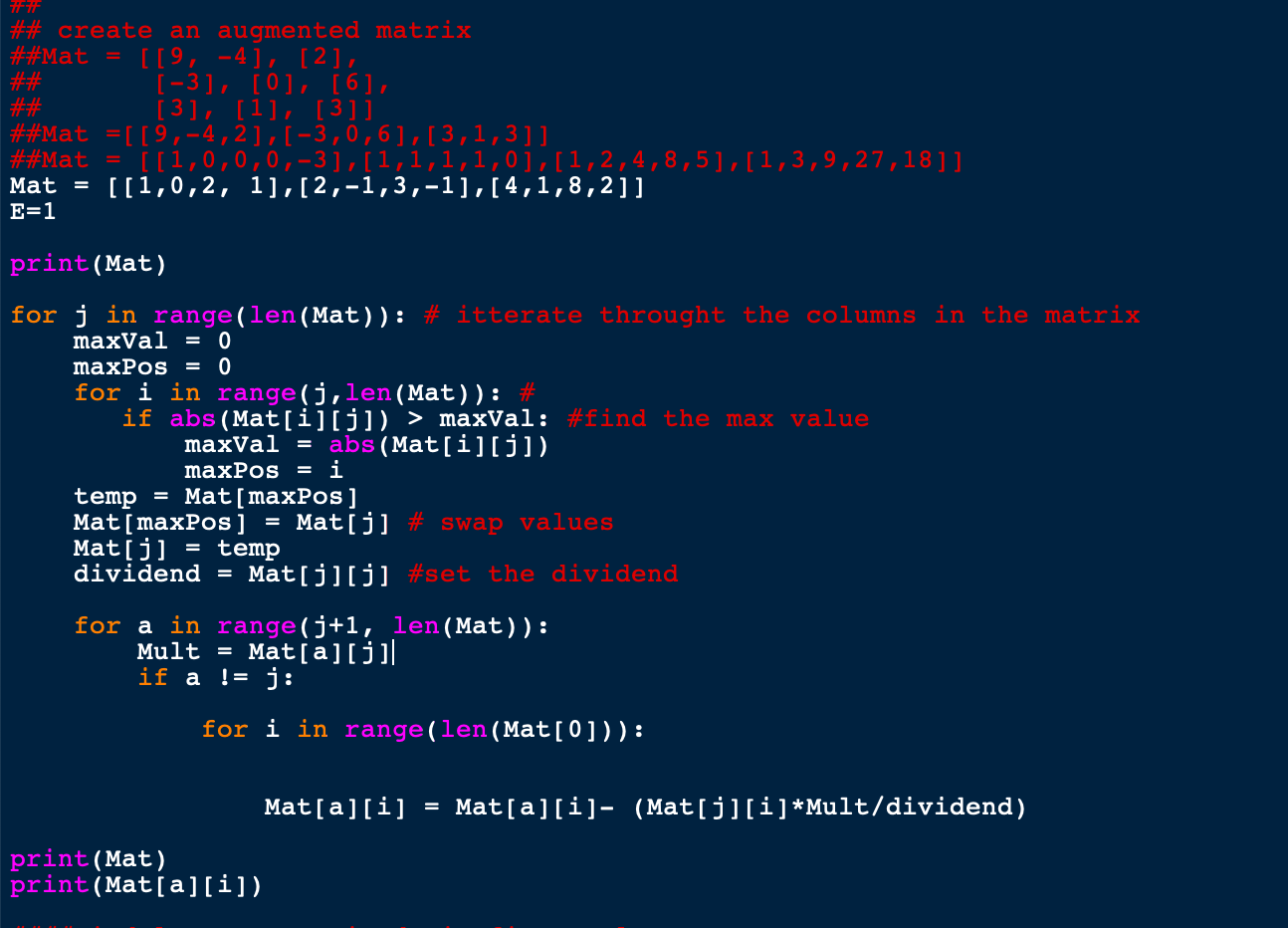
The code implementation as well as the result is below:





For the Gaussian elimination program, we follow just about the same steps:

First, we iterate through the columns in the matrix and then we also found the max value in the column itself. Then we would swap the max value to get it on the diagonal. Next, we would divide to get the ones on the diagonal and to make the upper triangular matrix with the lower triangle being where the zeros reside. We would continue this until we have an upper triangular matrix.



Finally, we created a program that finds the matrix determinant following the Gaussian elimination method. In this program first, we create a counter and then just as the Gaussian elimination we iterate through the columns to find the max value then we swap the rows if the max value is found not on the diagonal. We then zero out all of the values that fall under the diagonal creating the. Upper triangular matrix that is desired.

