Rounak Jain 1RV18CS139

(4) Implement a simple multicast routing mechanism.

<u>sender.c</u> :

```
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <arpa/inet.h>
#include <time.h>
#include <string.h>
#include <stdio.h>
#include <stdlib.h>
#define HELLO-PORT 12345
#define HELLO-GROUP 255.0.0.37
int main (int argc, char * argv[])
{
   struct sockaddr_in addr;
   int fd, cnt;
   struct ip_mreq mreq;
   char * message = "RVCE -CSE";
   if ((fd = socket (AF-INET, SOCK_STREM, 0)){
             perror ("socket");
             exit(1);
   }
   memset (& addr, 0, sizeof (addr));
   addr.sin_family = AF-INET;
   addr.sin_addr.s_addr = inet_addr (HELLO_GROUP);
   addr.sin_port = htons (HELLO_PORT);
   while(1) {
       if (sendto (fd, message , sizeof (message ), 0 , (struct sockaddr*)
                                                   addr, sizeof(addr))<0) }
```

```
            perror ("sendto");
            exit (1);
         }
         sleep (1);
      }
      return 0;
   }
```

listener.c

```
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <arpa/inet.h>
#include <time.h>
#include <string.h>
#include <stdio.h>
#include <stdlib.h>
#define HELLO_PORT 12345
#define HELLO_GROUP "255.0.0.37"
#define MSGBUFSIZE 25
int main (int argc, char *argv[]) {
      struct sockaddr_in addr;
      int fd, nbytes, addrlen;
      struct ip_mreq mreq;
      char msgbuf [MSGBUFSIZE];
      u_int yes=1;
      if ( (fd = socket(AF_INET, SOCK_STREAM, 0) {
            perror ("socket"); exit (1);
      }
if (setsockopt (fd, SOL-SOCKET, SO-REUSEADDR, &yes,
                                        sizeof (yes)) <0){
    perror ("Reusing ADD failed");
    exit(1);
}
memset (&addr, 0, sizeof (addr)) ;
addr.sin_family =AF-INET;
addr .sin_addr .s-addr = htonl (INADDR -ANY);
addr. sin-port = htons(HELLO _PORT);
if ( bind (fd, (struct sockaddr* )&addr, sizeof(addr))<0)){
        perror ("bind");
        exit (1);
}
mreq. imr _multiaddr. s_addr = inet -addr (HELLO _GROUP);
mreq. imr_interface .s-addr =htonl (INADDR_ANY);
if (setsockopt (fd, IPPROTO _IP, IP_ADD_MEMBERSHIP,
                                &mreq, sizeof(mreq))
                                              <0){
      perror("setsockopt");
      exit(1);
}
while (1){
   addrlen=sizeof (addr);
   if ((nbytes =recvfrom (fd, msgbuf , MSGBUFSIZE, 0,
                      (struct sockaddr*)&addr,
                            &addrlen) <0) {
       perror ("recvfrom"); exit(1);
   }
   puts (msgbuf);
 }
}
```