Programming Exercises – Multidimensional Arrays & Sorting

1. Alphabetize a string input. Remember that the value of the characters increase from 'a.'
2. Sort a randomized 2D array of integers using whichever method you wish. You should order the rows by their first elements.
3. Read in a multidimensional array of c-strings from a file and print them to the screen. Additionally, try sorting them alphabetically, and transferring the data to a one-dimensional array of c-strings (so technically a 2D array of characters).
4. Continue or start last week's bonus:

   Write a very simple game where the player, represented by a character, can input the characters w, a, s, or d to move around a grid to get to an object, represented by another character, and can't pass through other objects represented by another character. Additionally, try having the game board inputted from a file, so it can be changed between runs of the program.

   Example board – P = player, W = wall, G = goal:

   ```
   . . . . . . . . . .
   . G . . . . . . .
   . . . . . . W . . .
   . . . . . . . . . .
   . . . . W . . . . .
   . . . . . . W . . .
   . . . . . . . . . P
   ```

   Note: you can use system("cls") to clear the screen.