Project 1

Sliding Puzzle Game

CSC 5 – 40718

Andrew Campbell

Intro:

This is a game based on the sliding puzzle games. It is a 4x4 board equaling 16 spaces, with one spot being empty. The empty spot is what is used to move numbers around until the numbers or ending configuration is made. In this game's case it is listing the numbers in order with the very bottom right corner spot being left empty.
http://www.cs.brandeis.edu/~storer/JimPuzzles/SLIDE/CornellCrossword/KeithArticle2011.pdf

I chose this game because it seemed like it would be a good test of all the things learned in class so far, especially arrays because of the way I store the game board's current values, while at the same time being able to be built using simple ascii artwork and still keep its simplicity for the end user.

How to play:

The objective of the game is that you are trying to put the numbers in order from their current mixed state, in as few moves as possible if you want a challenge. There will be one blank spot on the board that will be used for moving numbers around, only a number directly above/below or to the left/right of the empty spot can be moved into it. The blank spot will then become the moved number's old spot, which can then accept another number to be moved into it and so on. A number that is not left/right or above/below the empty spot cannot be moved and if the user chooses such a number an error will come back letting them know it is an invalid move.

Example of playing:

```
5 ▌ 3 ▌ 4 ▌ 7 ▌
2 ▌ 14 ▌ 15 ▌   ▌
9 ▌ 1 ▌ 6 ▌ 10 ▌
8 ▌ 12 ▌ 11 ▌ 13 ▌
```

Choosing the 7 will have the following effect:

```
5 ▌ 3 ▌ 4 ▌   ▌
2 ▌ 14 ▌15 ▌ 7 ▌
9 ▌ 1 ▌ 6 ▌ 10 ▌
8 ▌ 12 ▌ 11 ▌ 13 ▌
```

From here only the 4 or 7 can be chosen as a valid move due to the empty spot being in the corner.

Pseudo code:

Create array by randoming 0-15 in the array and assigning numbers as for loop iterates.

Print board with current values

User inputs number they would like to move;

Function checks if that is valid move by checking where it is in the array to the 0 (the blank spot) if -/+ 1 it is accept and function swaps the values.

Reprint board using the new values

Checks if current game board matches winning game board, checks the current array against the pre done winning array for this.

If win, print message and save score (moves it took) to file, else repeat and ask for next move.

Methods used:

Cin – line 39

Cout – line 38

Variables – lines 34-37

If statement – line 85

Else if – line 123

Else – line 88

Nested else – line 93

Switch statement – line 40

For loop – line 106

While loop – line 105

Do while loop – line 54

File operations – line 67 – 69

Arrays – line 35/36 setup, then in almost every function are used as input

Void function – line 83

Return value function – line 102