#include<sys/types.h>

#include<sys/socket.h>

#include<sys/stat.h>

#include<stdio.h>

#include<stdlib.h>

#include<fcntl.h>

#include<unistd.h>

#include<netinet/in.h>

#include<arpa/inet.h>

void str\_echo(int connfd,int port){

int n,bufsize = 1024,len;

char \*buff = malloc(bufsize);

struct sockaddr\_in addr;

again: while((n=recv(connfd,buff,bufsize,0))>0){

printf("From client connected to %d :",port);

fputs(buff,stdout);

printf("Reply to the client connected to %d :",port);

fgets(buff,bufsize,stdin);

send(connfd,buff,n,0);

}

if(n<0)

goto again;

}

int main(){

int listenfd,connfd,addrlen,pid;

struct sockaddr\_in address;

if((listenfd = socket(AF\_INET,SOCK\_STREAM,0)) > 0)

printf("The socket was created\n");

else

printf("Error in Socket creation\n");

address.sin\_family = AF\_INET;

address.sin\_addr.s\_addr = INADDR\_ANY;

address.sin\_port= htons(15001);

if( bind( listenfd,(struct sockaddr \*)& address,sizeof(address)) ==0)

printf("Binding Socket\n");

else

printf("ERROR in binding\n");

if ((listen(listenfd, 3)) != 0){

printf("Listen failed\n");

exit(0);

}

else{

getsockname(listenfd,(struct sockaddr \*) &address,&addrlen);

printf("Server listening on port %d\n",address.sin\_port);

}

for(;;){

addrlen = sizeof(struct sockaddr\_in);

connfd = accept(listenfd,(struct sockaddr \*)& address,&addrlen);

if(connfd>0)

printf("A new client connected from port :%d \n",

address.sin\_port);

else

printf("A new client's connection wasn't accepted\n");

if((pid=fork())==0){

close(listenfd);

str\_echo(connfd,address.sin\_port);

exit(0);

}

close(connfd);

}

return 0;

}