Objectives

To understand client-server communication via sockets.

To explore basic structures of HTTP messages.

To understand basic functionalities of a Web proxy server.

Project Requirements

In this project, you will implement a Web proxy server. Please refer to socket programming assignment 4 of the textbook for more details on the functionalities of a Web proxy server and the skeleton code in Python. For this project, you are required to implement the following functionalities:

Receiving HTTP GET requests from a popular web browser (like Chrome, IE and Firefox), then display the HTTP GET requests.

Forwarding the requests to the origin server if this is the first time the objects are requested.

Receiving the corresponding HTTP response from the origin server, displaying the response, and storing the response.

Forwarding the response to the browser (the client).

If the requested object is already cached on the proxy, i.e., a cache hit occurs, the proxy just retrieves the object related response, displays it, and sends it back to the browser (the client). You do not need to implement any replacement or validation of the cached object response.

Handling multiple HTTP requests concurrently, i.e., spawning a new thread to serve each new request.

Handling errors when a client requests an object which is not available, or the proxy server get errors while reading the request from the client or the response from the server. In such cases, the proxy should send a corresponding status message to the requesting client.