CN lab program

//Tcp(server)

```
#include <stdio.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <string.h>
int main(){
 int welcomeSocket, newSocket;
 char buffer[1024];
 struct sockaddr_in serverAddr;
 struct sockaddr_storage serverStorage;
 socklen_t addr_size;
 welcomeSocket = socket(PF_INET, SOCK_STREAM, 0);
 serverAddr.sin_family = AF_INET;
 serverAddr.sin_port = htons(7891);
 serverAddr.sin_addr.s_addr = inet_addr("127.0.0.1");
 memset(serverAddr.sin_zero, '\0', sizeof serverAddr.sin_zero);
 bind(welcomeSocket, (struct sockaddr *) &serverAddr, sizeof(serverAddr));
 if(listen(welcomeSocket,5)==0)
  printf("Listening\n");
 else
  printf("Error\n");
 addr_size = sizeof serverStorage;
 newSocket = accept(welcomeSocket, (struct sockaddr *) &serverStorage, &addr_size);
 strcpy(buffer,"Hello World\n");
 send(newSocket,buffer,13,0);
 return 0;
}
```

//Tcp (client)

```
#include <stdio.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <string.h>
int main(){
 int clientSocket;
 char buffer[1024];
 struct sockaddr_in serverAddr;
 socklen_t addr_size;
 clientSocket = socket(PF_INET, SOCK_STREAM, 0);
 serverAddr.sin_family = AF_INET;
```

```
 serverAddr.sin_port = htons(7891);
 serverAddr.sin_addr.s_addr = inet_addr("127.0.0.1");
 memset(serverAddr.sin_zero, '\0', sizeof serverAddr.sin_zero);
 addr_size = sizeof serverAddr;
 connect(clientSocket, (struct sockaddr *) &serverAddr, addr_size);
 recv(clientSocket, buffer, 1024, 0);
 printf("Data received: %s",buffer);
 return 0;
}
```

//udp(server)

```
#include<stdio.h>
#include<netinet/in.h>
#include<sys/types.h>
#include<sys/socket.h>
#include<netdb.h>
#include<string.h>
#include<stdlib.h>
#define MAX 80
#define PORT 43454
#define SA struct sockaddr
void func(int sockfd)
{
char buff[MAX];
int n,clen;
struct sockaddr_in cli;
clen=sizeof(cli);
for(;;)
{
bzero(buff,MAX);
recvfrom(sockfd,buff,sizeof(buff),0,(SA *)&cli,&clen);
printf("From client %s To client",buff);
bzero(buff,MAX);
n=0;
while((buff[n++]=getchar())!='\n');
sendto(sockfd,buff,sizeof(buff),0,(SA *)&cli,clen);
if(strncmp("exit",buff,4)==0)
{
printf("Server Exit...\n");
break;
}
```

```
}
}
int main()
{
int sockfd;
struct sockaddr_in servaddr;
sockfd=socket(AF_INET,SOCK_DGRAM,0);
if(sockfd==-1)
{
printf("socket creation failed...\n");
exit(0);
}
else
printf("Socket successfully created..\n");
bzero(&servaddr,sizeof(servaddr));
servaddr.sin_family=AF_INET;
servaddr.sin_addr.s_addr=htonl(INADDR_ANY);
servaddr.sin_port=htons(PORT);
if((bind(sockfd,(SA *)&servaddr,sizeof(servaddr)))!=0)
{
printf("socket bind failed...\n");
exit(0);
}
else
printf("Socket successfully binded..\n");
func(sockfd);
close(sockfd);
}
```

//udp(client)

```
#include<sys/socket.h>
#include<netdb.h>
#include<string.h>
#include<stdlib.h>
#include<stdio.h>
#define MAX 80
#define PORT 43454
#define SA struct sockaddr
int main()
{
char buff[MAX];
```

```
int sockfd,len,n;
struct sockaddr_in servaddr;
sockfd=socket(AF_INET,SOCK_DGRAM,0);
if(sockfd==-1)
{
printf("socket creation failed...\n");
exit(0);
}
else
printf("Socket successfully created..\n");
bzero(&servaddr,sizeof(len));
servaddr.sin_family=AF_INET;
servaddr.sin_addr.s_addr=inet_addr("127.0.0.1");
servaddr.sin_port=htons(PORT);
len=sizeof(servaddr);
for(;;)
{
printf("\nEnter string : ");
n=0;
while((buff[n++]=getchar())!='\n');
sendto(sockfd,buff,sizeof(buff),0,(SA *)&servaddr,len);
bzero(buff,sizeof(buff));
recvfrom(sockfd,buff,sizeof(buff),0,(SA *)&servaddr,&len);
printf("From Server : %s\n",buff);
if(strncmp("exit",buff,4)==0)
{
printf("Client Exit...\n");
break;
}
}
close(sockfd);
}
```

//Tcpftp(server)

```
#include<stdio.h>
#include<sys/types.h>
#include<sys/socket.h>
#include<netinet/in.h>
#include<arpa/inet.h>
#include<fcntl.h>
#include<string.h>
```

```
main()
{
   struct sockaddr_in clientaddr,serveraddr;
   int serversock,newserversock,clientsize,n,f,rc;
   char filename[100],filedata[300];
   fflush(stdin);
   serversock=socket(AF_INET,SOCK_STREAM,0);
   bzero((char*)&serveraddr,sizeof(serveraddr));
   serveraddr.sin_family=AF_INET;
   serveraddr.sin_port=2000;
   serveraddr.sin_addr.s_addr=inet_addr("127.0.0.1");
   bind(serversock,(struct sockaddr*)&serveraddr,sizeof(serveraddr));
   sizeof(serveraddr);
   listen(serversock,5);
   while(1)
   {
      clientsize=sizeof(clientaddr);
      newserversock=accept(serversock,(struct sockaddr*)&clientaddr,&clientsize);
      n=read(newserversock,filename,100);
      filename[n]=0;
      printf("\nThe requested file from the client is  %s.\n",filename);
      //write(1,filename,n);
      f=open(filename,O_RDWR);
      n=read(f,filedata,300);
      printf("\nThe contents of the file: \n\n");
      printf("%s",filedata);
      write(newserversock,filedata,n);
   }
   close(serversock);
   close(newserversock);
}
```

//Tcpftp(client)

```
#include<stdio.h>
#include<sys/types.h>
#include<sys/stat.h>
#include<sys/socket.h>
#include<netinet/in.h>
#include<arpa/inet.h>
#include<fcntl.h>
#include<string.h>
```

```
#include<netdb.h>
#include<stdlib.h>

main()
{
   struct sockaddr_in serveraddr;
   int clientsock,n,rdret,length;
   char filename[20],filedata[300];
   bzero((char*)&serveraddr,sizeof(serveraddr));
   serveraddr.sin_family=AF_INET;
   serveraddr.sin_port=2000;
   serveraddr.sin_addr.s_addr=inet_addr("127.0.0.1");
   clientsock=socket(AF_INET,SOCK_STREAM,0);
   if(connect(clientsock,(struct sockaddr*)&serveraddr,sizeof(serveraddr))<0)
   {
      printf("\nError:Cannot connect...");
      exit(0);
   }
   printf("Enter the name of the file : ");
   scanf("%s",filename);
   length=strlen(filename);
   write(clientsock,filename,length);
   rdret=read(clientsock,filedata,300);
   printf("\nThe contents of the file: \n\n");
   printf("%s",filedata);
   close(clientsock);
}
```

//Udpftp(client)

```
#include<stdio.h>
#include<string.h>
#include<sys/stat.h>
#include<sys/socket.h>
#include<sys/types.h>
#include<netinet/in.h>
#include<arpa/inet.h>
#include<fcntl.h>
#include<stdlib.h>
main()
{
   struct sockaddr_in server,client;
```

```
    int s,n,ret;size_t fp;
    char filename[20],downloaded[10],filedata[100],c[25];
    mode_t mode = S_IRUSR | S_IWUSR | S_IRGRP | S_IROTH;
    s=socket(AF_INET,SOCK_DGRAM,0);
    server.sin_family=AF_INET;
    server.sin_port=2000;
    server.sin_addr.s_addr=inet_addr("127.0.0.1");
    n=sizeof(server);
    printf("Enter the name of the file: ");
    scanf("%s",filename);
    printf("\nEnter a name to save: ");
    scanf("%s",downloaded);
    printf("\nDownloading...\n");
    sendto(s,filename,sizeof(filename),0,(struct sockaddr *)&server,n);
    fp = open(downloaded, O_WRONLY | O_CREAT | O_TRUNC, mode);
    if(fp==-1)
    {
       printf("\nError...");
       exit(0);
    }
    recvfrom(s,filedata,sizeof(filedata),0,NULL,NULL);
    printf("\nProcessing Contents...\n");
    while(1)
    {
       if(strcmp(filedata,"end")==0)
          break;
    printf("%s",filedata);
       ret=write(fp,filedata,strlen(filedata));
       bzero(filedata,100);
       recvfrom(s,filedata,sizeof(filedata),0,NULL,NULL);
    }
    printf("\nFile downloaded successfully...\n");
}
```

//Udpftp(server)

```
#include<stdio.h>
#include<sys/types.h>
#include<sys/stat.h>
#include<sys/socket.h>
#include<netinet/in.h>
#include<arpa/inet.h>
```

```
#include<fcntl.h>
#include<string.h>
main()
{
   struct sockaddr_in server,client;
   int serversock,n,fp,end;
   char filename[20],buffer[100];
   serversock=socket(AF_INET,SOCK_DGRAM,0);
   server.sin_family=AF_INET;
   server.sin_port=2000;
   server.sin_addr.s_addr=inet_addr("127.0.0.1");
   bind(serversock,(struct sockaddr *)&server,sizeof(server));
   n=sizeof(client);
   recvfrom(serversock,filename,sizeof(filename), 0,(struct sockaddr *)&client,&n);
   fp=open(filename,O_RDONLY);
   while(1)
   {
      end=read(fp,buffer,sizeof(buffer));
      if(end==0)
         break;
      sendto(serversock,buffer,sizeof(buffer),0,(struct sockaddr *)&client,n);
      bzero(buffer,100);
   }
   strcpy(buffer,"end");
   sendto(serversock,buffer,sizeof(buffer),0,(struct sockaddr *)&client,n);
}
```

//Sliding window(rec)

```
#include<sys/socket.h>
#include<sys/types.h>
#include<netinet/in.h>
#include<netdb.h>
#include<stdio.h>
#include<string.h>
#include<stdlib.h>
#include<unistd.h>
#include<errno.h>
int main()
{
int sock,bytes_received,i=1;
char receive[30];
```

```
struct hostent *host;
struct sockaddr_in server_addr;
host=gethostbyname("127.0.0.1");
if((sock=socket(AF_INET,SOCK_STREAM,0))==-1)
{
perror("Socket not created");
exit(1);
}
printf("Socket created");
server_addr.sin_family=AF_INET;
server_addr.sin_port=htons(17000);
server_addr.sin_addr=*((struct in_addr *)host->h_addr);
bzero(&(server_addr.sin_zero),8);
if(connect(sock,(struct sockaddr *)&server_addr,sizeof(struct sockaddr))==-1)
{
perror("Connect");
exit(1);
}
while(1)
{
bytes_received=recv(sock,receive,20,0);
receive[bytes_received]='\0';
if(strcmp(receive,"exit")==0||strcmp(receive,"exit")==0)
{
close(sock);
break;
}
else
{
if(strlen(receive)<10)
{
printf("\n Frame %d data %s received\n",i,receive);
send(0,receive,strlen(receive),0);
}
else
{
send(0,"negative",10,0);
}
i++;
}
}
```

```
close(sock);
return(0);
}
```

//Sliding window(snd)

```
#include<sys/socket.h>
#include<sys/types.h>
#include<netinet/in.h>
#include<netdb.h>
#include<stdio.h>
#include<string.h>
#include<stdlib.h>
#include<unistd.h>
#include<errno.h>
int main()
{
int sock,bytes_received,connected,true=1,i=1,s,f=0,sin_size;
char send_data[1024],data[1024],c,fr[30]=" ";
struct sockaddr_in server_addr,client_addr;
if((sock=socket(AF_INET,SOCK_STREAM,0))==-1)
{
perror("Socket not created");
exit(1);
}
if(setsockopt(sock,SOL_SOCKET,SO_REUSEADDR,&true,sizeof(int))==-1)
{
perror("Setsockopt");
exit(1);
}
server_addr.sin_family=AF_INET;
server_addr.sin_port=htons(17000);
server_addr.sin_addr.s_addr=INADDR_ANY;
if(bind(sock,(struct sockaddr *)&server_addr,sizeof(struct sockaddr))==-1)
{
perror("Unable to bind");
exit(1);
}
if(listen(sock,5)==-1)
{
perror("Listen");
exit(1);
```

```
}
fflush(stdout);
sin_size=sizeof(struct sockaddr_in);
connected=accept(sock,(struct sockaddr *)&client_addr,&sin_size);
while(strcmp(fr,"exit")!=0)
{
printf("Enter Data Frame %d:(Enter exit for End): ",i);
scanf("%s",fr);
send(connected,fr,strlen(fr),0);
recv(sock,data,1024,0);
if(strlen(data)!=0)
printf("I got an acknowledgement : %s\n",data);
fflush(stdout);
i++;
}
close(sock);
return (0);
}
```

//date and time

Server

```
#include <stdio.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <string.h>
#include <arpa/inet.h>
#include<time.h>
Int main(){
  Int welcomeSocket, newSocket;
  Char buffer[1024];
  Struct sockaddr_in serverAddr;
  Struct sockaddr_storage serverStorage;
  Socklen_t addr_size;
Time_t ticks;
  welcomeSocket = socket(PF_INET, SOCK_STREAM, 0);
  serverAddr.sin_family = AF_INET;
  serverAddr.sin_port = htons(7896);
  serverAddr.sin_addr.s_addr = inet_addr("127.0.0.1");
  Memset(serverAddr.sin_zero, '\0', sizeof serverAddr.sin_zero);
  Bind(welcomeSocket, (struct sockaddr *) &serverAddr, sizeof(serverAddr));
```

```
 If(listen(welcomeSocket,5)==0)
  Printf("Listening\n");
 Else
  Printf("Error\n");
 Addr_size = sizeof serverStorage;
 newSocket = accept(welcomeSocket, (struct sockaddr *) &serverStorage, &addr_size);
Ticks=time(NULL);
Strcpy(buffer, ctime(&ticks));
 //strcpy(buffer,"Hello World\n");
 Send(newSocket,buffer,50,0);
 Return 0;
}
Client:
#include <stdio.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <string.h>
#include<arpa/inet.h>
Int main(){
 Int clientSocket;
 Char buffer[1024];
 Struct sockaddr_in serverAddr;
 Socklen_t addr_size;
 clientSocket = socket(PF_INET, SOCK_STREAM, 0);
 serverAddr.sin_family = AF_INET;
 serverAddr.sin_port = htons(7896);
 serverAddr.sin_addr.s_addr = inet_addr("127.0.0.1");
 Memset(serverAddr.sin_zero, '\0', sizeof serverAddr.sin_zero);
 Addr_size = sizeof serverAddr;
 Connect(clientSocket, (struct sockaddr *) &serverAddr, addr_size);
 Recv(clientSocket, buffer, 1024, 0);
 Printf("Data received: %s",buffer);
 Return 0;
}
```