CS118 Project1 Report

Team Members:

1. Yuanqi Li, 504759910, lyuanqi@cs.ucla.edu
2. Zhengkai Zhang, 604582162, kirkzhang49@gmail.com

High-level Description:

The server code is written on top of the provided example code.

The server is designed in a way such that once the server is started, it will first establish a socket connection on the specified port. Once a connection is successfully established, the server program will keep listening for requests on the socket connection. For every HTTP request received, it will parse the requested file name and file type. If both the file name and file type are valid, the requested file will be returned through the same socket connection. Otherwise, a "404 Not Found" status will be sent back to the client.

Difficulties:

The most difficult problem we faced was that the compiler on the virtual machine was very outdated: gcc (Ubuntu/Linaro 4.6.1-9ubuntu3) 4.6.1. This makes it impossible for us to use some recent powerful C++ functions, because the compiler is not C++11 compliant. For example, when we want to parse the file name from the Http request, it would be very easy if we could use <regex> from C++ standard library, but since it was not compliable on the virtual machine, we had to switch to <regex.h> from the C library, which was very inconvenient.

Instructions:

1. navigate to the folder that contains the makefile and the cpp files
2. execute “make clean” on terminal/console
3. execute “make” on terminal/console
4. execute “./cs118_webserver [port#]” to start server program
   e.g. "./cs118_webserver 5001"
5. with any browser, connect to server by entering URL:
   "http://localhost:[port#]/[filename]"
   e.g.: http://localhost:5001/index.html

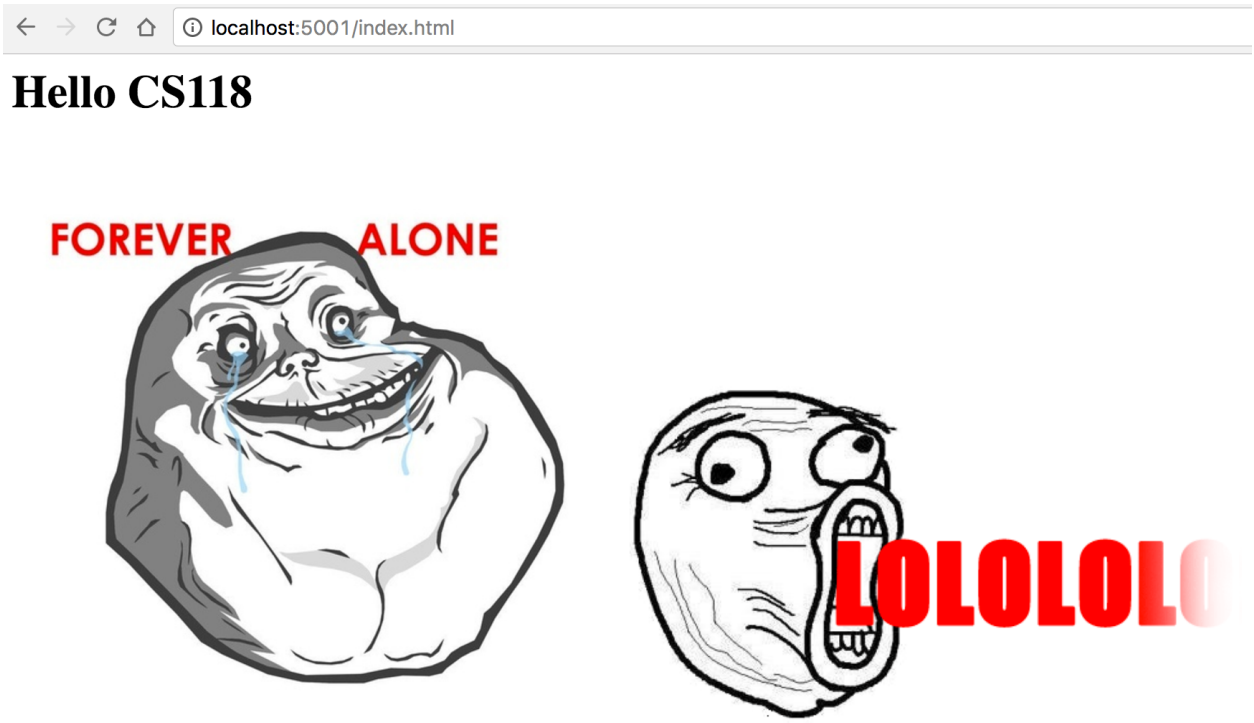
Sample Outputs (Part A):

HTTP GET Request for index.html file:

```
GET /index.html HTTP/1.1
Host: localhost:5002
Connection: keep-alive
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_11_6)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/55.0.2883.95
Safari/537.36
Accept: text/html,a
```

Sample Outputs (Part B):

Using Chrome as client (screenshot):



---

index.html file:

index.html

```
1  <html>
2  <header><title>Index File</title></header>
3  <body>
4  <h1>Hello CS118</h1>
5  <img src="forever_alone.jpeg" alt="" width="30%"/>
6  <img src="lol.gif" alt="" width="30%"/>
7  </body>
8  </html>
9
```