CSE 344 Final Project

Important: For using client we should change folder from client.c .We should change clientFolder what is the clients folder path.

```
250
251         off_t fileSizeLimit = 1024*1024*1024;
252
253         directory = (char*) malloc (sizeof(argv[1
254         strcpy(directory,argv[1]);
255         port = atoi(argv[2]);
256         char *clientFolder = "clientfol";
257
258         char logFilePath[PATHS];
259         strcpy(logFilePath,clientFolder);
260         strcat(logFilePath,"/");
261         strcat(logFilePath,"logfile.txt");
262         FILE* logFile = fopen(logFilePath, "w");
263
```

Firstly I created a fileInfo structure.

```
struct FileInfo {
    char name[PATHS];
    char parentName[PATHS];
    time_t modifyTime;
    off_t size;
    int isDirectory;
    int flag;
};
```

unlink_cb and rmrf for deleting folders both empty or full.

name is the name of file
parent name represent parent directories of file or directory
modifyTime is time the file or directory lastly modified.
Size is the size of file
isDirectory is control for if its a file or directories
flag is for control files in subdirectories. If code once compared this file with other client files then it makes flag 1 to don't compare again server files.With this same files on different directories can be understanded by the program.

In scanDirectory function,collects information about the files and directories within the given directory, storing it in an array of FileInfo structures. By opening the directory and iterating through its entries, excluding the current and parent directories, the code fills details such as file names, modification times, sizes, directories or not etc. The function handles both files and directories by either continuing the recursive scan or skipping files that exceed a given size limit. With this exploration of a directory hierarchy completed succesfully.

In writeFile function client or server uses socket for reading from other side and creates the file.It write that file all information what is read from socket.

In sendFile function client or server uses socket for writing from other side contents of file.It sends all things in file to the other side with socket.

In printDirectoryChanges function, there is three operation;

Finding Newly Added Files and Directories: It compares each file and directory in the currentFiles array with the previousFiles array. If a file or directory is found in currentFiles but not in previousFiles, it founds a newly added file.It prints info about file and operation.

Finding Deleted Files and Directories: Similarly, it compares each file and directory in the previousFiles array with the currentFiles array. If a file or directory is found in previousFiles but not in currentFiles, it founds a deleted item. It prints info about file and operation.

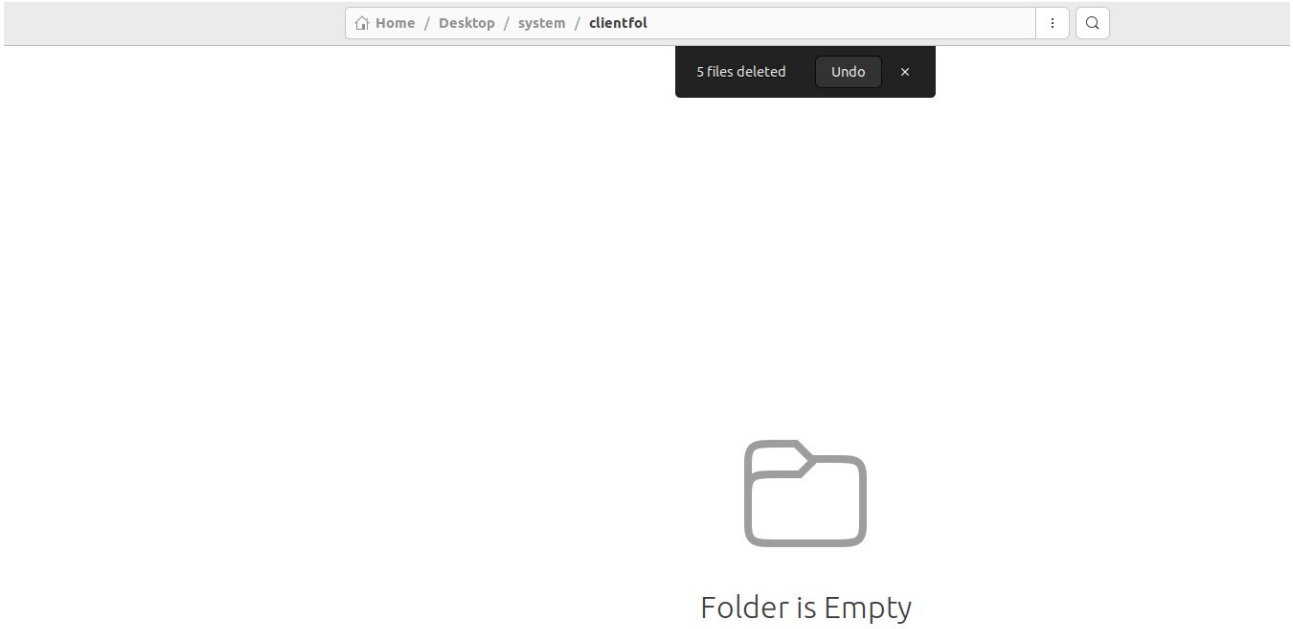
Finding Modified Files: For each file in the previousFiles set, it compares it with each file in the currentFiles array based on the file name and parent directory name. If a match is found and the modification time of the file differs between the two array, it is considered a modified file. It prints info about file and operation.

In server.c and client.c file;
Firstly I am fullfilling required values from arguments. I am scanning directory of server for hiding starting files of server. Then with arguments information socket created. Connect client and server socket for communicating. Then starting communication between server and client. Firstly it send size of server files and content of files. In client side, client too make preparations for connecting socket and then connects to the server. It scan his files to for comparing server files and client files to find should be deleted files and added files. It gets from server content of folder of server. It's type is struct FileInfo so it can provide all information for client needs.After this part, client compares it's files and server files. If client file name,size is equal and client file is first time comparing with the server file, this means that there is one same file. This file should not be deleted.So index of this file is hided in
++countOfSameFiles; Increases count of same files.
tempArrayServer[i]=1; Hides server file array input
tempArrayClient[j]=1; Hides client file array input
clientFiles[j].flag=1; Make client file compare for prevent ambiguity with other files.
If file is not in client directory, then this means that that is a different file so differentfilescount must be increased. We found same and different files so now we should delete different files. Firstly we hide our current path to come back here again.Then find different file and go to directory of it with the help of fileINfo.parentname and delete it.Then come back to the previous directory.With this we delete all different files.We get indexes of different files too. We hide informations about different files in filesWillCoppiedFromServer for requesting them from our server. Then we communicate with server and request this files. Then in server side, we search this files and folders with sended informations.We use isDirectory for founding directories. Then firstly we go to the client directory and create directories server had but client does not had.We use same operations with deleting instead of now we are creating folders with mkdir.Additionally we use paretnName for creating directories succesfully.Then we send server to succes message. After that, we sent client to file it should create. Client creates file and send request to the server for entries of file with writeFile function. Server send entries of file with sendFİle function.After that if operation succes, we send server succes again.And we succesfully copy files and directories from server to client. After that both client and server sides, we create an infinite loop for understand modified,deleted and created files and folders. I could not make

project with threads and I could not send files from client to server or server to client. I can only find changes,deletes,created files. I only have syncronization in beginnig. Synching client directory with server directory.

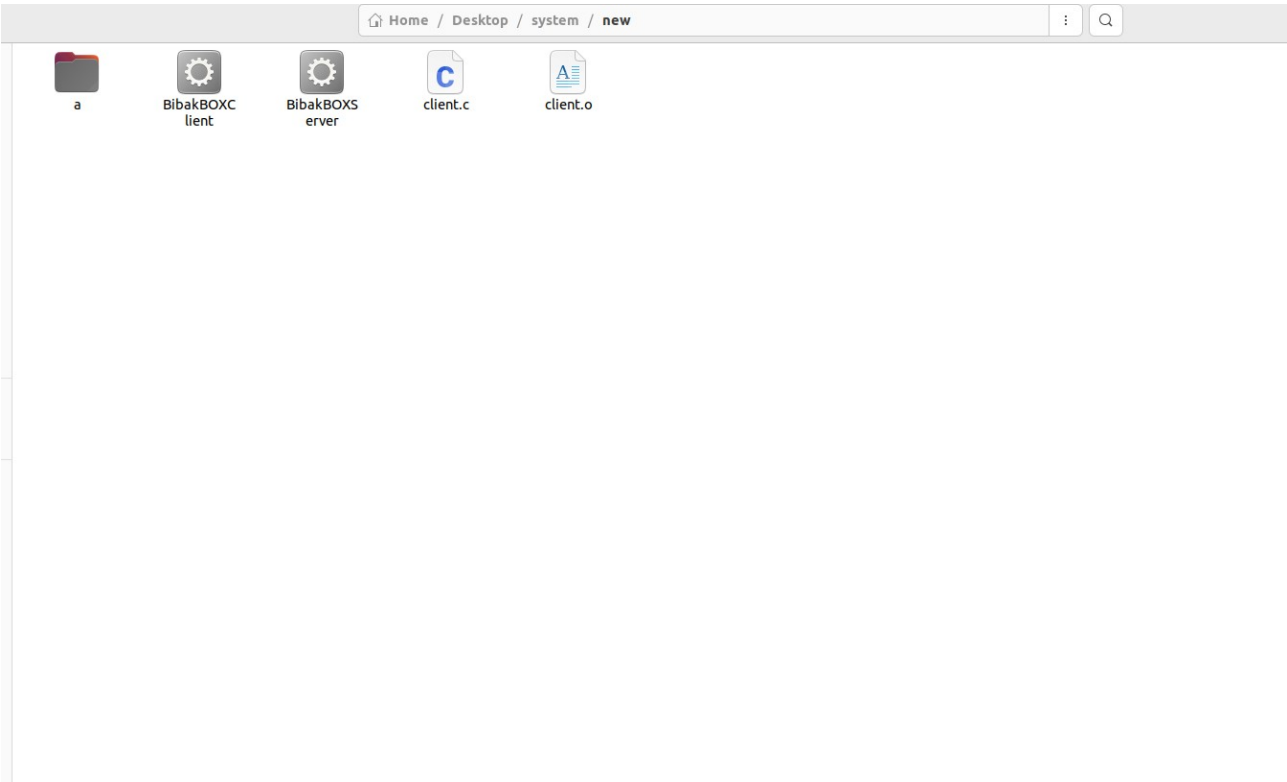
Important: For using client we should change folder from client.c .We should change clientFolder what is the clients folder path.

```
250
251     off_t fileSizeLimit = 1024*1024*1024;
252
253     directory = (char*) malloc (sizeof(argv[1
254     strcpy(directory,argv[1]);
255     port = atoi(argv[2]);
256     char *clientFolder = "clientfol";
257
258     char logFilePath[PATHS];
259     strcpy(logFilePath,clientFolder);
260     strcat(logFilePath,"/");
261     strcat(logFilePath,"logfile.txt");
262     FILE* logFile = fopen(logFilePath, "w");
263
```

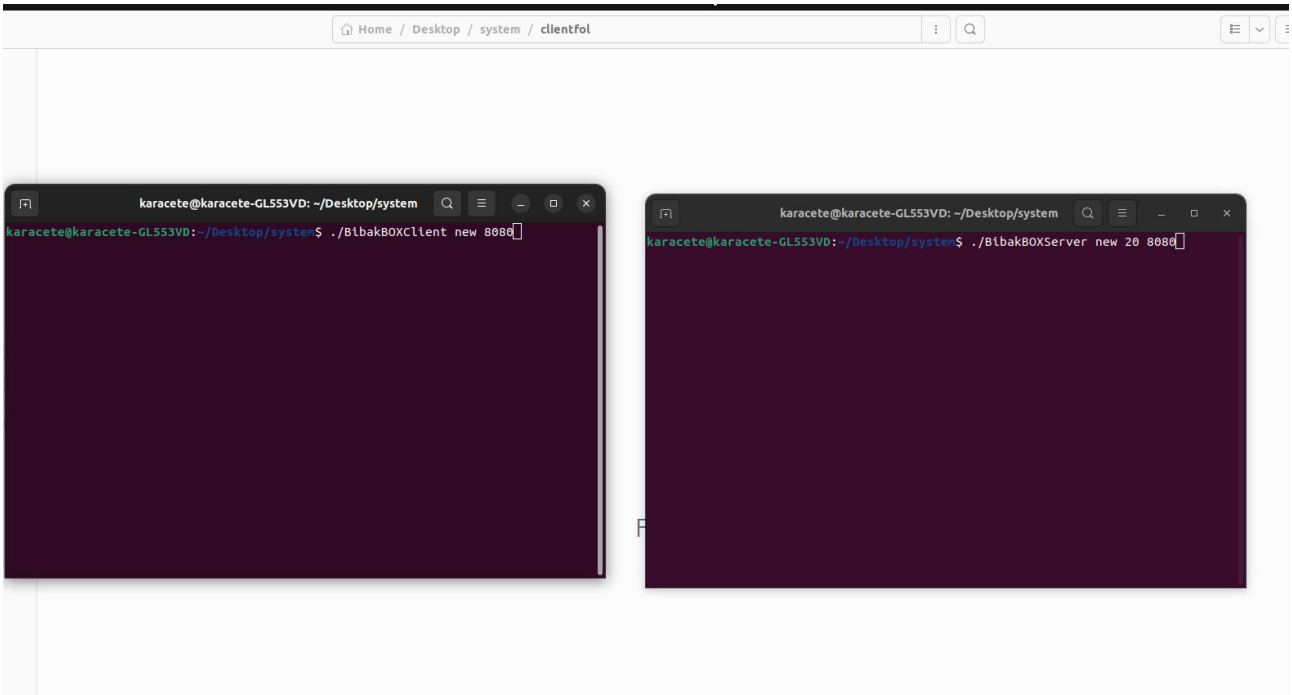


Firstly client file is empty.

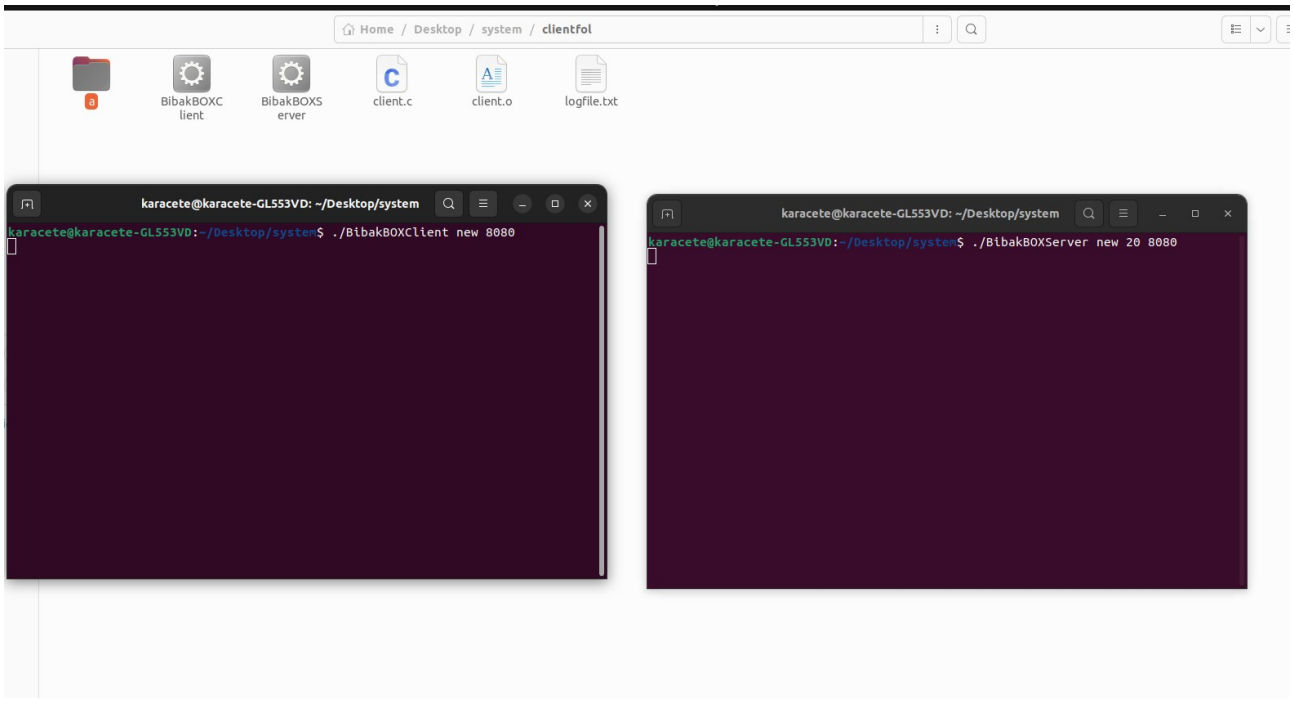
Server file’s entries are them.



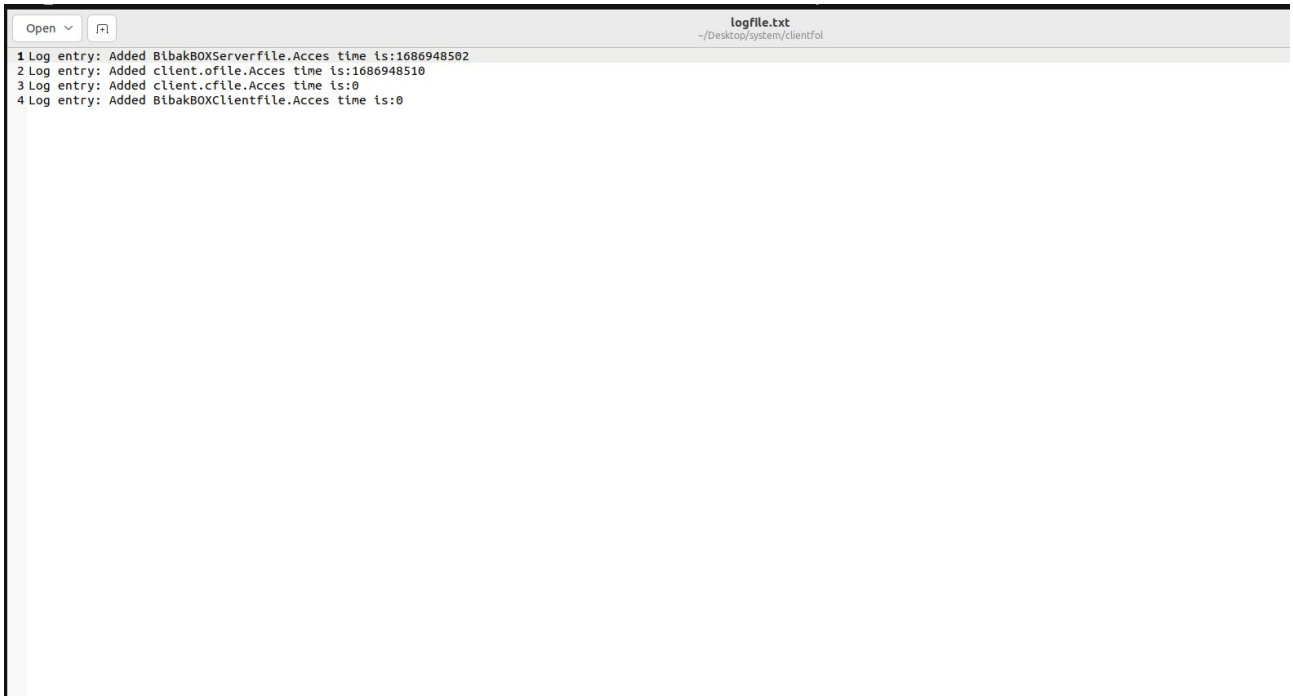
Runnig code.



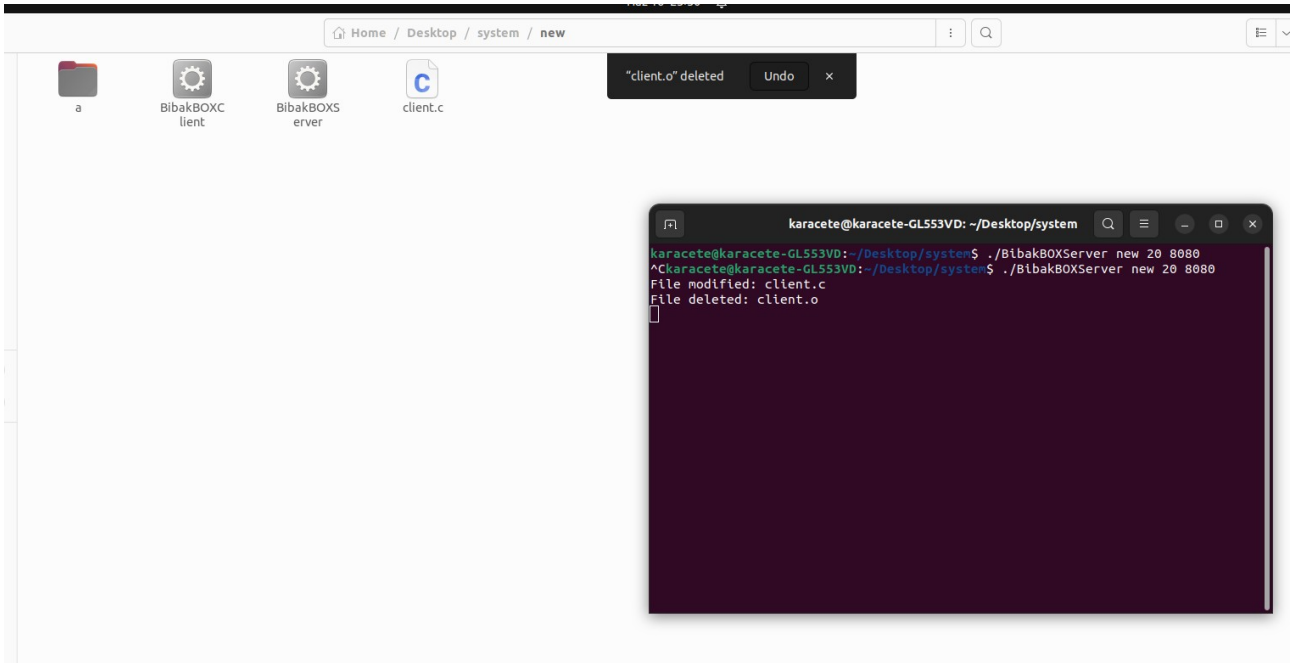
After run code



LogFile



For changes:


Home / Desktop / system / new
a
BibakBOXClient
BibakBOXServer
client.c
"client.o" deleted
Undo
karacete@karacete-GL553VD: ~/Desktop/system
karacete@karacete-GL553VD:~/Desktop/system$ ./BibakBOXServer new 20 8080
^Ckaracete@karacete-GL553VD:~/Desktop/system$ ./BibakBOXServer new 20 8080
File modified: client.c
File deleted: client.o