You are to write a console game in C++ conforming to the following specification.

Up to two players begin on a tiled rectangular board in random positions (single player games should be an option). Players can move up down left and right. If a player bumps into the edge of the board, themselves, or the other player then the game is over.

At the start of the game each player only takes up one tile, but as the game progresses the players will get longer and longer, like a snake. In order to add body segments to your player, and thus be able to have more pieces to try and make your opponent encounter, you have to gather food objects that are placed on the board. You should have up to 6 pieces of food randomly placed around the board at any given time but never less than 1. The food will "respawn" in a new random location at some finite number of turns after it is eaten (say 10 turns) unless all six food items are eaten, in which case one will immediately spawn on the next turn.

Whenever a player runs over a food item, they will add one segment to their body length. The player directs the movement of the snake by using either the arrow keys or the keys WSAD for up down left and right movements with the body of the snake trailing along after it on each move.

Movement is continuous. Once a direction is selected, the player will move in that direction each turn until a new direction is selected. You should provide the user with some sort of speed control that will change the rate at which a snake moves. You are free to do this with a compile time constant or an in game key for each snake. The in game key makes for a more competitive game but is slightly more difficult to implement as it complicates matters if one snake can move faster than the other.

Once the game is over you should display some sort of scoring(for how many items each snake has eaten) and ask if they would like to play again.

Hints: Use the gotoxy function that I provided you in class to move the cursor to particular points on the board where the picture is changing. Only change elements of the board that need to be changed, instead of redrawing the entire board otherwise your screen update will be too slow.

You are free to design this project in any way that you choose but will be graded on the modularity of the design and readability. You will have the opportunity to make use of all the skills that you have been working on in C++ up to this point in the creation of this game.

I suggest that you begin the project by making a single object move around the screen based on the key that is currently pressed. Do not add death on collision or body segments until you have a working version of just the heads moving around.

Be sure to make a list of the steps involved in the process of your design before you begin coding. A plan at the start will save you hours of grief later. You will find it useful to try to write your code in modules and units that are testable by themselves. You can create test programs that test certain aspects of your design as you try to discover tools that will help you make the final functioning product.