How to run

Step 1, create three instances, where one we will use as server and the other two as clients

Step 2. Connect to each of them and upload the files Client.c, Server.c, & DieWithError.c
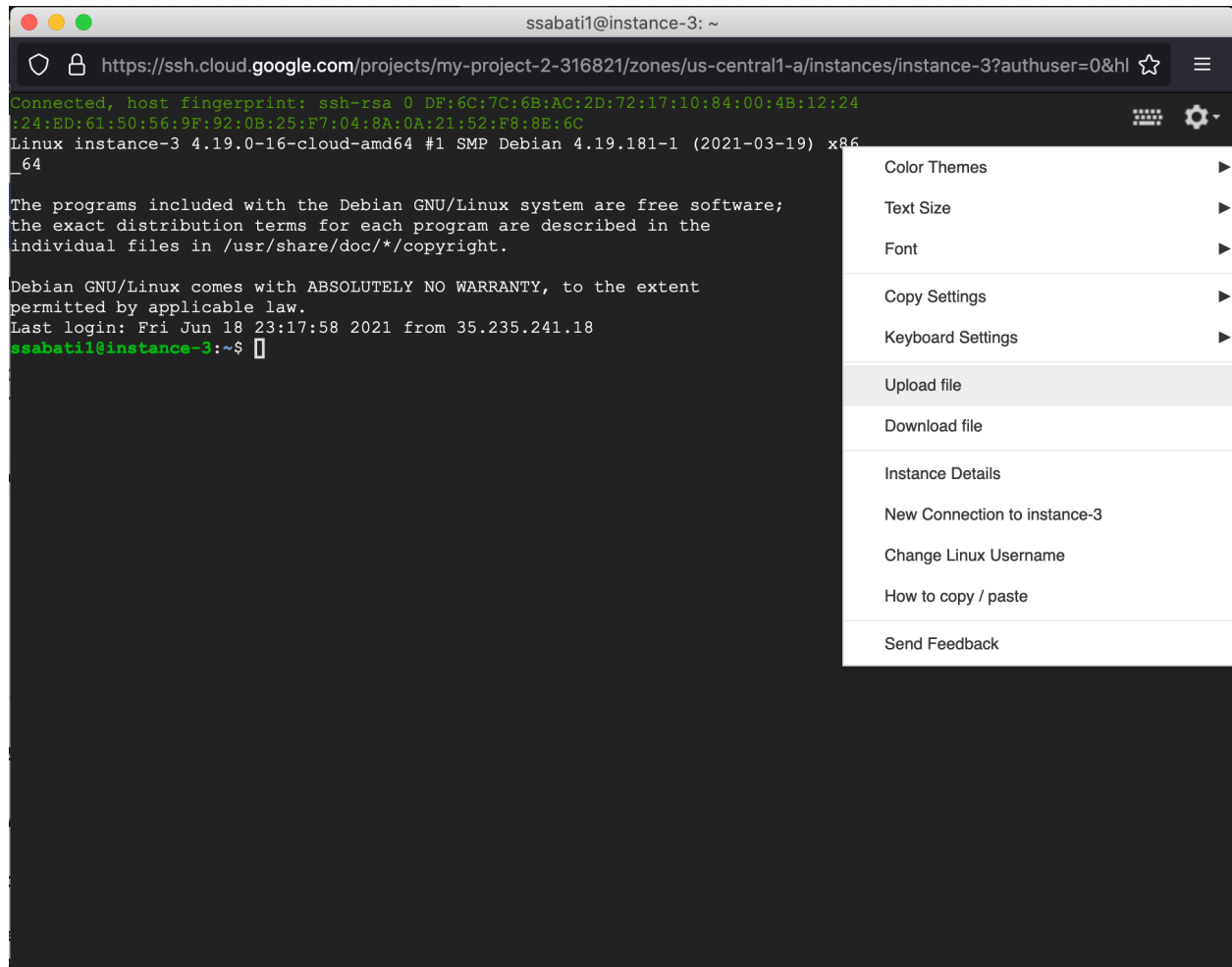
Step 3, once uploaded, Run in instance 1 command gcc -o server Server.c DieWithError.c, and command gcc -o client Client.c DieWithError.c in instance 2 and 3, as shown below

https://ssh.cloud.google.com/projects/my-project-2-316821/zones/us-central1-a/instances/instance-2?authuser=0&hl

```
Connected, host fingerprint: ssh-rsa 0 F2:E7:F4:D2:36:4E:E7:BE:6E:68:92:C7:4E:C6
:39:A0:9D:42:D8:D0:E0:80:52:61:D6:D9:D0:2D:BE:F0:81:97
Linux instance-2 4.19.0-16-cloud-amd64 #1 SMP Debian 4.19.181-1 (2021-03-19) x86
_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Fri Jun 18 23:17:58 2021 from 35.235.241.19
ssabati1@instance-2:~$ ls
Client.c  DieWithError.c  Server.c
ssabati1@instance-2:~$ gcc -o client Client.c DieWithError.c
ssabati1@instance-2:~$ ls
Client.c  DieWithError.c  Server.c  client
ssabati1@instance-2:~$
```
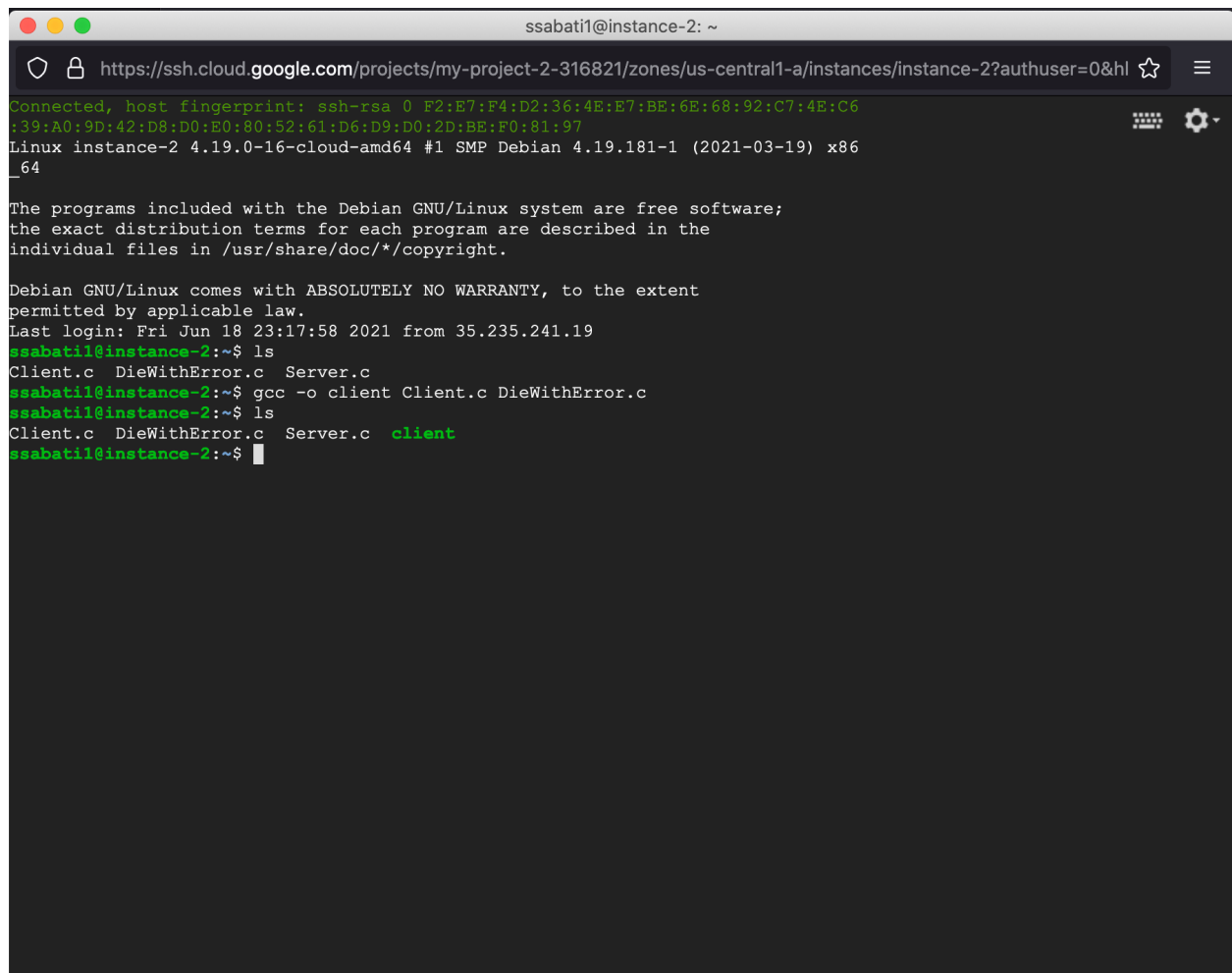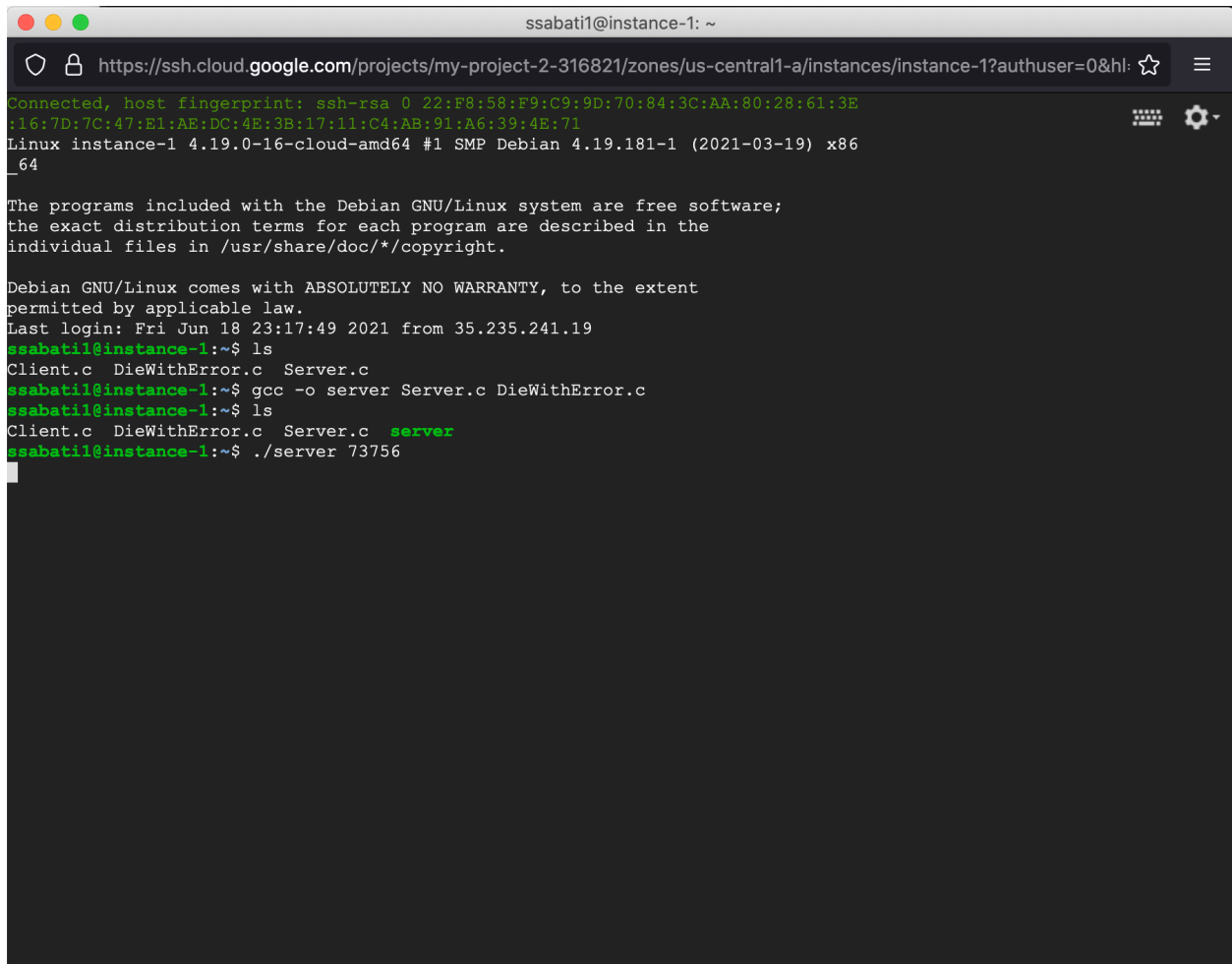
(instance 3 should look like last picture as well)

Step 4, connect to server in instance 1 by running command ./server [Port Number], example ./server 73756, and in instance 2 and 3 connect as clients running the following command ./client [IP address from instance 1] [Port number], example ./client 10.128.0.2 73756, as shown below

(instance 3 should look like last screenshot)

Step 5, Enter player ID, TCP connection Port number in the clients' instances, as shown below

```
Connected, host fingerprint: ssh-rsa 0 DF:6C:7C:6B:AC:2D:72:17:10:84:00:4B:12:24
:24:ED:61:50:56:9F:92:0B:25:F7:04:8A:0A:21:52:F8:8E:6C
Linux instance-3 4.19.0-16-cloud-amd64 #1 SMP Debian 4.19.181-1 (2021-03-19) x86
_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Fri Jun 18 23:17:58 2021 from 35.235.241.18
ssabati1@instance-3:~$ ls
Client.c  DieWithError.c  Server.c
ssabati1@instance-3:~$ gcc -o client Client.c DieWithError.c
ssabati1@instance-3:~$ ls
Client.c  DieWithError.c  Server.c  client
ssabati1@instance-3:~$ ./client 10.128.0.2 73756

Enter the playerID :- 5678

Enter the TCP connection port number  :- 25678

1 for who, 2 for lookup, 3 for logout :
```

Step 6, shows options 1 (for identifying other players that are logged in), option 2 (for looking player by ID), option 3 to log out. Choosing option 2 is shown below to for looking up the player 5678 that client 1234 wants to play with

Step 7, Gives option to send request to play by entering 1, or 2 if they don't want to send request. While client 1234 in instance 2 send request, client 5678 in instance 3 receives the request.


```
Connected, host fingerprint: ssh-rsa 0 F2:E7:F4:D2:36:4E:E7:BE:6E:68:92:C7:4E:C6
:39:A0:9D:42:D8:D0:E0:80:52:61:D6:D9:D0:2D:BE:F0:81:97
Linux instance-2 4.19.0-16-cloud-amd64 #1 SMP Debian 4.19.181-1 (2021-03-19) x86
_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Fri Jun 18 23:17:58 2021 from 35.235.241.19
ssabati1@instance-2:~$ ls
Client.c  DieWithError.c  Server.c
ssabati1@instance-2:~$ gcc -o client Client.c DieWithError.c
ssabati1@instance-2:~$ ls
Client.c  DieWithError.c  Server.c  client
ssabati1@instance-2:~$ ./client 10.128.0.2 73756

Enter the playerID :- 1234

Enter the TCP connection port number  :- 21234

1 for who, 2 for lookup, 3 for logout : 1

 logged in players are :-1234 5678

1 for who, 2 for lookup, 3 for logout : 2

 Enter the id of the player : 5678

 Player 5678 IP Adress is 10.128.0.4  and TCP port is 25678
 Enter 1 to send request, 2 to not send request :- 1
```
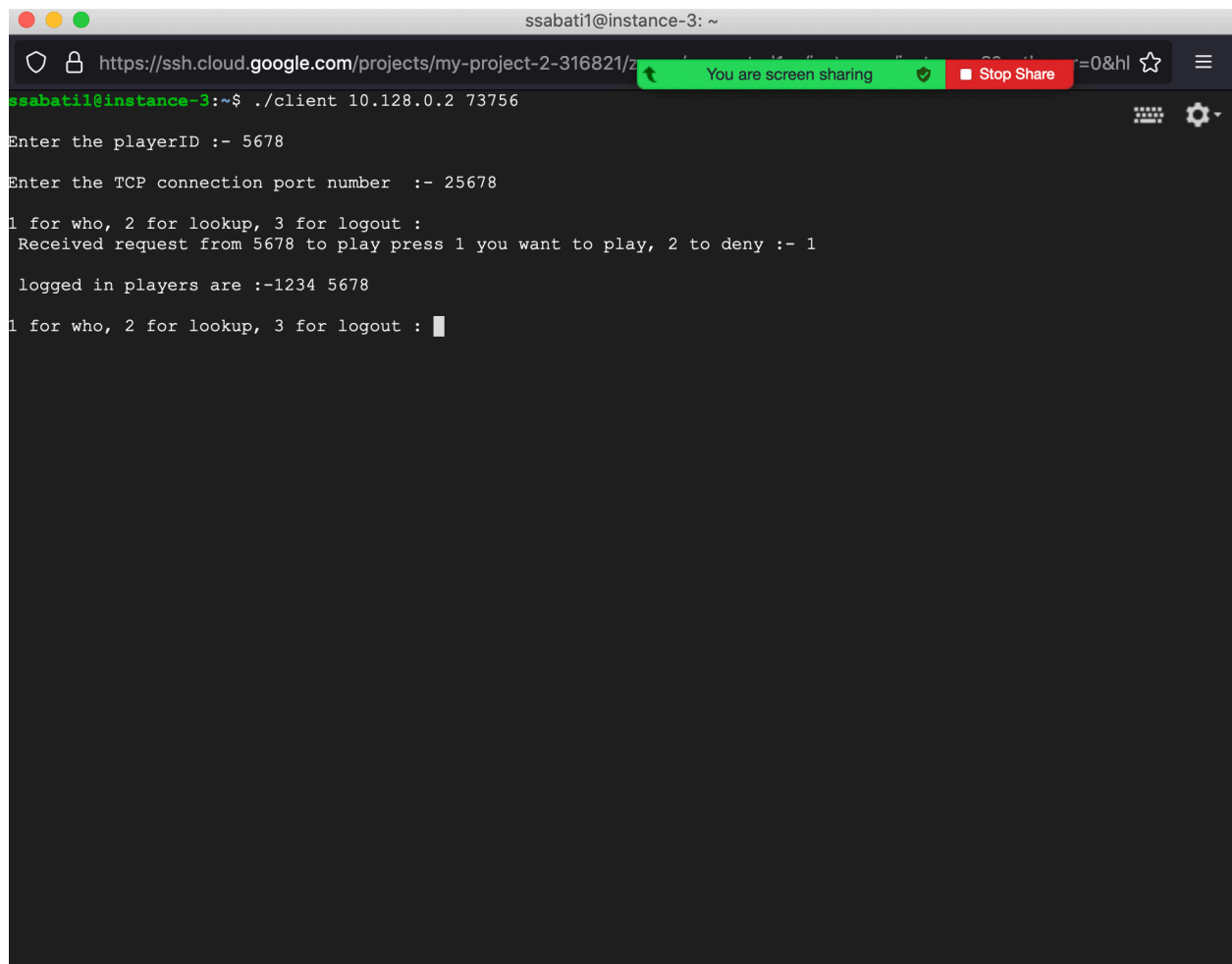
Step 8, Client 1234 was able to send request and 5678 was able to receive request

Error we are getting is when client 5678 receives the approval/deny request, the scanf does not work properly, it considers the scanner for the parent process and prints the "who list"
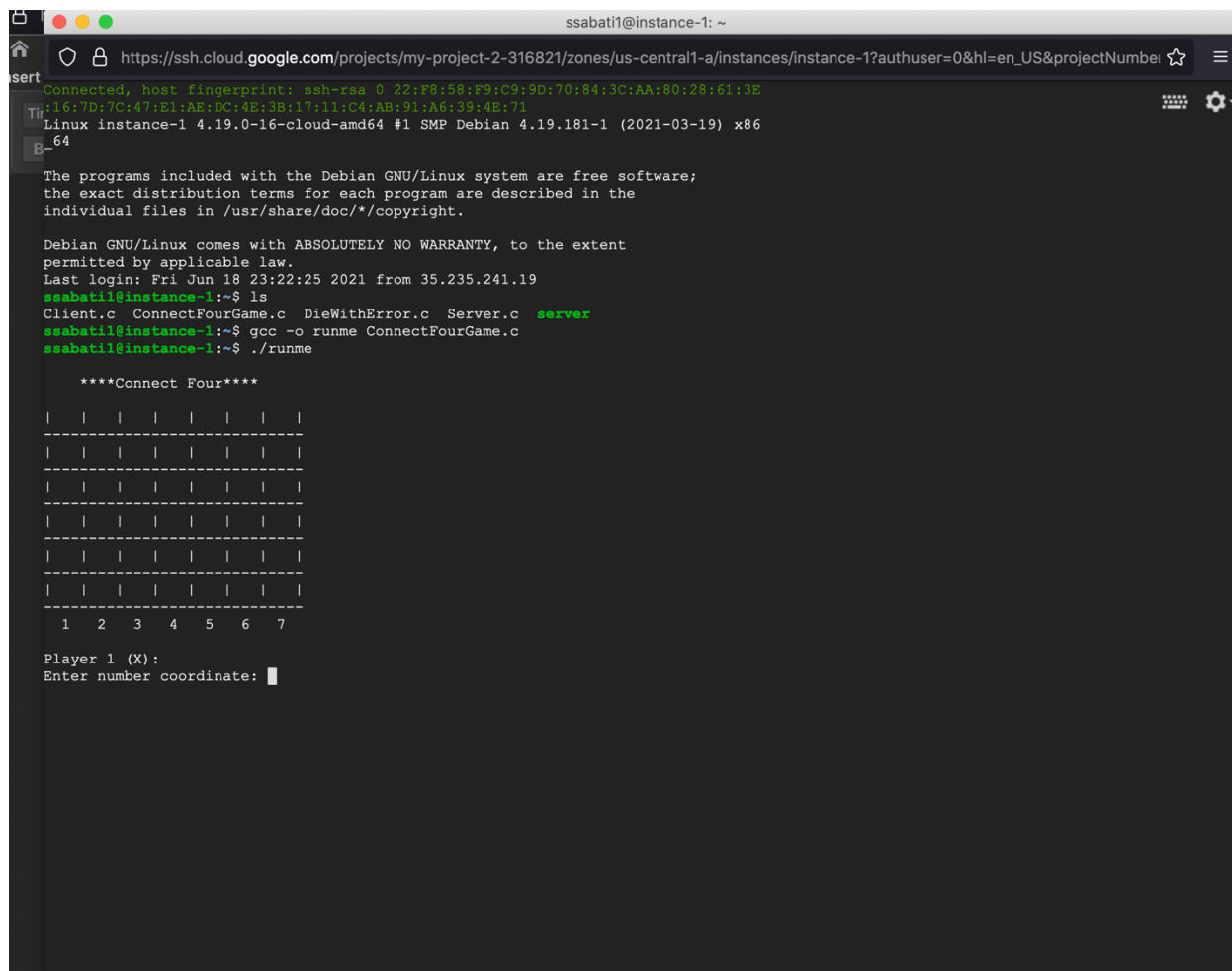
However the game on its on works, please look attached photos

More…

```
|   |   |   |   |   |   |   |
-------------------------
|   |   |   |   |   |   |   |
-------------------------
|   |   |   |   |   |   |   |
-------------------------
|   |   |   |   |   |   |   |
-------------------------
|   |   |   |   |   |   |   |
-------------------------
  1   2   3   4   5   6   7

Player 1 (X):
Enter number coordinate: 1

     ****Connect Four****

|   |   |   |   |   |   |   |
-------------------------
|   |   |   |   |   |   |   |
-------------------------
|   |   |   |   |   |   |   |
-------------------------
|   |   |   |   |   |   |   |
-------------------------
|   |   |   |   |   |   |   |
-------------------------
| X |   |   |   |   |   |   |
-------------------------
  1   2   3   4   5   6   7

Player 2 (O):
Enter number coordinate: 2

     ****Connect Four****

|   |   |   |   |   |   |   |
-------------------------
|   |   |   |   |   |   |   |
-------------------------
|   |   |   |   |   |   |   |
-------------------------
|   |   |   |   |   |   |   |
-------------------------
|   |   |   |   |   |   |   |
-------------------------
| X | O |   |   |   |   |   |
-------------------------
  1   2   3   4   5   6   7

Player 1 (X):
Enter number coordinate: ^[^[
```

And so on by the time x wins

```
|   |   |   |   |   |   |   |
-----------------------
|   |   |   |   |   |   |   |
-----------------------
|   |   | X |   |   |   |   |
-----------------------
| X | X | O | X |   |   |   |
-----------------------
| X | O | O | O |   |   |   |
-----------------------
  1   2   3   4   5   6   7

Player 2 (O):
Enter number coordinate: 4

    ****Connect Four****

|   |   |   |   |   |   |   |
-----------------------
|   |   |   |   |   |   |   |
-----------------------
|   |   |   |   |   |   |   |
-----------------------
|   |   | X | O |   |   |   |
-----------------------
| X | X | O | X |   |   |   |
-----------------------
| X | O | O | O |   |   |   |
-----------------------
  1   2   3   4   5   6   7

Player 1 (X):
Enter number coordinate: 4

    ****Connect Four****

|   |   |   |   |   |   |   |
-----------------------
|   |   |   |   |   |   |   |
-----------------------
|   |   |   | X |   |   |   |
-----------------------
|   |   | X | O |   |   |   |
-----------------------
| X | X | O | X |   |   |   |
-----------------------
| X | O | O | O |   |   |   |
-----------------------
  1   2   3   4   5   6   7

Player 1 (X) wins!
Would you like to play again? (Y/N): Would you like to play again? (Y/N):
```