```
//##################################################################//
//            File : 8puzzle                                        //
//      Purpose : Project Files for CPSC3620 : Data Structures &    //
//                Algorithms, as a way to test understanding of     //
//                 searching algorithms with Manhattan Distance.    //
//      Creator : Austin Kothig                                     //
//         Date : December 3, 2016                                  //
//         Files : 8puzzle.cc Board_Tile.h Board_Tile.cc            //
//                Sliding_Solver.h Sliding_Solver.cc Makefile       //
//##################################################################//
```

Referance :
    This is a Puzzle Solver for a 3x3 Sliding Tile Puzzle

    Input is taken in from a single line with no spaces between
    i.e. '012345678' and will store them in private memory like so

    0 1 2
    3 4 5
    6 7 8

    The Goal of the Puzzle Solver is to reach an end state that is
    either provided by the user or is the default solution of '123456780'

    There is error checking for if a user attempts to enter illegal
    characters such as letters or 9, as well as if the length of a
    configuration is too long or too short will return an error
    and ask the user to try again.

    The Algorithm used, uses A* heuristics, with Manhattan Distance as
    its weight.

To Compile :
    - Use the command 'make' in the current directory
    - Can run afterwards the command 'make clean' which will
      remove any extra .o files
    - To remove the executable and any .o files run command
      'make clean-all'