Introduction to Computer Networks

Lab1 (Deadline: will be announced on ilms)

1. Description

Implement a simple billboard. The server should create a billboard server and allow users (clients) to read and upload articles. The client should be able to access the billboard, read articles, and upload articles. Please use TCP socket and the programs should be implemented in C/C++.

2. Requirements

I. Function:

1) Before connection

The server side has several text files. These files include all articles and a list of the filenames of the articles.

2) Initialization

The server and client will first establish a TCP connection. Then, the server will send the list of the filenames the articles to the client. After that, the server will wait for commands or messages from the client.

3) User chooses what to do on the client side

User has two options. One of them is to read article. The other is to upload an article. The server will react base on what it receives.

For reading article, the client displays the current list of the filenames of the articles on the screen. When the user chooses an article, the client will send the identical number of the article to the server. Then, the server will find the corresponding text file and send its content to the client. The client receives the content and display the content on the screen.

For uploading article, the client reads a text file and sends the name of the file name and content of the article to the server. The server receives the article, saves it as a new article, update the current list of the filenames of all articles, and sends the current list to the client. The client displays the updated list of articles.

II. Note:

1) For simplicity, we assume that there is only one client at a time.
2) All data must be transmitted via TCP socket.

3) Use fixed port numbers on both the client side and server side.
4) Whenever the server receives a new article. It will update the list of the filenames of all articles and send it to the client.
5) All articles are saved as text files on the server side.
6) You need to handle illegal-command exceptions. For example, a client requests for an article that is not in the list.
7) The server needs to save all articles on the disk if it is shut down.
8) Use “winsock2.h”, ”socket.h” or any other socket library you can find.
9) Please do not copy other’s programs.

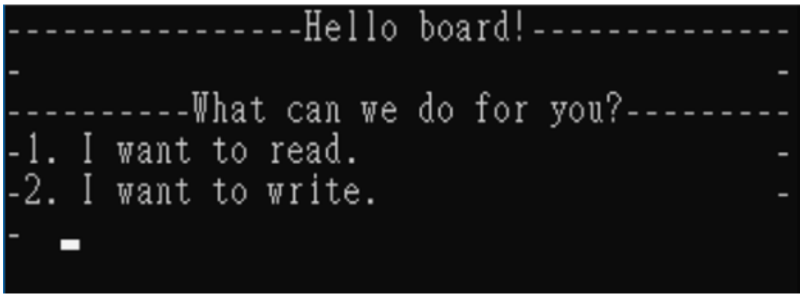
3. Evaluation:

I. (80%) Programs
II. (20%) A written report. File name of the report: studentID_report.pdf. Your report should include:
   1) Details of your implementation, including server-side and client-side.
   2) Step-by-step screenshots and explanations of the execution of each function.
   3) Descriptions of difficulties you encountered and your solutions.
III. (0%) Readme file
   1) File name for readme: studentID_readme.txt
   2) Please write a readme file to show how to execute your programs.

4. Sample screenshots

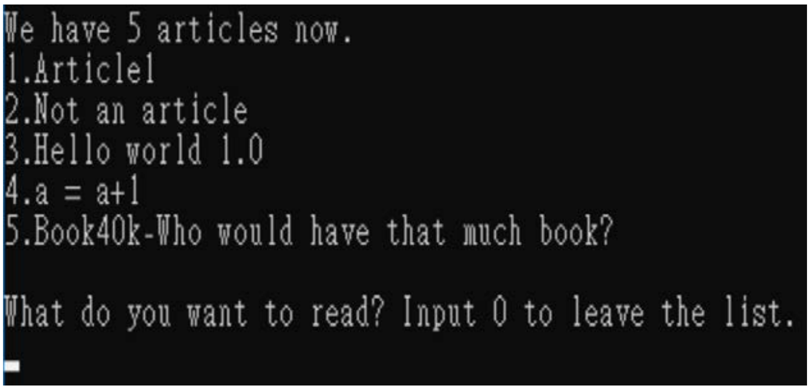
You don’t need to follow the sample exactly.

Menu



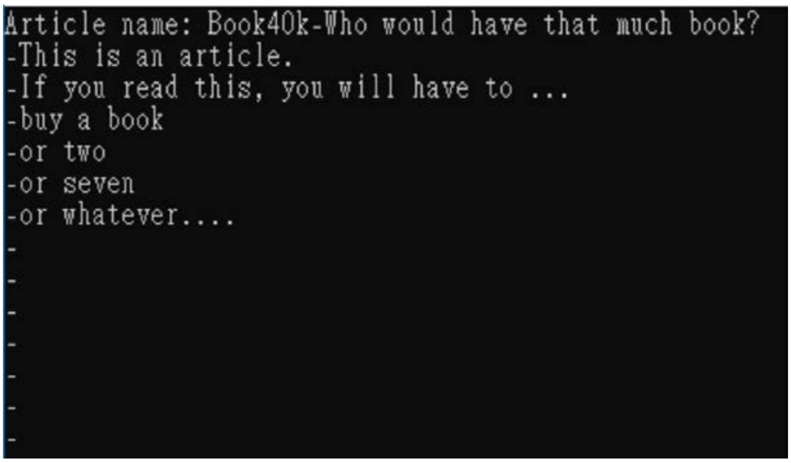
```
---------------Hello board!-------------
-                                      -
---------What can we do for you?--------
-1. I want to read.                    -
-2. I want to write.                   -
-
```

Read



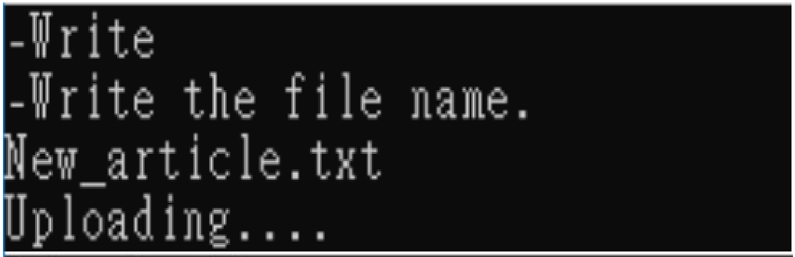
```
We have 5 articles now.
1.Article1
2.Not an article
3.Hello world 1.0
4.a = a+1
5.Book40k-Who would have that much book?

What do you want to read? Input 0 to leave the list.
```

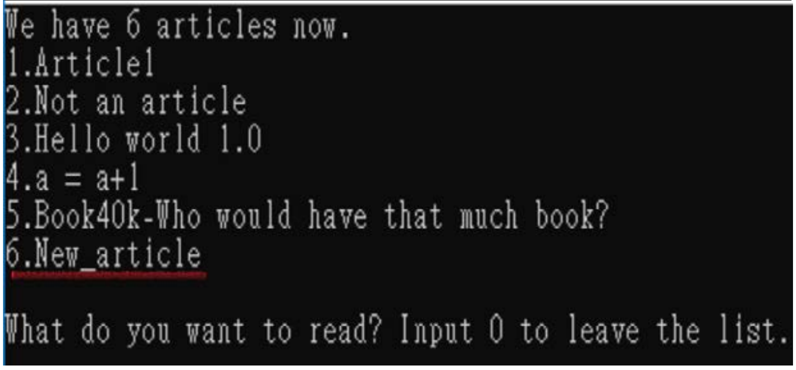


```
Article name: Book40k-Who would have that much book?
-This is an article.
-If you read this, you will have to ...
-buy a book
-or two
-or seven
-or whatever....
-
-
-
-
-
-
-
```

Upload article



```
-Write
-Write the file name.
New_article.txt
Uploading....
```



```
We have 6 articles now.
1.Article1
2.Not an article
3.Hello world 1.0
4.a = a+1
5.Book40k-Who would have that much book?
6.New_article

What do you want to read? Input 0 to leave the list.
```

5. Submission
   I. Please upload the following files to iLMS.
      1) studentID_ser.c
      2) studentID_cli.c
      3) studentID_report.pdf
      4) studentID_readme.txt
   II. All the files should be included, otherwise no grade will be given for the programming lab.
   III. Deadline: will be announced on iLMS.