CENG393 Computer Networks
Labwork 9

1 Socket Programming: HTTP Client

1.1 Definition

HTTP (Hyper Text Transfer Protocol) is a standard protocol used mainly for transmission of web page contents. By default, it operates on TCP port 80. HTTP operates by request and reply messages. The requests are usually sent by web browsers and specifies which resource is requested from the server. The server responds to this request with a header and reply content.

1.1.1 Request

A typical HTTP request is of the following form:

```
GET /index.html HTTP/1.0
```

This request is made of three fields: HTTP method (GET, POST, HEAD...), requested source (index.html, media/logo.jpg...) and the HTTP protocol (HTTP/1.0, HTTP/1.1, HTTP/2.0). Every HTTP request must be followed by two newlines (\n\n or sometimes \r\n\r\n).

1.1.2 Reply

```
HTTP/1.1 200 OK
Date: Wed, 27 Dec 2017 20:06:59 GMT
Server: Apache/2.4.18 (Ubuntu)
Vary: Accept-Encoding
Content-Length: 892
Connection: close
Content-Type: text/html; charset=UTF-8

<html>
  <head><title>Sample Web Page</title></head>
  <body><h1>Hello World!</h1></body>
</html>
```

This reply has two fields: reply header and reply content. HTTP headers usually contain information such as reply status code, reply length, server information, etc. Some of the most popular status codes are “200 OK” and “404 Not Found”. Reply content may contain the requested source or error messages if a problem occurs. Headers are seperated by contents with double newlines.

1.2 Exercise

Study and execute the following program. Then modify it such that contents of a requested source are stored in appropriate files.

1.3 http.c

```
#include <ctype.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <arpa/inet.h>
#include <stdio.h>
#include <string.h>
#include <netdb.h>

#define SIZE sizeof(struct sockaddr_in)

int main() {
    int sockfd, nread;
    short int port = 80;
    char buf[24000], address[256], request[256];
    struct sockaddr_in server;

    printf("Enter address of the server: ");
    scanf("%s", address);
    server.sin_family = AF_INET;
    server.sin_addr = *((struct in_addr*) gethostbyname(address)->h_addr);
    server.sin_port = htons(port);
    memset(&(server.sin_zero), 0, 8);

    if ((sockfd = socket(AF_INET, SOCK_STREAM, 0)) == -1) {
        perror("socket");
        return 0;
    }
    if ((connect(sockfd, (struct sockaddr *) &server, SIZE)) == -1) {
        perror("connect");
        return 0;
    }

    strcpy(request, "GET / HTTP/1.0\r\n\r\n");
    send(sockfd, request, strlen(request) + 1, 0);

    nread = recv(sockfd, buf, 24000, 0);
    buf[nread] = '\0';
    printf("The response received is:\n%s", buf);
}
```