韩国尹圣雨《TCP/IP网络编程》阅读笔记

*TCP套接字通信可以类比电话机（server），接听方法流程如下：*

**socket-------安装电话机**

**bind-------分配电话号码**

**listen-------连接电话线**

**accept---------拿起话筒**

*打电话端（client）只需要如下两个函数：*

**socket----------安装电话机**

**connect---------拨号**

实例（只能处理单一请求）

服务器端接受连接请求，向请求的客户端返回“HelloWorld！”：

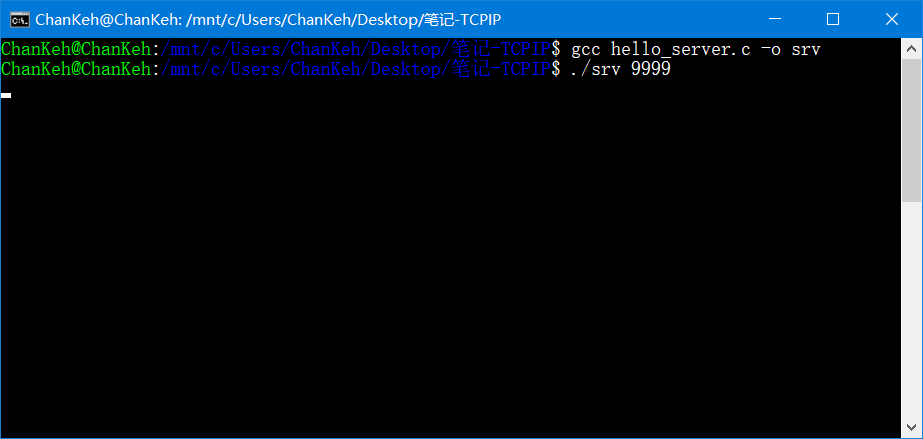
1. #include **<stdio.h>**
2. #include **<stdlib.h>**
3. #include **<string.h>**
4. #include **<unistd.h>**
5. #include **<arpa**/inet.h**>**
6. #include **<sys**/socket.h**>**
8. void error\_handling(char \*message);
10. int main(int argc, char \*argv[]){
11. int serv\_sock;
12. int clnt\_sock;
14. struct sockaddr\_in serv\_addr;
15. struct sockaddr\_in clnt\_addr;
16. socklen\_t clnt\_addr\_size;
18. char message[]="hello world!";
19. if(argc != 2){
20. printf("Usage : %s **<port>**\n",argv[0]);
21. exit(1);
22. }
23. serv\_sock = socket(PF\_INET,SOCK\_STREAM,0);
24. if(serv\_sock == -1){
25. error\_handling("socket() error");
26. }
28. memset(&serv\_addr,0,sizeof(serv\_addr));
29. serv\_addr.sin\_family=AF\_INET;
30. serv\_addr.sin\_addr.s\_addr=htonl(INADDR\_ANY);
31. serv\_addr.sin\_port=htons(atoi(argv[1]));
33. if(bind(serv\_sock,(struct sockaddr\*)&serv\_addr,sizeof(serv\_addr)) == -1){
34. error\_handling("bind() error");
35. }
37. if(listen(serv\_sock,5) == -1){
38. error\_handling("listen() error");
39. }
41. clnt\_addr\_size = sizeof(clnt\_addr);
42. clnt\_sock = accept(serv\_sock,(struct sockaddr \*)&clnt\_addr,&clnt\_addr\_size);
43. if(clnt\_sock == -1){
44. error\_handling("accept() error");
45. }
47. write(clnt\_sock,message,sizeof(message));
49. close(clnt\_sock);
50. close(serv\_sock);
52. return 0;
54. }
56. void error\_handling(char \*message){
58. fputs(message,stderr);
59. fputs("\n",stderr);
60. exit(1);
61. }

client客户端实现代码：

1. #include <stdio.h>
2. #include <stdlib.h>
3. #include <string.h>
4. #include <unistd.h>
5. #include <arpa/inet.h>
6. #include <sys/socket.h>
8. **void** error\_handling(**char** \*message);
10. **int** main(**int** argc,**char** \*argv[]){
11. **int** sock;
12. **struct** sockaddr\_in serv\_addr;
13. **char** message[30];
14. **int** str\_len;
16. **if**(argc != 3){
17. printf("Usage : %s <IP> <port>\n",argv[0]);
18. exit(1);
19. }
21. sock = socket(PF\_INET,SOCK\_STREAM,0);
22. **if**(sock == -1){
23. error\_handling("socket() error");
24. }
26. memset(&serv\_addr,0,**sizeof**(serv\_addr));
27. serv\_addr.sin\_family = AF\_INET;
28. serv\_addr.sin\_addr.s\_addr = inet\_addr(argv[1]);
29. serv\_addr.sin\_port = htons(atoi(argv[2]));
31. **if**(connect(sock,(**struct** sockaddr \*)&serv\_addr,**sizeof**(serv\_addr)) == -1){
32. error\_handling("connect() error\r\n");
33. }
34. str\_len = read(sock,message,**sizeof**(message)-1);
35. **if**(str\_len == -1)
36. {
37. error\_handling("read() error");
38. }
40. printf("Message from server : %s \n",message);
42. close(sock);
43. **return** 0;
44. }
46. **void** error\_handling(**char** \*message){
47. fputs(message,stderr);
48. fputs("\n",stderr);
49. exit(1);
50. }

运行结果：

服务器端运行:



客户端运行、服务端反馈结果：

