**시스템 프로그래밍**

**13주차 과제**

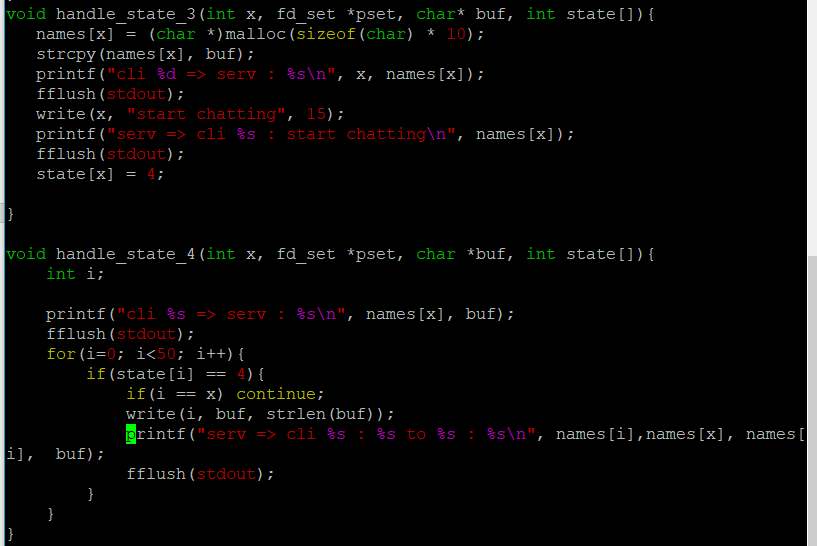
**12181759**

**김태완**

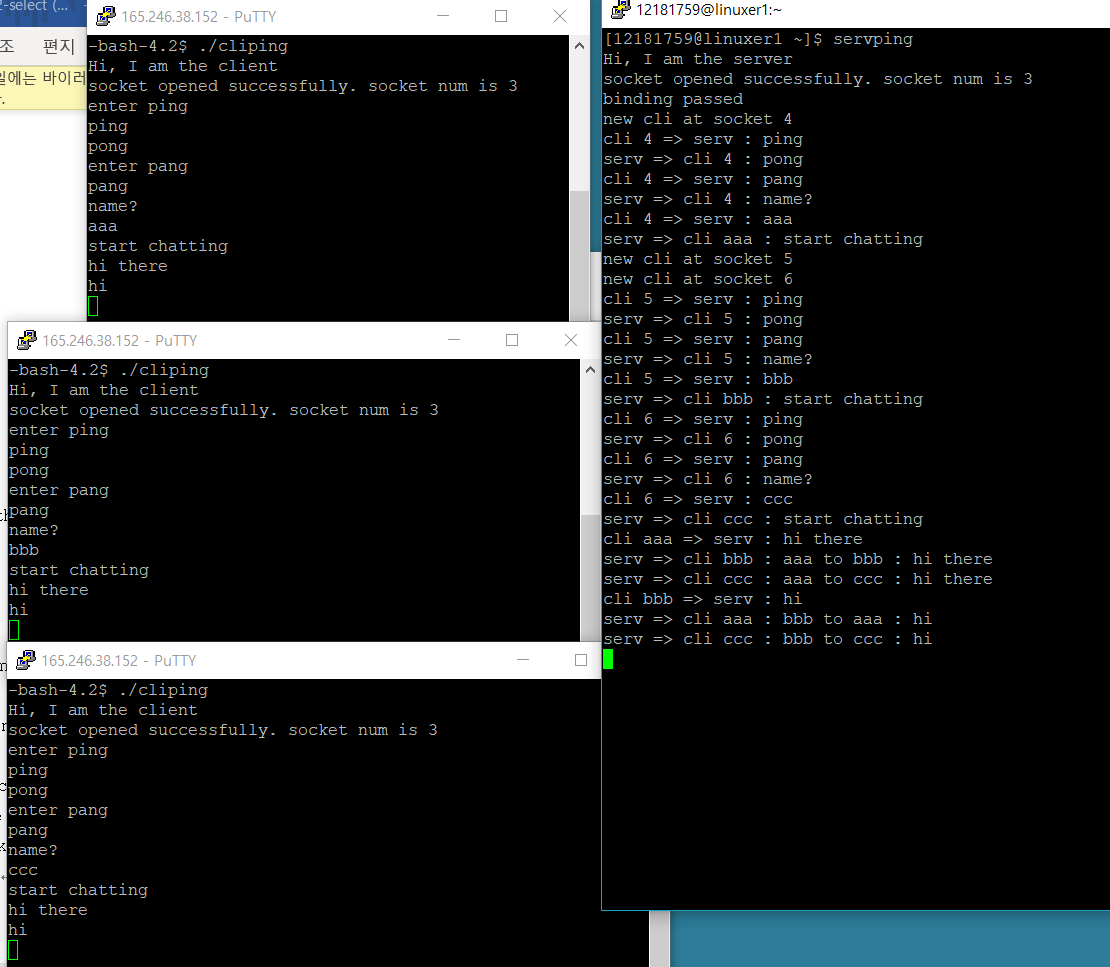
5)



수정한 cliping.c 코드이다. pang 다음으로 돌아올 pung 대신 name? 이라는 string을 입력받고 내 이름을 입력해 주어야하기 때문에 이 부분을 추가해 주었다.

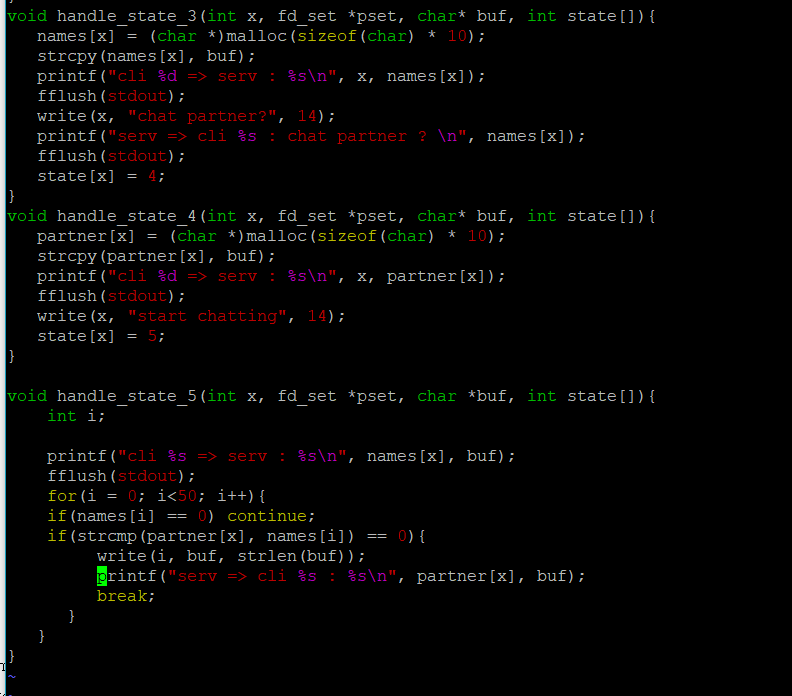


수정한 servping.c 코드. 구현을 위해 먼저 문자형 포인터 배열 names 를 선언해 주었다. state 3, 4는 위와 같이 수정하였고 state2에서는 마지맞 pung 대신, name? 가 x 로 write 된다. 강의노트 예제 에서는 3명의 cli 가 모두 연결된 후, 동시에 3개의 start chatting 가 cli 로 보내졌지만, 몇 명의 cli가 들어올 지 모르고, 프로토콜이 끝나고 연결이 되는 즉시 보내지는게 맞다고 생각하여 그렇게 구현했다. x는 현 buf 를 입력한 socket의 번호이므로 이를 index로 해서 names 배열에 계속 접근하여 buf를 보낸 사용자의 이름을 사용했고, I 는 내가 사용하는, 즉 내가 정의한 socket의 번호이므로 이를 index로 다른 cli 들에게 접근했다.

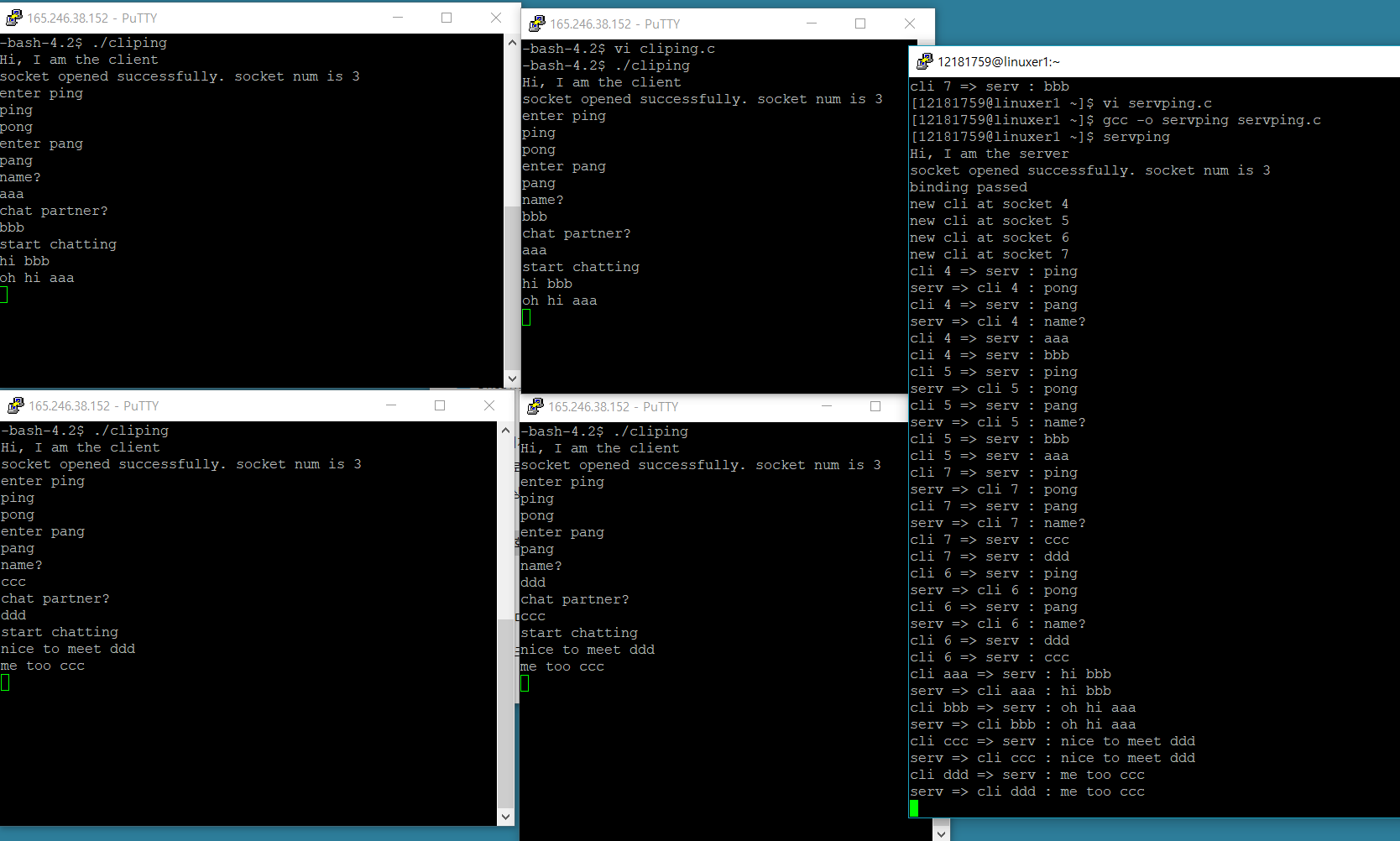


모든 기능이 정상적으로 수행됨을 볼 수 있었다.

6)

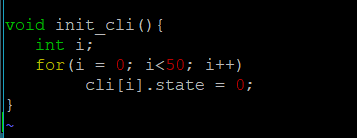
구현을 위해 수정한 servping.c 파일 이다. state 3, 4, 5를 다음과 같이 수정했다.

구현을 위해 문자형 포인터 배열 partner 를 선언했다. state3에서 추가된 점은 이름을 입력받고 난 후에, 그 사람의 partner를 입력받을 준비를 한다. 이후 state4에서 받은 파트너의 이름을 보낸 사람의 socket 번호를 index로 하는 배열에 저장한다. state5 에서는 그 이름을 찾아서 그 이름의 index를 socket 번호로 하는 secreen에 보낸 메시지를 출력한다.

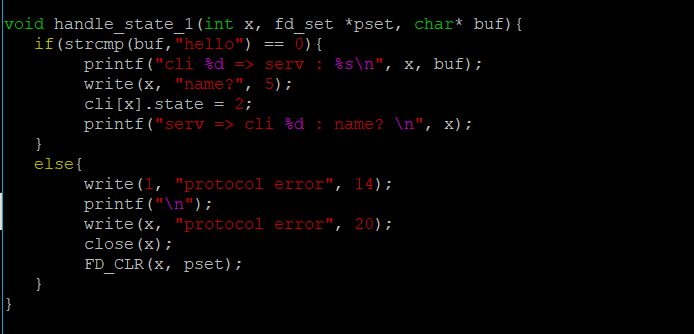


모든 기능이 정상적으로 구현됨을 확인 하였다.

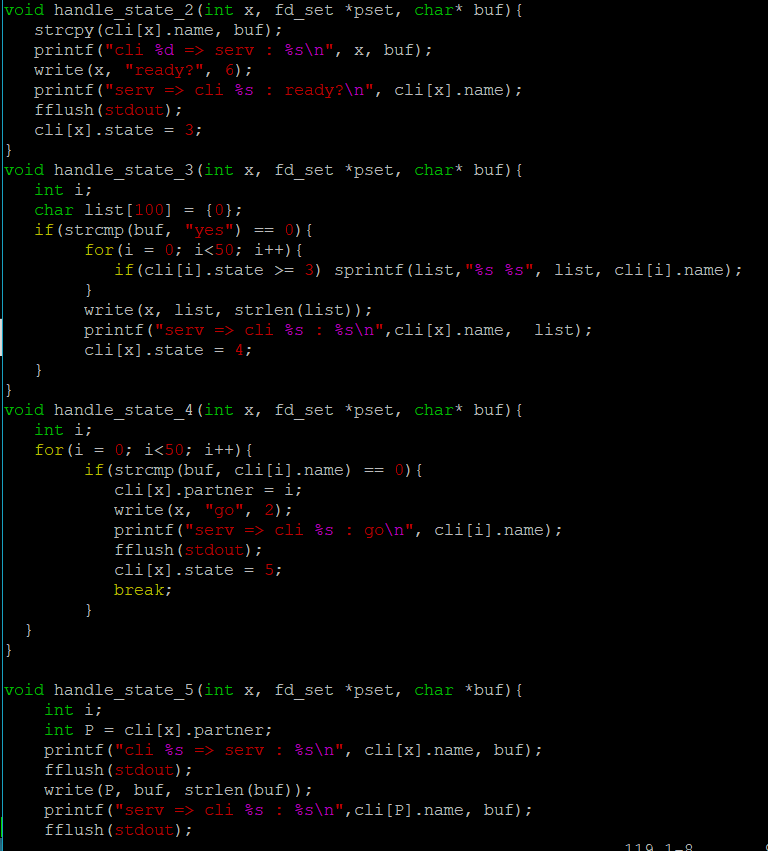
7)



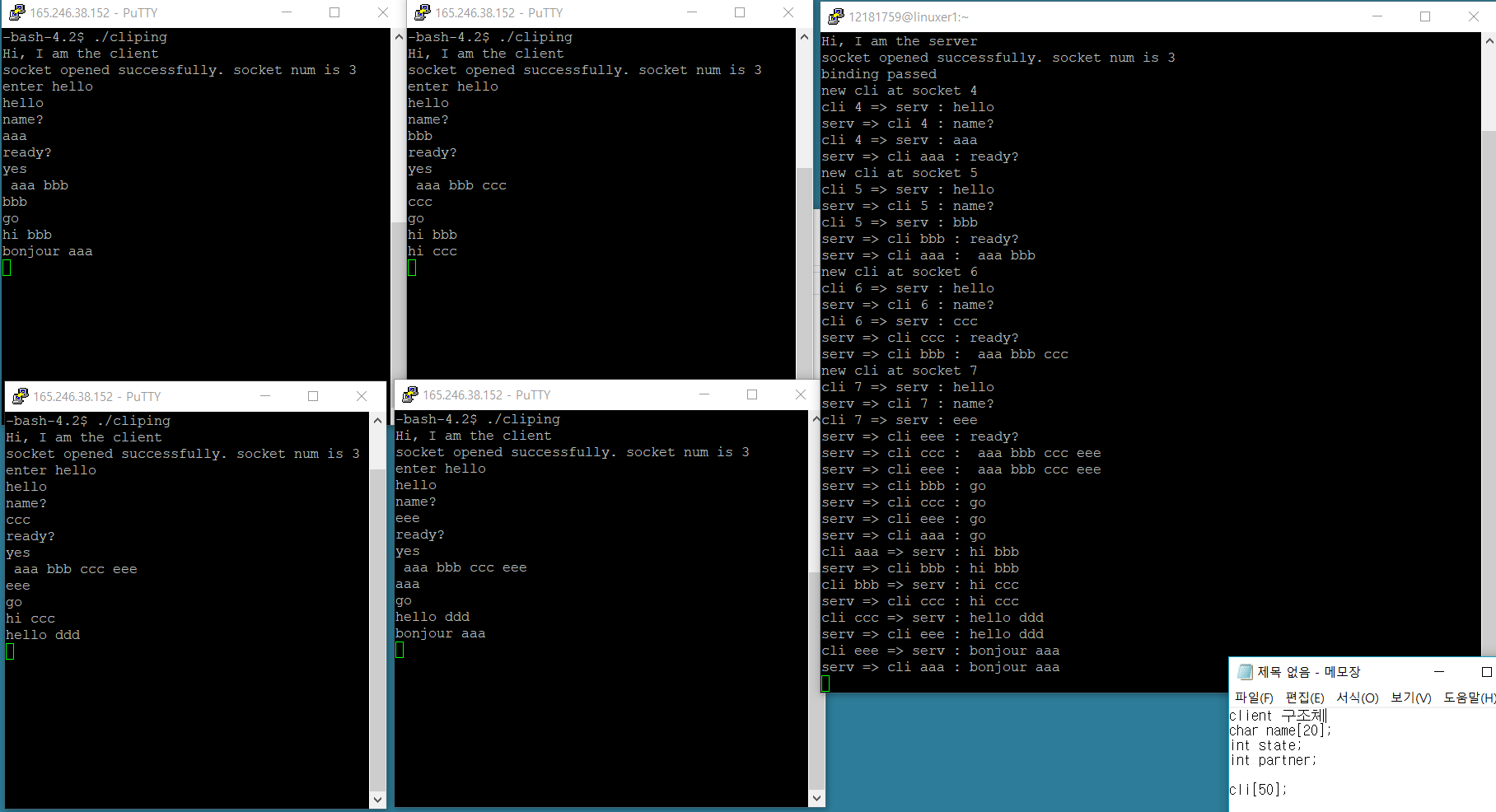
먼저, 비어있는 cli에 접근해서 값을 받으려 하니까 오류가 생겨서 state를 초기화 해주기 위한 init 함수를 새로 작성했습니다.



문제에서 요구한 대로 프로토콜을 수정했습니다. hello가 들어오면 name 을 묻고 state를 2로 수정.



state2에서는 이름을 받고 ready? 를 cli에게 물어봄, state3에서는 그 대답이 yes 라면, 현재 state가 3 이상은 cli 들의 이름을 sprintf로 모두 하나의 list 안에 묶어서 하나의 buf로 cli로 보내어줌, state4에서는 for문으로 cli 배열을 모두 돌며, 입력받은 buf와 동일한 이름이 있는 지 찾음,(입력받은 buf 는 list안에 있는 사람 중에 partner가 될 사람) 만약 찾으면 그 사람의 index를 cli의 partner 의 정수값으로 가져옴. 다음으로 state5에서는 cli 와 partner cli간에 계속 채팅이 이루어지게 구현하였습니다.



모든 기능이 정상적으로 구현됨을 확인 했습니다.