**Team notes:**

team notes 09/21/18 in class: All members present

Tim forked the program from team FourFour8's github.

We tested the program a little. Flagging doesn't work? Need to investigate further.

Possible ideas for new features: saving/loading, store high scores in a database.

Planning to meet at 4 in LEEP2.

Use doxygen for documentation generation.

team notes 09/21/18 in LEEP2: All members present

Looks like we don't have access to EECS Databases yet, so discussing possibilities on where to store.

Other things to possibly include:

* timer
* save/load feature? how do we do it?
* Eli to be in charge of doxygen.

Bugs found in new code so far:

* reveal/flag out of bounds
* flag a revealed space, gives error message, but still gets decrements flag count
* if you spread reveal a flagged spot, you lose the flag and can't get it back
* board size prompts don't make sense

4 Phases: Fix bugs 09/26, Implement cheat mode 09/28, Implement new feature 10/03, Testing 10/05

This weekend, individually test for bugs and add any that you find to a google doc.

team notes 09/24/18 in class: All members present

Not everyone got a chance to look for bugs. Take an extra day to find bugs.

team notes 09/26/18 in class: All members present

Tim take bounds issues.

Eli take flag issues.

Matthew take end game issues.

Tanner start looking at cheat mode.

Plan to have everything done by class on Friday so we can start investigating the new feature.

team notes 09/28/18 in class: No Eli

Discussed our changes.

Planned to meet at 1 in LEEP2.

team notes 09/28/18 in LEEP2: No Eli

Group coding sesh.

TODO: modularize? Check with Gibbons to determine whether or not we should invest our time.

TODO: change board size on new game?

TODO: check for memory leaks

team notes 10/01/18 in class: All members present

We decided to drop the idea of using a database due to the difficulty of using a database with C++.

Other ideas:

* sound effects, but they are OS dependent, could lock down OS
* helper guy that walks around with a flashlight and temporarily reveals spaces
* game timer
* number of moves

Will meet at lab room and decide where to go from there. Eli hasn't committed fixed flag bugs. Will go home and commit it, then come meet us.

team notes 10/01/18 in lab: All members present

Group coding sesh.

team notes 10/03/18 in class: All members present

Use an interactive scout mode where you can move the scout around with aswd and he reveals the adjacent spaces. If he steps on a mine, he "dies".

team notes 10/03/18 in LEEP2: No Eli

Group coding sesh.

Tim going to start on implementing scout.

team notes 10/05/18 in class: No Eli

Eli got doxygen setup.

Tim got started on implementing scout.

Plan to meet in LEEP2 tonight and finish up scout implementation.

Started bouncing around ideas for Project 3.

Gantt maker? <https://creately.com/Gantt-Chart-Software>

team notes 10/05/18 in LEEP2: No Eli

Group coding sesh. Scout is almost finished.

Plan to meet at 4 in LEEP2 to finish up.

team notes 10/05/18 in LEEP2: Eli on Zoom.

Group coding sesh. Finishing up helper. Run all the tests.

team notes 10/10/18 in LEEP2: All members present

Finish up documentation.

**Retrospective:**

When we first inherited the program, we ran it to get a general idea of what we were dealing with. We noticed a few bugs right off the bat, so we decided to dedicate some time to testing and fixing bugs before moving on to cheat mode and implementing our new feature. We defined four phases and gave each of them a deadline: find and fix bugs by 09/26, implement cheat mode by 09/28, implement new feature by 10/03, and complete testing by 10/05.

We found bugs and divided them among ourselves based on general area in the code to try to avoid interfering with each other’s code and ease merging all of our branches. We started out well, but due to exams in other classes, we got set back a little. To compensate, we chose to implement cheat mode in parallel with the remaining bug fixes.

Regarding our new feature, our initial idea was to implement a database and scoring system allowing us to maintain a list of high scores. Due to the difficulty of implementing a database with C++, we chose to scrap that idea due to time constraints. We bounced around a few other ideas, such as sound effects or a save/load feature, but we settled on a little helper, which would serve as a sort of mini cheat mode. Even within the little helper feature, we considered several different things, such as how many cells should it reveal, should it die if it steps on a mine, etc. At the end of the day, we chose to just reveal the eight surrounding spaces, including what lies underneath a flag, and have it "hover" over mines.

In retrospect, where as we struggled with communication during the first project, we had much better communication this time around. We devised a much more rigorous plan of attack and allowed ourselves enough buffer time at the end to complete a thorough series of tests.