System Programming Final Exam, Summer 2020

Modify your chatting server so that it asks the client login ID and password. The server should have login.txt file which contains the login ID and password for all registered users and continue the protocol only for the registered users. For an unregistered user, the server should ask whether the user wants to register and if yes the server should register the user before proceeding with the chatting protocol. Your answer should provide the **full source code** capture, output capture, and explanation about the code.

state2에서 이름 (id)를 받으면,

login.txt 파일을 열어서 그 안의 정보를 “ ” 기준으로 toknize 한 후,

tok 된 x 배열과 이름을 비교하여 만약

같은게 있으면 chk flag=1로, 바로 state 3으로 넘어가 chat가능하게 하고

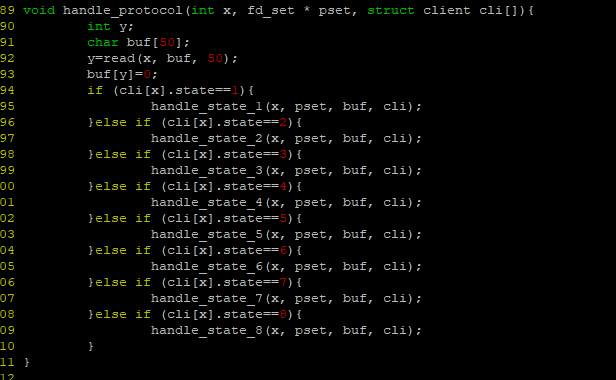
같은게 없으면 chk flag=0으로, state6으로 넘어가

yes 답이오면 txt 파일에 이름과, 비밀번호를 txt파일에 “ ”로 띄어쓰기 하여 저장합니다.

그 후, state 3으로 가서 chat을 하게 합니다.

 참고 tok 코드

server 코드입니다.



원 과제 4와 달리, state 2를 수정해주었습니다.

state 6,7,8을 추가해 주었습니다.

satae2:



fopen을 통해 login txt파일을 읽기전용으로 엽니다.

이미 아이디와 비밀번호가 “ ”로 나눠져 입력되어있기 때문에

toknize를 이용해 “ ”기준으로 z[]배열에 나눠 입력합니다.

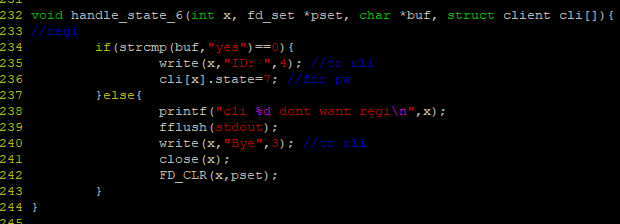
그후, cli로부터 입력받은 name(ID)값 buf와 z[]의 값을 비교합니다

만약 같은 이름 값이 없다면 chk flag 값은 그대로 0 일것입니다.

이경우에는 want regi? 를 cli에 보내 state6으로 갑니다. (regi 할지 안할지 결정하는 곳)

만약 같은 것이 있다면 이미 회원인 경우이므로,

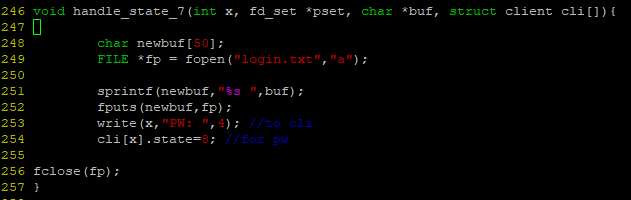
state3 - chat으로 바로 넘어가 채팅을 시작할 수 있도록 합니다.



6: yes( 회원가입 한다) 는 trigger가 오면

state 7로 이동합니다.

yes가 아닌 값이 오게되면, 종료합니다.



state7 에서는 login.txt 파일을 쓰기용으로 열어

받은 정보를 입력 합니다.

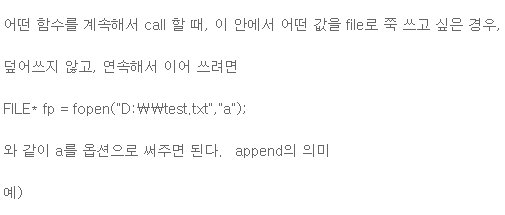
이때, sprintf를 통해 받은 buf값에 “ ”를 추가해줍니다. (나중에 toknize 하기위해)

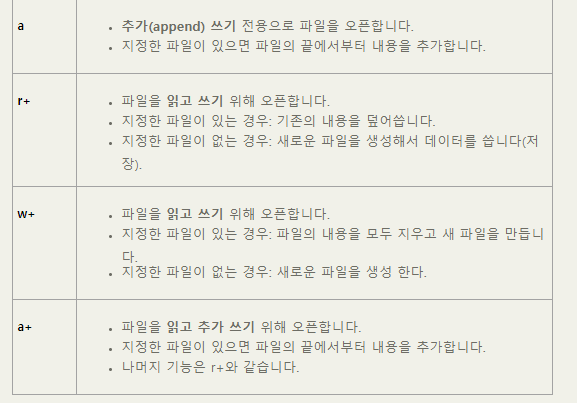
파일에 입력하고,

state 8로 갑니다 (PW받기위해)

fclose해줍니다.

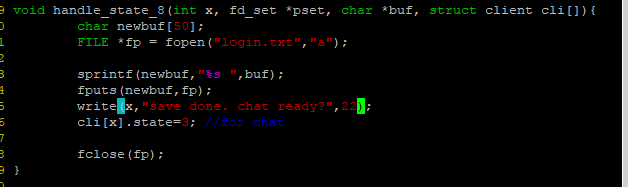
(txt파일에 정보를 입력할떄 그 위에 정보가 게쏙해서 겹쳐지지 않도록, 어디까지 차있는지 확인해봐야 할 것같습니다.)





이러한 방법이있습니다.

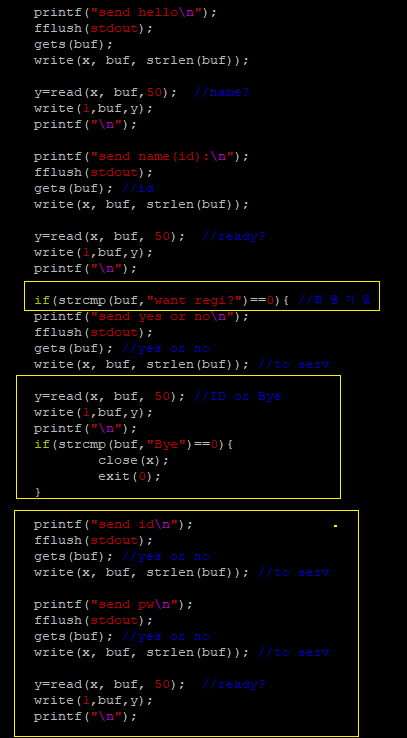
옵션을 a로 하여 txt파일에 이어서 쓸수 있도록 해주었습니다.



앞서 ID떄와 같이 파일에 정보를 save done을 한 후, chat ready? 를 출력해줍니다.

state3으로 가줍니다.

client>

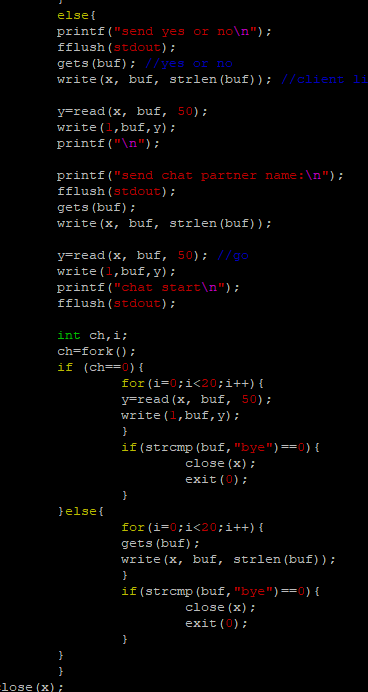


client 코드입니다.

거의 동일하지만, reigister를 하게되는 경우와 아닌경우를

wantregi? 가 오느냐 안오느냐에 따라 구별하여,

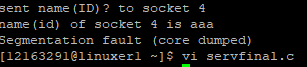
registerㄱ 필요한 경우는 원 chat코드에서 위의 박스친 네모 부분들의 정보를 입력하는 부분을 추가해주었습니다 (id, pw정보입력)



regi?를 만나지 않고 바로chat을 하게되는 else{}부분은

전의 과제 코드와 동일 합니다.

실행결과.



실행하였는데, id를 입력하는 부분에서 core dump가 발생했습니다.



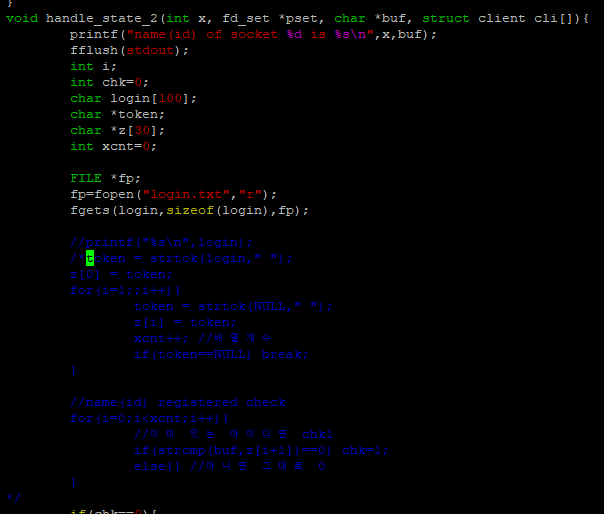
txt파일이 없는데 읽어서 일어난 일 같아 r+로 옵션을 수정해 주었습ㄴ다.

또 같은 에러가발행하여,

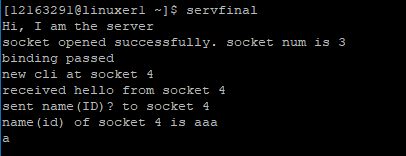
login.txt.파일을 만들어주었습니다.



분리 정의



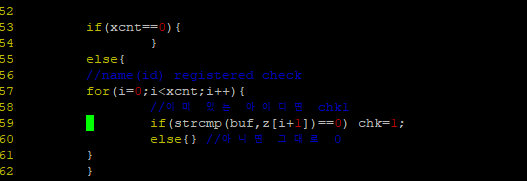
자꾸 같은 에러가 떠, file을 잘 읽어오나 순서대로 확인을 해주었습니다.



미리 생성해둔, login.txt 파일은 잘 읽어오 고 있습니다.

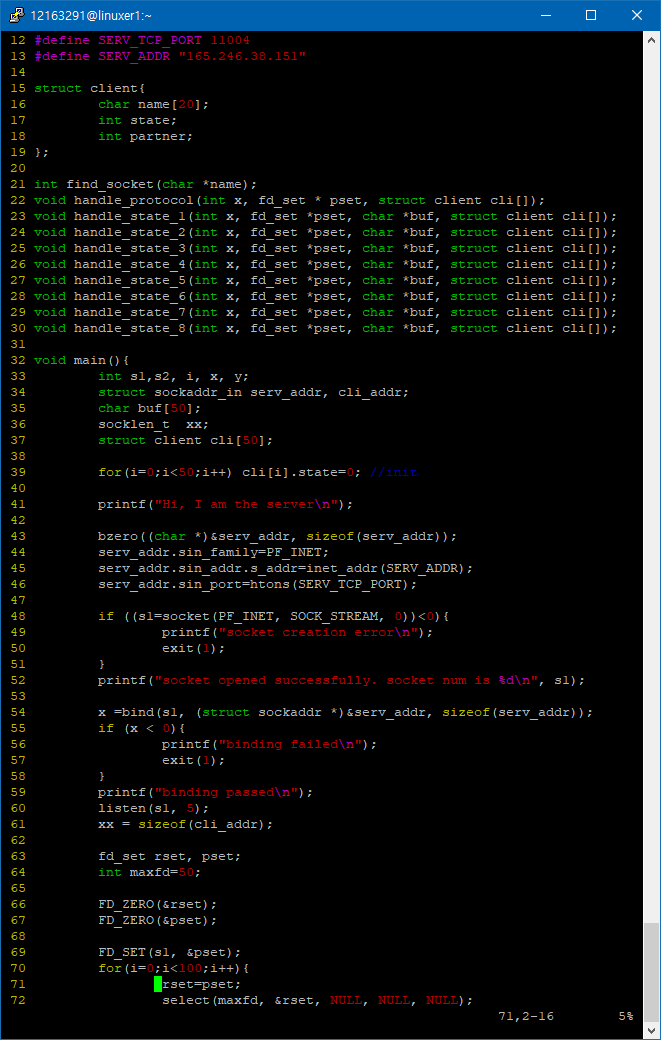
toknize부분에서 오류가 생긴 것 같습니다.

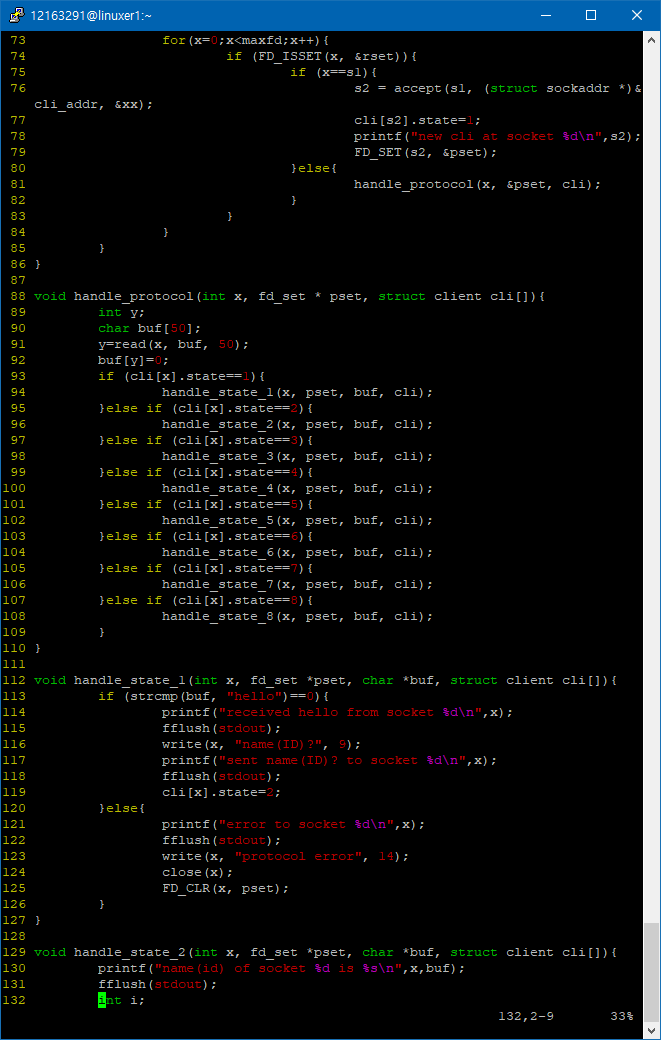
이를 보완하면 해낼 수 있을 것 같습니다.

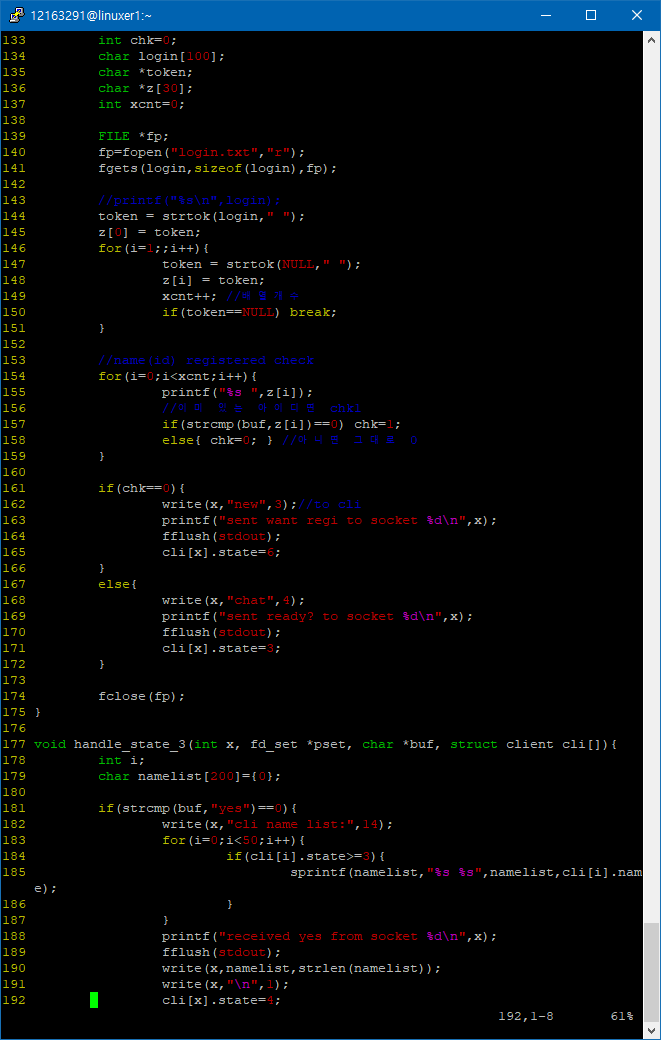


tok 로 분리된 단어 개수가 0일때와 아닐떄로 구분해 주었습니다.

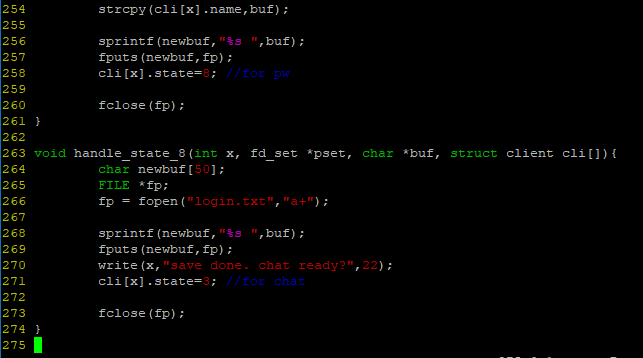
1. full serv code



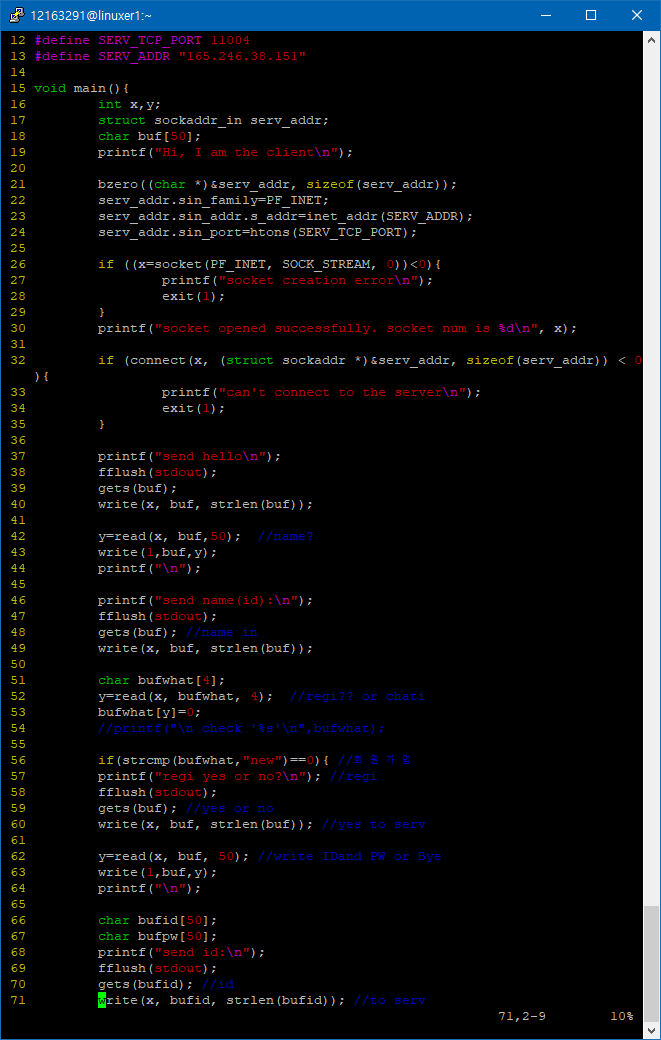


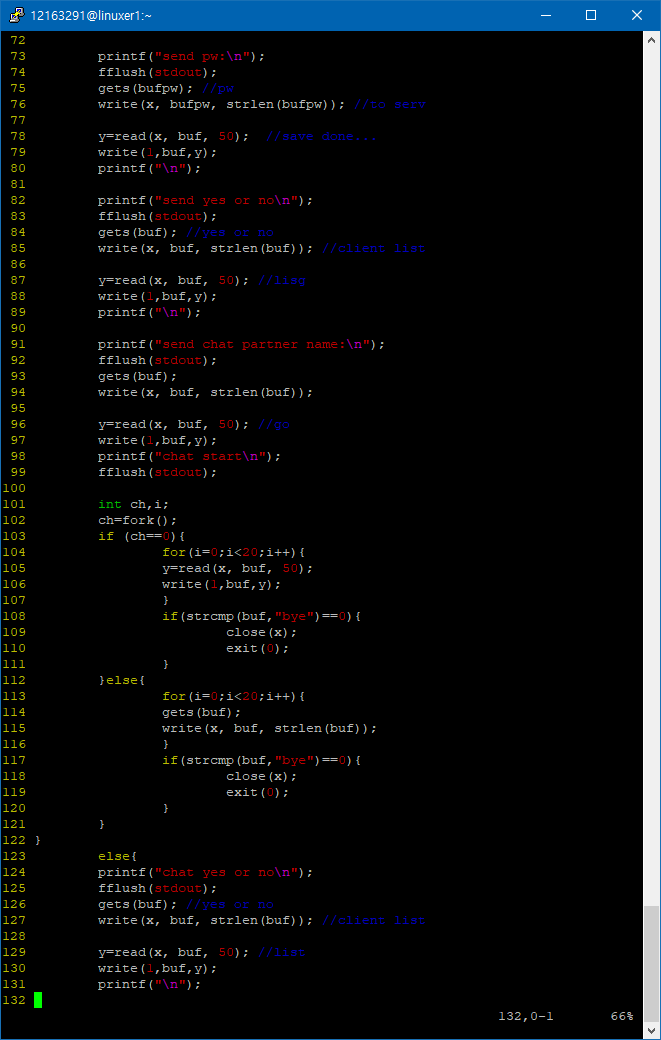




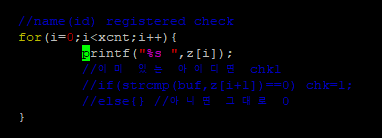


2. full client code





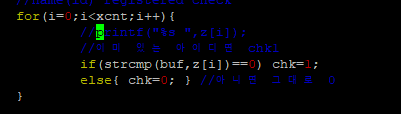




state 2부분에서 자꾸 dumped되어, 어디에 문제가 있나 차근차근 확인해 보앗습니다.



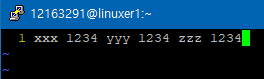
파일의 tokenize는 잘 되고있습니다.



core dump는 위 server 코드의 부분에서(회원가입 아이디 확인 부분) 전에 사용했던 z[i+1] 에서의 ‘+1’때문에 발생했음을 깨닫고, 이를 수정해 주었습니다.

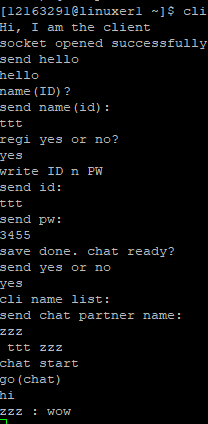
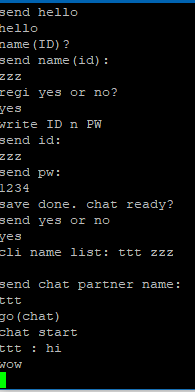
그 후, dump가 발생하지 않았습니다.

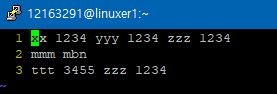
+



login.txt 파일을 미리 만들어 두었습니다.

, f open (a)를 통한 입력은 잘 이루어 지고 있는 것을 확인할 수 있엇습니다.

 cli ttt cli zzz



login.txt

**회원가입 파일안에 새로운 아이디와(ttt, zzz) 비밀번호 값(3455 1234)이 저장되고,**

chat을 진행 할 수 있는 것을 확인할 수 있었습니다.

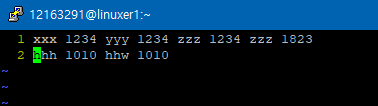
이 값들과 입력된 id(buf)값을 비교하여 회원가입 유무를 알 수 있습니다.

현재 보완해야 하는 부분은

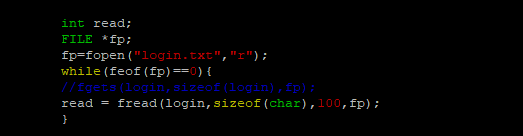
**fgets에서 생기는 줄바꿈 이전까지만 buf로 가져오는 문제를 해결해 주어야합니다.**

**(현재 login.txt의 마지막 줄 정보만 가져오고 있습니다.)**

->

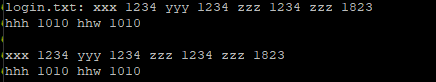


**login.txt**

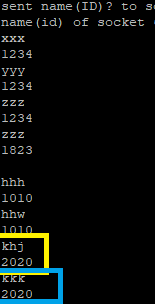
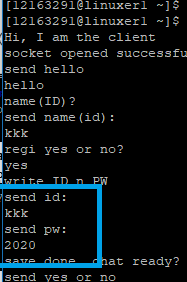
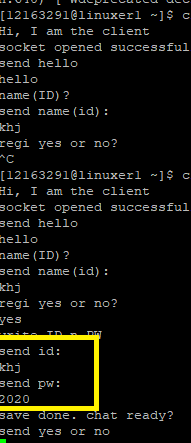


**serve 코드에서 fgets -> fread로 변경하여 읽어주었더니 + feof() 를통해**

**파일을 끝까지 읽는 것을 확인할 수 있었습니다.**

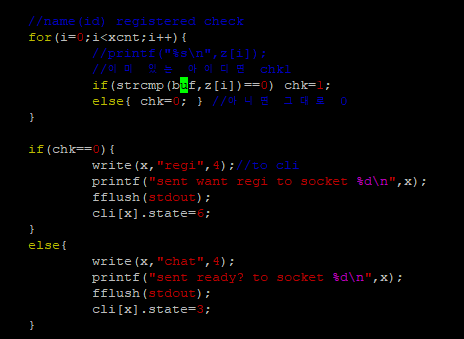


**이후 tken도 “ ” 기준으로 잘 되는지 확인해보았습니다 (z[i] 마다 \n 넣어줘 출력함으로**

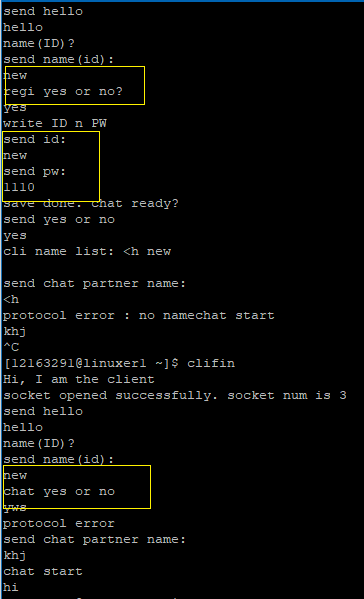
****

**cli cli serv**

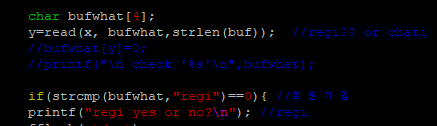
**register부분은 완벽해 졌습니다.**



**serv 의 회원이냐 아니냐 판단 부분입니다.**

****

**cli에서 실행 결과입니다. new가 처음에 들어왔을 때에는 regi를 할것이냐 물어보고, 아닐 떄에는 chat을 할 것이냐 물어보는 부분에서 회원 유무가 구분이 되고 있음을 할 수 있습니다.**



**..**



**cli 에서 regi와 chat중 무엇을 해야하는지는 cli의 위 부분을 통해 구현하였습니다.**