Bug List:

-~~Flagging out of bounds -flag out of bounds crashes program- TF~~

-~~The column bounds works but the row bounds does not on the reveal - crashes program -TF~~

~~-to expand on this, both row and column bounds checking works if you enter a negative number. When you enter too big of a number, column still works, but row doesn’t.~~

~~-when entering number of rows in board creation. The prompt says upper bound is 4, which is incorrect. -TF~~

~~-if you say you do not want to play another game. The program says you are the winner. -TF~~

~~-if you say you want to play another game, also says you win. TS~~

~~-I placed a flag and then revealed and it made my flag disappear but I could not use another flag. I could not finish the game. TF~~

~~-to add to this, it looks like it only happens during the spread. It doesn’t let you reveal a flagged space directly. TS~~

~~-I started a new game after winning and lost all of my flags. TF~~

~~-if you try and reveal a numbered square. You lose a flag. TF~~

~~-if you flag a revealed space, it correctly says to can’t flag a revealed space, but it takes a flag away anyway TS~~

~~-that also happens if you flag an flag space. TS~~

~~-when you try to unflag out of bounds, it crashes if you enter a number that’s too big, it appropriately disallows you from entering a negative number for the row, but accepts a negative number for the column (but doesn’t crash) TS~~

~~-prompts for board size incorrectly say the max is 4 TS~~

~~-this is just an aesthetic thing, but the lines above and below the board are actually the same size as the board TS~~

~~-from an ease of use standpoint, the menu has numbers, but you can’t actually enter a number, so it’s a little confusing. TS~~

~~-if you enter a command in 1 line, (like f 8 8), it works, but still writes the prompts to the screen MS~~

~~-output is: Please enter your row:Please enter your column:---------------------------------------~~

(Note from Tanner)

Just for reference, if you have trouble getting the code to compile like I did, I was able to hack it with the following steps:

1. Navigate to Minesweeper.cpp.
2. Include <cstdlib> to let you use srand and rand.
3. Include <sstream> (we’ll use this later).
4. Go to the very last line of RecCheck { this line: Uboard[Row][Col]=to\_string(Check(Row,Col)) } and directly above it, add:
   1. stringstream strm;
   2. strm << Check(Row,Col);
5. Modify the original line to say: Uboard[Row][Col]=strm.str(); This should correct the to\_string issue.
6. Now the code should be good to go (assuming you only ran into trouble with srand, rand, and to\_string like I did. If you have other problems, you’re on your own).

Final testing checklist:

~~invalid prompt at board creation:~~

~~number~~ decimals round down, decided to ignore

~~character~~

~~string~~

~~board creation:~~

~~too few rows~~

~~too many rows~~

~~too few cols~~

~~too many cols~~

~~too few mines~~

~~too many mines~~

~~correct mine calculation~~

~~invalid prompt at main menu:~~

~~number~~

~~character~~

~~string~~

~~correct prompts:~~

~~R/r~~

~~F/f~~

~~U/u~~

~~C/c~~

~~H/h~~

~~E/e~~

~~reveal out of bounds:~~

~~row: lower~~

~~row: upper~~

~~col: lower~~

~~col: upper~~

~~reveal a revealed space~~

~~reveal a flagged space~~

~~flag out of bounds:~~

~~row: lower~~

~~row: upper~~

~~col: lower~~

~~col: upper~~

~~flag a revealed space~~

~~correct flag count~~

~~flag a flagged space~~

~~correct flag count~~

~~spread reveal a flagged space~~

~~unflag out of bounds:~~

~~row: lower~~

~~row: upper~~

~~col: lower~~

~~col: upper~~

~~unflag a revealed space~~

~~unflag a flag space~~

~~cheat mode~~

~~helper placement out of bounds:~~

~~row: lower~~

~~row: upper~~

~~col: lower~~

~~col: upper~~

~~invalid prompt at helper:~~

~~decimals~~ rounds down, decided to ignore

~~character~~

~~string~~

helper movement:

~~up~~

~~down~~

~~left~~

~~right~~

~~exit~~

invalid

~~number~~

~~character~~

string? allows you to enter a string that starts with a valid character. fix?

~~move helper over flag~~

~~game win~~

~~game lose~~

~~new game prompts:~~

~~Y/y~~

~~N/n~~

new game invalid prompts:

~~number~~

~~character~~

string? allows you to enter a string that starts with a valid character. fix?