Rupinder Kaur

Part 3- Create a sort for 2d matrices. Do a bubble sort

   In my main function, first I declared an array and an integer temp, r, c. After that I print two statements to enter rows and columns and I use the "cin" command for where I put an integer "r" and "c" that I declared before to enter the array elements. And there is another statement I print for entering the array elements for columns. I then created a for loop to iterate through the array and which helps to gather the data. A user can print elements here in an unsorted way. This means that the program could randomly generate an array of size n. For example, put array size i.e. r = 2 and c = 3 and array elements put random values {5, 2, 1, 8, 0 3}. Here there is no need to enter elements in sorted order.

   Next, we wanted to sort the array using the bubble sort method. So, then I need a function to implement bubble sort. For that we use loops (for loop) and swap methods. Bubble Sort is basically the simplest sorting algorithm that works by repeatedly swapping the adjacent elements if they are in wrong order. The swap method will help to sort the array elements that we entered in unsorted order. In the first pass, this algorithm will compare the first two elements of the list {5, 2, 1, 8, 0 3}. and move the value 5 in position of 2, {5, 2, 1, 8, 0 3}. and the process will go on until it's done. If the elements are already in order then there is no need for swapping.

   We began the Bubble Sort function with a for loop. In the first for loop, the i-th path, the last element is already sorted there. The next loop will help in executing the pass. We used 4 loops here because we are doing this for a 2d array. For 1d array we only have two loops. And there is an if statement which sets the list in ascending order. Then next is swap function, that will swap the elements until it's done as i mentioned above. Once it is done, then it exits the loop and we set another for loop which helps in printing the sorted 2d array list. After that the program will run properly and provide a sorted list of arrays and time taken to sort the array list.