

Category	Description	Reviewers Comment	Action
Build	Could you clone from Git and build using the README file?	Everything worked splendidly. I was able to easily run the software using the instructions provided by the README. After following through, I was able to see the output and use of the dummy data. It would have been nice to have some control over the dummy data, but I am glad to see it run nonetheless.	Instruction for editing the dummy data was added.
Legibility	Was the flow sane and were variable names and methods easy to follow? Does the code adhere to general guidelines and code style?	<p>It would be nice to see some organization in the file hierarchy. I like that main.c is the entry file, but maybe including a directory for:</p> <ul style="list-style-type: none"> <li>- Distance calculation</li> <li>- Logging</li> <li>- Connection checking</li> </ul> <p>So that it is easier to browse through.</p> <p>Otherwise, code was semantic and easy to follow. I could read it pretty easily.</p>	<p>Testing calculations are in their own folder now.</p> <p>Directories were added for further organization.</p>

Implementation	<p>Is it shorter/easier/faster/cleaner/safer to write functionally equivalent code? Do you see useful abstractions?</p>	<p>I think the code is solid and there is little to criticize. The one thing I can point to is the way the team compiles. The Makefile could use improvement since the same command is being repeated just for different files. Consider: <u>Wildcards</u> which allow for a singular makefile that does not need to be changed. <u>Here is an example</u> in my GitHub if that is better. I used both “.cc” and “.cpp” file extensions though, so you only need one.</p> <p>Additionally, I feel having a singular header file, util.h, with a lot of gcc libraries really increases compile time due to header space. This is probably just me considering <u>Premature Optimization</u> though. In any case, it looks good!</p>	Improved the Makefile.
Maintainability	<p>Are there unit tests? Should there be? Are the test covering interesting cases? Are they readable?</p>	<p>The entire project is a black box test. The output is manual. I guess you could add in some assertions just to alleviate that. I honestly don't see the point in adding unit tests. However, here is <u>a C testing suite</u>. You could also just use <u>gcov</u>.</p>	Added black box tests in the test folder.

Other	Are there other things that stand out that can be improved?	<p>I would love more documentation/comments on the Moving Pattern syntax. Why you chose each character. What each character stands for : Up, Right, Left, Down, etc.</p> <p>I also want to say how cool this project is. It is really really cool. Like, REALLY cool. Doing a capstone in C is hard. I wimped out for C++, and to think that these folks went an extra level, that is a boss move honestly! Additionally, the only things I could think of were project management. The actual code is 100% spot on. I hope this group is patting themselves on the back.</p>	Instruction for editing the dummy data was added.
-------	---	---	---

Category	Description	Reviewers Comment	Action
Build	Could you clone from Git and build using the README file?	Docs were excellent. Project goals and installation very easy to understand. Could have used a little more documentation covering what to expect on data display when you run ./tracker.	Picture of sample data was added to the README.

Legibility	<p>Was the flow sane and were variable names and methods easy to follow?</p> <p>Does the code adhere to general guidelines and code style?</p>	<p>Very descriptive function names and variables. Encapsulating all thread details, including error messages, in corresponding functions could make main.c easier to read. Lots of commented code in display.c makes it hard to read. While lots of comments in main.c, many of them uninformative. Consider one comment to explain a block of code in detail.</p>	<p>Comments in display.c were edited.</p> <p>Comments revised in main.c.</p>
Implementation	<p>Is it shorter/easier/faster/cleaner/safer to write functionally equivalent code? Do you see useful abstractions?</p>	<p>Functionality is abstracted to relevant files. Cannot think of ways to improve code. Very thorough.</p>	<p>N/A</p>

Maintainability	Are there unit tests? Should there be? Are the test covering interesting cases? Are they readable?	Unit tests were listed as WIP. Random unit tests within a feasible interval might be useful if you have a lot of chain reactions.	Testing was added. (Black box and coverage.)
Other	Are there other things that stand out that can be improved?	I really appreciate all the function prototypes, and type definitions are well explained.	N/A

Group 28 - Mingming Su

Category	Description	Reviewers Comment	Action
Build	Could you clone from Git and build using the README file?	Yes, it is very clear about how to clone and run for the project in the Readme file. I can easily clone the file from GitHub and run on the flip server.	N/A

Legibility	Was the flow sane and were variable names and methods easy to follow? Does the code adhere to general guidelines and code style?	Yes, and the code adheres to general guidelines and code style. The code is clear and there is almost no dead code. The file and value names are easy to understand.	N/A
Implementation	is it shorter/easier/faster/cleaner/safer to write functionally equivalent code? Do you see useful abstractions?	Yes, it is shorter/easier/faster/cleaner/safer to write functionally equivalent code. For each function, it is short and easy to learn. They have comments to explain the function which is really useful to help to understand.	N/A
Maintainability	Are there unit tests? Should there be? Are the test covering interesting cases? Are they readable?	Yes, In the readme file shows that run “\$ ./testing” to test.	N/A
Other	Are there other things that stand out that can be improved?	It is a really great project, and I like it. For me, I don't think there is anything that needs to be improved.	N/A (yay!)

Group 28 - Julie Kuang

Category	Description	Reviewers Comment	Action
----------	-------------	-------------------	--------

Build	Could you clone from Git and build using the README file?	Yes, I could clone the project folder to the flip server and make files using the instructions.	N/A
Legibility	Was the flow sane and were variable names and methods easy to follow? Does the code adhere to general guidelines and code style?	All the files are easy to follow. Code style looks good and makes the files clean. I would suggest adding some general descriptions to each method or .c file. That would help the reviewers understand the code better.	General file descriptions added to documentation.
Implementation	is it shorter/easier/faster/cleaner/safer to write functionally equivalent code? Do you see useful abstractions?	The code is short and easy enough and it does the job as expected.	N/A
Maintainability	Are there unit tests? Should there be? Are the test covering interesting cases? Are they readable?	Unit tests are mentioned in the README file but I didn't find the file. I think the team is still working on it and will have it done in the future.	Testing has been added.
Other	Are there other things that stand out that can be improved?	This team has done a good job so far. There's no other thing I can think of that they need to improve.	N/A

Category	Description	Reviewers Comment	Action
Build	Could you clone from Git and build using the README file?	Yes, all codes are available on Github, it can be cloned and built correctly using README file.	N/A
Legibility	Was the flow sane and were variable names and methods easy to follow? Does the code adhere to general guidelines and code style?	Yes, the flow is sane and the methods are easy to follow. Each part of code have comments to help the user understand.	N/A



Implementation	is it shorter/easier/faster/cleaner/safer to write functionally equivalent code? Do you see useful abstractions?	The code is short and clean, I didn't see useful abstractions.	N/A
Maintainability	Are there unit tests? Should there be? Are the test covering interesting cases? Are they readable?	Yes, there are functions to check the connection, which is easy to read. It's should be there so they can figure out what is going wrong when the program can't run correctly.	N/A

Other	Are there other things that stand out that can be improved?	It can be great if I can have more guides on how to execute the program, maybe add more details in comments? Overall the project is really great.	More documentation has been added to the “docs” folder in the project.
-------	---	---	--

Category	Description	Reviewers Comment	Action
Build	Could you clone from Git and build using the README file?	The directions in the read me made sense and on top of that did a great job explaining what the product is. This was especially useful when actually running the program, as at first, I was confused on what exactly was happening. I had one issue cloning and running the program and that came from the ./testing for unit testing. I assume the WIP above it means that it is work in progress though, so I guess it is not an issue.	Clarified run instructions on Readme.

Legibility	<p>Was the flow sane and were variable names and methods easy to follow?</p> <p>Does the code adhere to general guidelines and code style?</p>	<p>Very neatly written code by my standards, it was easy to follow and read. I found that the variable names also contributed to the ease of understanding at times. The addition of function headers and the comments also help the reviewer understand exactly what is going on flow wise throughout the code. Files that don't have comments still make sense but would greatly benefit from having them.</p>	<p>Comments were added.</p>
Implementation	<p>is it shorter/easier/faster/cleaner/safer to write functionally equivalent code? Do you see useful abstractions?</p>	<p>Looking through the code it seems every C function is well optimized and specific to the role it is trying to accomplish. At the moment, I can't think of a way to write equivalent code that runs faster. I do see some useful abstractions when looking through the code.</p>	<p>N/A</p>

Maintainability	<p>Are there unit tests?</p> <p>Should there be? Are the test covering interesting cases? Are they readable?</p>	<p>There was unit testing mentioned in the progress, and I do believe it is a project that could benefit from unit testing. From what I can tell it is still a work in progress, but it is good to see that unit testing is going to be implemented.</p>	<p>Testing has been added.</p>
Other	<p>Are there other things that stand out that can be improved?</p>	<p>I think CS team 28 did an excellent job with project, but I did have trouble reading the information after running ./tracker. I found the information would flicker on screen, this might be something on my end, I just wanted you to have a heads up. If it is working as intended, then feel free to ignore this comment.</p>	<p>N/A (flicker is normal)</p>