TÉCNICO LISBOA

```
#include <unistd.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <netdb.h>
#define PORT 58000
...
int fd, newfd;
struct hostent *hostptr;
struct sockaddr_in serveraddr, clientaddr;
int clientlen;
...
```

TCP Server

```
fd=socket(AF_INET,SOCK_STREAM,0);
```

```
memset((void*)&serveraddr,(int)'\0',
       sizeof(serveraddr));
serveraddr.sin_family=AF_INET;
serveraddr.sin_addr.s_addr=htonl(INADDR_ANY);
serveraddr.sin_port=htons((u_short)PORT);
```

```
bind(fd,(struct sockaddr*)&serveraddr,
         sizeof(serveraddr));
```

```
listen(fd,5);
```

```
clientlen=sizeof(clientaddr);
newfd=accept(fd,(struct sockaddr*)&clientaddr,
             &clientlen);
```

blocks until connection from client

TCP Client

```
fd=socket(AF_INET,SOCK_STREAM,0);
```

```
hostptr=gethostbyname("tejo.ist.utl.pt");

memset((void*)&serveraddr,(int)'\0',
       sizeof(serveraddr));
serveraddr.sin_family=AF_INET;
serveraddr.sin_addr.s_addr=((struct in_addr *)
    (hostptr->h_addr_list[0]))->s_addr;
serveraddr.sin_port=htons((u_short)PORT);
```

```
connect(fd,(struct sockaddr*)&serveraddr,
        sizeof(serveraddr));
```

connection establishment

TCP three-way handshake

```
write(fd,...);
...
read(fd,...);
...
close(fd);
```

```
read(newfd,...);
...
write(newfd,...);
...


close(fd); close(newfd);
```

TÉCNICO LISBOA

```
#include <unistd.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <netdb.h>
#define PORT 59000
...
int fd;
struct hostent *hostptr;
struct sockaddr_in serveraddr, clientaddr;
int addrlen;
...
```

UDP Server

```
fd=socket(AF_INET,SOCK_DGRAM,0);
```

```
memset((void*)&serveraddr,(int)'\0',
       sizeof(serveraddr));
serveraddr.sin_family=AF_INET;
serveraddr.sin_addr.s_addr=htonl(INADDR_ANY);
serveraddr.sin_port=htons((u_short)PORT);
```

```
bind(fd,(struct sockaddr*)&serveraddr,
         sizeof(serveraddr));
```

```
addrlen=sizeof(clientaddr);
recvfrom(fd,buffer,sizeof(buffer),0,
          (struct sockaddr*)&clientaddr,
          &addrlen);
```

blocks until datagram received from a client

```
...
sendto(fd,msg,strlen(msg)+1,0,
       (struct sockaddr*)&clientaddr,addrlen);
...

close(fd);
```

UDP Client

```
fd=socket(AF_INET,SOCK_DGRAM,0);
```

```
hostptr=gethostbyname("tejo.ist.utl.pt");

memset((void*)&serveraddr,(int)'\0',
       sizeof(serveraddr));
serveraddr.sin_family=AF_INET;
serveraddr.sin_addr.s_addr=((struct in_addr *)
    (hostptr->h_addr_list[0]))->s_addr;
serveraddr.sin_port=htons((u_short)PORT);

addrlen=sizeof(serveraddr);
```

```
sendto(fd,msg,strlen(msg)+1,0,
       (struct sockaddr*)&serveraddr,addrlen);
...
addrlen=sizeof(serveraddr);
recvfrom(fd,buffer,sizeof(buffer),0,
         (struct sockaddr*)&serveraddr,&addrlen);
...

close(fd);
```