Louis Kueh 01078797

Below are the functions that I have used in my program.

```
void InputIntoArray(int(&input)[4][4], ifstream &infile); //function reads values from text file and places them into array
//N.B. infile needs to be passed by reference as fstream objects are not copyable

void PrintNumbers(const int(&input)[4][4]); // function that prints all the numbers in the array
```

The first function is used to read from the text file and put the values into an array, this will be used at the beginning of the program. Then I have a function to print all the numbers in the array. These are used as functions so that the code becomes neater in the main, and allows for easy function calls. (e.g. Printnumbers needs to be called every time the user inputs a shift)

```
void MainShift(int(&input)[4][4]); // This function will have the shift and add Num together to complete the shifting
void AddNum(int(&input)[4][4], int n, int i); // This will be adding the numbers
void Shift(int(&input)[4][4], int i, int  n); //this will be shifting the numbers left/right
```

Here we have the actual shifting of the array. I did the shifting in two parts, firstly adding the two numbers if they are equal, and then shifting them to the left. The two functions AddNum and Shift and combined into one in MainShift, so that it can be easily called

```
void CopyArray(const int(&input)[4][4], int(&copy)[4][4]); // copies array
void RotateS(int(&input)[4][4]); //Shift S
void RotateW(int(&input)[4][4]); //Shifts w
void RotateD(int(&input)[4][4]); //Shifts D
```

My MainShift function performs a left shift on the array, moving all the elements to the left and adding the like terms. So to complete the shifts for up, down and right I decided to rotate the matrix each time and use the same left shift each time to perform the shifts. (e.g for Shift right I would rotate the matrix 180 degrees)

This way I would only need to write one shifting function, which can be applied to all the different shifts. The copy function is to copy the array so that I can rotate the matrix appropriately, and the rotate functions all have the MainShift inside them.

```
void Direction(int(&input)[4][4], string Direction); //Determines which rotate to call (wasd)
```

This function determines which direction we will shift it in, up, down, left or right.

```
int CheckArrayChanged(int(&input)[4][4], int(&copy)[4][4]); // Check if array has changed
int EndCase(int (&input)[4][4]); //pass by value as we do not want to save the changed values
int CheckZero(int(&input)[4][4]);// check if array contains a zero
```

These three functions are to check the end cases. The first case is when the array is shifted but nothing has changed, this is done through the CheckArrayChanged function. If it is true then we do not print the matrix.

Next is the endcase, where all the shifts are applied and checked if there is a zero in the matrix using the CheckZero function. This way we can tell when the matrix reaches the end case.(Game over)

```
void Rand2(int(&input)[4][4]); //Randomly generate
```

This function is to generate a random 2 in any of the zeroes in the matrix.