Networks Assignment 1 Client-Side.

Made By: Mohamed Said Ibrahem.
ID: 57

Part 1: Client Side.

handlegetresponse(): Reads the file in the response from the server and writes it in a file in the current directory of the client.

handlepostresponse(): checks the response from the server if the status code is ok and sends the file existing in the current directory of the client to server.

startConnection(): Handles the http requests pool with creating sockets and parsing each command.

isOk() : checks if the response has a status code of 200 (ok)

```c
void handleGetResponse(char* filename, int socket);

void handlePostResponse_Request(char* filename, int socket, int fifd);

int isOK(char* msg); //status

void sendFileToServer(char *filename, int socket, int filefd); //post request

void startConnection(char * command, char* command_type, char* file_name, char* host_name, int port_num, int sock);

void print(char * str);

int get_file_size(char * filename);
```

Part 2: Server Side.

getResponse(): handles the get request of the client and sends the data back to it.

postResponse(): handles the post request of the client by receiving the data and storing it in the diectory of the server.

SendBytes(): Send files to client.

SendMessageToClient() : send the response of the request on the socket including the status code.

```c
void interruptHandler(int signal);

void responseForClient(int sckt);

void sendMessageToClient(int sckt, int code, int file_size);

void postResponse(int sckt, char *filename, int file_size);

void print(char * str);

void getResponse(int sckt, char* filename, char* fileType);

void sendBytes(int sckt, int fileSize, int filed);

void DieWithError(char *errorMessage) {
    perror(errorMessage);
}
```

Performance Evaluation:

1. The Time Taken.

2. The number of Requests.

3. The Total Files sizes.

Handling Multiple Clients:

Using select() method to watch the ready file descriptors to avoid the overhead of context switch of using threads as well as using threads.

HTTP 1.1

Handling multiple requests on the same connection