CSCI 4220 Assignment 3

CRI Server

Due Date: Monday, November 5th, 11:59:59 PM

This is a team-based (max. 2) assignment. We want to build a "Chatting Remotely over the Internet" (CRI) server. The protocol is similar to a nearly unknown chat server from decades ago known as IRC. Our CRI server must support an arbitrary number of users and channels. Each user can be a member of zero or more channels. Channel names should *always* be prefixed with a hash (#). All commands, users, and channel names are case-sensitive. See the slides from Lecture 13.

Submit your code, a README file, and a Makefile that produces a hw3.out

Output your randomly chosen port

Implement the following commands:

- USER
- LIST
- JOIN
- PART
- OPERATOR
- KICK
- PRIVMSG
- QUIT

USER <nickname>

The USER command must be the first command sent to the server immediately following a connection from a client. Otherwise, the server MUST disconnect. Users cannot change their nickname at a later time. User names should be at most 20 characters and should adhere to the following regular expression: `[a-zA-Z][_0-9a-zA-Z]*`

LIST [#channel]

The LIST command should output the current count and list all of the channels currently available on the server. If a valid channel is supplied, LIST should output the current count and print the names of all users on channel #channel.

JOIN <#channel>

The JOIN command must add the current user to the named channel. If the named channel does not exist, it must be created. Channel names not starting with a # should not be created. Users currently on the channel

1

should receive a message that USER has joined. Channel names should be at most 20 characters and should adhere to the following regular expression: `#[a-zA-Z][_0-9a-zA-Z]*`

PART [#channel]

The PART command must remove the current user from the named channel. Users currently on the channel should receive a message that USER has left. PARTing from a channel the user is not a member of should produce an error message and not modify the user's current channel list. With no arguments, this command should remove the current user from all channels.

OPERATOR <password>

The OPERATOR command must make the current user an operator of the *server* if and only if the given password matches the password supplied on the server's command line. When starting the CRI server, an *optional* flag `--opt-pass=<password>` may be supplied. If no such flag is passed then no users can be promoted to operator status. If the password is incorrect (or none was given on the command line) the server should print an error message but not disconnect the user. Passwords should be at most 20 characters and should adhere to the following regular expression: `[a-zA-Z][_0-9a-zA-Z]*`

KICK <#channel> <user>

The KICK command must remove the named user from the named channel if and only if the current user is an operator of the server.

PRIVMSG (<#channel> | <user>) <message>

The PRIVMSG command will send the specified message to either all members of the named channel, or to the named user alone. Messages can be any printable ASCII character strings at most 512 characters in length. For individual users, `<<Sender` shows a received message and `Recipient>>` shows a sent message.

QUIT

The QUIT command will remove the current user from all channels and disconnect from the server.

Example output below. User input is in bold.

```
$ ./cri.out --opt-pass=netprog
35232
```

On another terminal, use netcat to connect to the CRI server:
```
$ nc localhost 35232
JOIN #programming
Invalid command, please identify yourself with USER.
```

```
$ nc localhost 35232
OPERATOR foobar
Invalid command, please identify yourself with USER.
```

```
$ nc localhost 35232
USERblah
Invalid command.
USER justin
Welcome, justin
```

```
LIST
There are currently 3 channels.
* netprog
* llvm
* csci4220
LIST #netProg
There are currently 3 channels.
* netprog
* llvm
* csci4220
LIST #netprog
There are currently 3 members.
#netprog members: Alice Bob grace
JOIN #netprog
Joined channel #netprog
#netprog> Alice: Hi Justin!
#netprog> Bob: what's up man?
<<Bob I need some help on the llvm channel when you get a minute!
#netprog> Alice left the channel.
PRIVMSG #netprog Hey guys how's it going?
#netprog> justin: Hey guys how's it going?
#netprog> grace: Hey you guys should install some backdoors in this service
PRIVMSG grace Did you leave the stove on?
Grace>> Did you leave the stove on?
OPERATOR NetProg
Invalid OPERATOR command.
OPERATOR netprog
OPERATOR status bestowed.
KICK #netprog grace
#netprog> grace has been kicked from the channel.
PART #NetProg
You are not currently in #NetProg.
PART #netprog
#netprog> justin left the channel.
```

Meanwhile, on grace's terminal:

```
...
#llvm> Bob: I have a question about how to access instruction dependencies in LLVM
#llvm> Bob: hmm I wish Justin would log on
#netprog> justin joined the channel.
#netprog> Alice: Hi Justin!
#netprog> Bob: what's up man?
#netprog> Alice left the channel.
PRIVMSG #netprog Hey you guys should install some backdoors in this service
#netprog> grace: Hey you guys should install some backdoors in this service
#netprog> grace has been kicked from the channel.
<<justin Did you leave the stove on? QUIT
$
```

Extra Credit: IPv4 and IPv6 support

In addition to supporting IPv4, include IPv6 support. Correctly implementing this will be worth an additional 20 points.