Implementation of an HTTP Proxy Server

In this assignment you would Implement your own HTTP proxy server in C/C++. The HTTP proxy sits between the client and the server. In the simplest case, instead of sending requests directly to the server the client sends all its requests to the proxy. The proxy then opens a connection to the server and downloads the object on behalf of client. The proxy server then sends the object to the client. The proxy acts like an HTTP client (to the remote server) and an HTTP server (to the client). Proxy server should perform the following tasks.

1. Single threaded proxy server: Create a proxy server on a specific port which receives HTTP request from a client (use HTTP client from your previous assignment). Create another socket to the intended remote server by taking the HTTP payload from the request. Send the response from the remote server to the client. The proxy server should be able to handle the “Head,” “GET,” and “POST” requests. You can use the code in your previous assignment for processing the HTTP header. (10 Points)
2. Multi threaded proxy server: Extend the program in the previous step, to handle concurrent requests by the clients. You can test it by configuring your browser to use your proxy server. (10 Points)

3. Caching: Implement a caching mechanism in the proxy server. The proxy server should fulfill the future requests for the same content. You need to check the cacheable field in the HTTP response to decide if the content can be cached. If it is cacheable, store it locally with the object link as index to the file. Evaluate the page/object load time when content is available and not available in the cache. (5 Points)
4. Content statistics: Implement two simple web statistics in the proxy server. E.g., popular website accessed by users, list of websites that uses cookies, etc.