/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*TOP DOCUMENTATION\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Program: PaulaAgyemanBubble.cpp

Programmer: Paula Agyeman

Program Description: Write a program swapping a single-dimension array and Bubble sorting a two-dimensional array.

Sample Output:

\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#include <iostream> //\_\_preprocessor directive\_ to include the iostream library

#include <string>

using namespace std; //std is the abbreviation for standard

void swapping(string &a, string &b) //swap the content of a and b that are type string

{

string temp;

temp = a;

a = b;

b = temp;

}

void swapping(int& a, int& b) //swap the content of \_a\_ and \_\_b\_\_ that are type \_\_string\_

{

int temp;

temp = a;

a = b;

b = temp;

}

void display(string arraySt[], int arrayIn[][10], int rsize, int csize) //\_\_ordering\_\_ the arrays

{

for (int i = 0; i < rsize; i++) {

cout << arraySt[i] << " ";

for (int j = 0; j < csize; j++)

cout << arrayIn[i][j] << " ";

cout << endl;

}

cout << endl;

}

void bubbleSort(string arraySt[], int arrayIn[][10], int rsize, int csize) //sorts by the \_\_integers\_\_ array

{

for (int i = 0; i < rsize; i++)

{

int swaps = 0; // swap is used as a \_\_flag\_\_ to detect any swap has happened

for (int j = 0; j < rsize - i - 1; j++)

{

if (arraySt[j] > arraySt[j + 1]) //when the current item is \_\_larger\_\_ than the next... SWAP

{

swapping(arraySt[j], arraySt[j + 1]);

for (int k = 0; k < csize; k++)

swapping(arrayIn[j][k], arrayIn[j + 1][k]);

swaps = 1; //set swap flag

}

}

if (!swaps)

break; // \_\_no\_\_ swap in this pass, so array is sorted

}

}

int main()

{

int rows, columns; // the \_\_actual\_\_ number of rows and columns in the arrays

string arrSt[10]; //create a string array no greater than \_\_10\_\_ elements

int arrIn[10][10]; //create a \_\_10x10\_\_ dimensional integer array no greater than \_\_10x10\_\_ elements

cout << "Enter the number of rows: ";

cin >> rows;

cout << "Enter the number of columns: ";

cin >> columns;

if (rows > 10 || columns > 10) // \_\_validation\_\_check for row and \_\_column\_\_

cout << "There is a limit of 10 elements for rows and columns" << endl;

else

{

for (int i = 0; i < rows; i++)

{

cout << "Enter string element number " << i + 1 << ": ";

cin >> arrSt[i];

for (int j = 0; j < columns; j++)

{

cout << "Enter integer element number [" << i + 1 << ", " << j + 1 << "] : ";

cin >> arrIn[i][j];

}

}

cout << "Array before Sorting: " << endl;

display(arrSt, arrIn, rows, columns); //displays the \_\_unordered\_\_list

bubbleSort(arrSt, arrIn, rows, columns); //\_\_sorted\_\_ the data by the \_\_integers\_\_ array

cout << "Array after Sorting: " << endl;

display(arrSt, arrIn, rows, columns); //displays the \_\_sorted\_\_ list

}

cout << endl;

system("pause");

return 0;

}

//Function swapping defined with different types of parameters is known as \_\_function overload\_\_.