

When I started this assignment I had very little experience in C, having only completed Hayo's C/C++ course last year with a focus on abstract data structures.

The following diary documents the processes I went through to develop cgull and the successes and failures I experienced along the way.

Done

- Learnt about safe string parsing from `strtok_r` to `strncmp`, `strncat`, `strdup` etc.
- Learnt about using multiple files in C, header includes, guards, general best practices
- Wrote a lot of core data structure definitions, requests, responses etc.
- Started writing the core architecture, linked lists of 'request handlers'
- Learnt about print formatting, `vsprintf`, `sprintf` etc.
- Learnt about variadic arguments
- Made a custom logger with different granularity levels, using formatting and string parsing knowledge
- Wrote HTTP parsing system to extract request type and path
- Learnt how sockets work, made a simple ping server and expanded this to read in http requests from clients
- Combined parsing system and sockets to start serving a string defined `index.html` to all requests
- Wrote custom request handling system building on previous request handlers allowing custom routing e.g. `route(GET, "/hello", text("Hello World!"))`;
- Learnt about file I/O and started reading html files into buffers
- Expanded file system into `server_static.c` to start reading files and serving them properly based on URI
- Had the basic single threaded HTTP server working, with lots of segfaults and tears
- Learnt about threads, `pthread` and about how to use function pointers
- Added in threading and by some miracle the number of segfaults didn't go up and it appeared to work first time
- Went back to parser and added internal URL rewriting from null uri to `index.html`
- Started learning Docker, had a basic config to spin up an ubuntu container and install gcc etc.
- Used `FROM` directive in Docker to use a pre-made GCC image from docker hub, had code copying in and compiling
- Finished the docker system to allow code to be copied in, compiled, run and accessed on the host system
- Learnt how Makefiles work (very basic) and added docker commands to these (probably not best practice)

- Added HEAD request support
- Went back and did a lot of code clean up and hopefully fixed most memory errors
- Wrote a lot of comments, this basically took forever but I identified some things that needed to be fixed

Still need to do

- Make sure everything is fully thread safe and memory safe, the baptism of fire of learning this stuff from scratch means there still could be issues.
- Get the bloody content types right (HAVE TO DO THIS BEFORE SUBMISSION)
- Start working on `rewrite("/", "/index.html");` directive