Report of Assignment 2

For Option 2

The files

Cache folder, *Client* folder, *Server* folder and *SocketUtils* folder include the code implementing cache, client and server. Double click *CDN.sln* can open the project in Visual Studio.

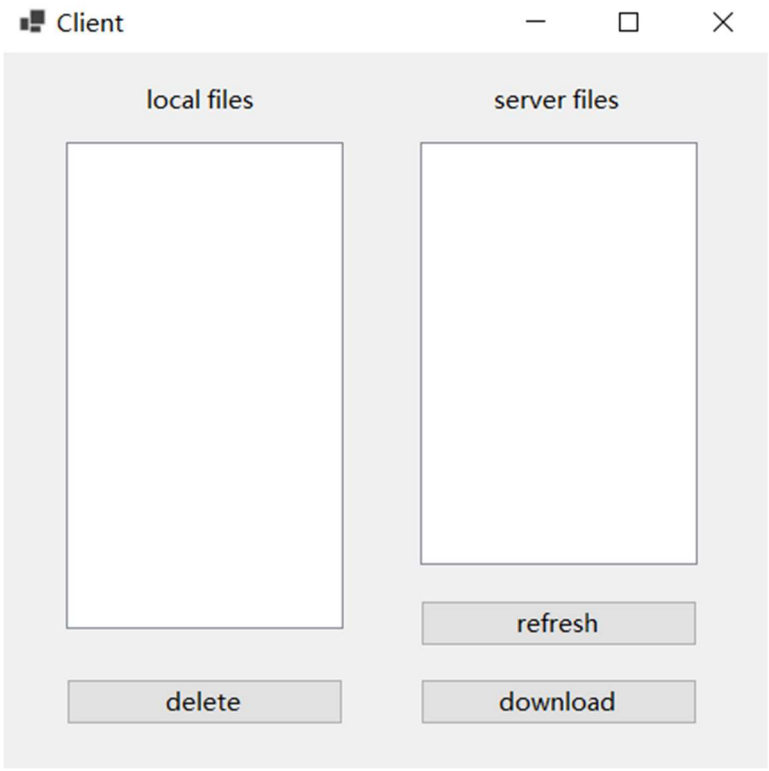
output includes the executable program and library file generated by the project.

How to run and how to use

Double-click *start*.bat in *output* folder and the desktop programs corresponding to the server, client and cache will run.

Cache will use port 11111 to receive and send data. Server will port 22222 to receive and send data.

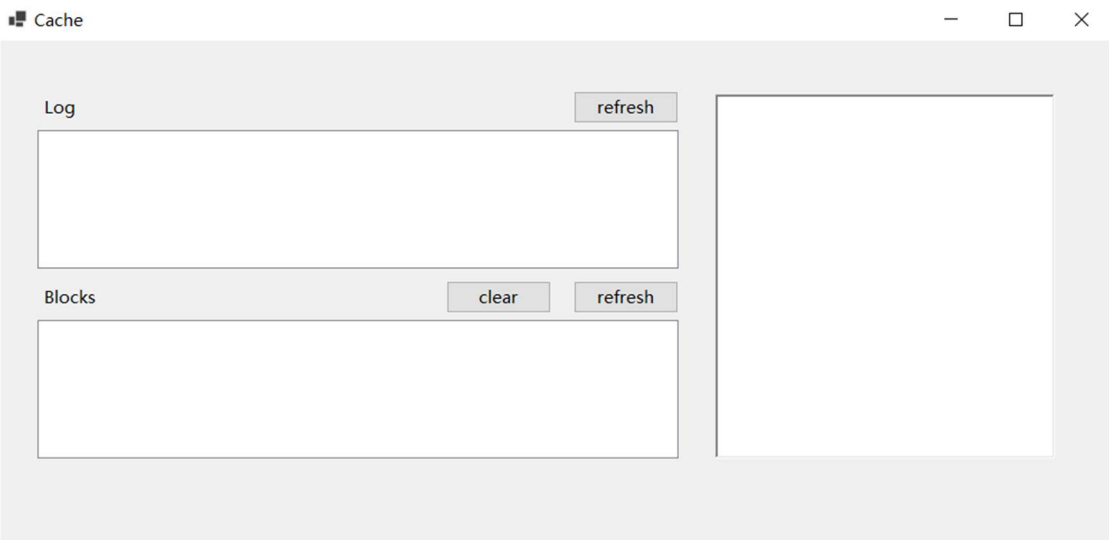
Client



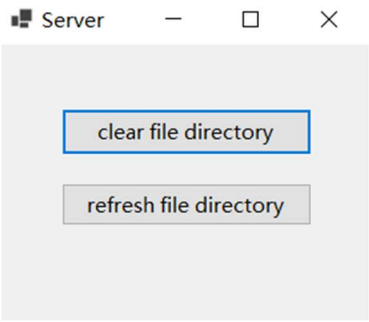
Click “refresh” button to get the files which can downloaded from server. Click “download” button to download the file chosen in “server files” box. The file will be put in */output/client/Files* The downloaded file will show in “local files” box. Click “delete” to delete the downloaded file

Cache

Click “refresh” button above the “Log” box will refresh it. New log will show in the box. Click “refresh” above “Blocks” box will refresh it. MD5 string of blocks will show in the box. Click the MD5 string, and the content of the block will show in the right box. “clear” button will clear all the blocks cached in */output/cache/cachedFiles*.



Server



Click “clear file directory” will clear the blocks in */output/server/blocks*. Click “refresh file directory” will generate blocks from the files in */output/server/files*. If the contents of this folder will change, I strongly recommend clicking “clear file directory” firstly and then change the files in the folder. And last, click “refresh file directory”.

The process of caching

The server will divide the file in *file* folder into several blocks in advance and store them in *blocks* folder. When cache want to download file from the server, cache will send the name of file. Server will send an array of md5 strings of blocks. Cache will check the md5 strings of cached blocks and send the md5 strings of uncached blocks back to server. And the server will send blocks according to the md5 strings received. Cache will store received blocks and send all the blocks back to client

Summary

Compared to option 1, option 2 can save amount of bandwidth. When cache download files from server, it will just download a part of the file. This will save time of transmitting and bandwidth. I can clearly feel that the first file is downloaded faster than the second file. But in option 2, the server will spend more time and computer resources to split the files. If considering the total time, it's hard to say which option is faster. So I divide the file before a user send a request