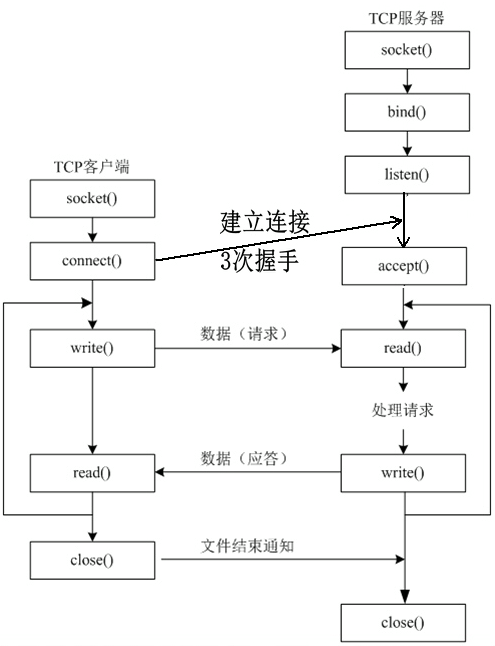
# 一、tcp编程的核心步骤



# 二、connect()函数分析

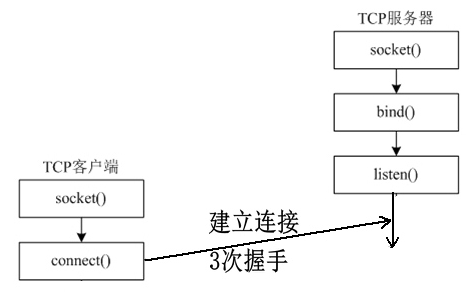
这个函数的作用仅仅是通知 Linux 内核，让 Linux 内核自动完成 TCP 三次握手连接，客户端的 connect() 函数默认会一直阻塞，直到三次握手成功或超时失败才返回

# 三、listen()函数

listen() 函数的主要作用就是将套接字( sockfd )变成被动连接的监听套接字（被动等待客户端的连接），至于参数 backlog 的作用是设置内核中连接队列的长度

listen()函数不会阻塞，它主要做的事情为，将该套接字和套接字对应的连接队列长度告诉 Linux 内核，然后，listen()函数就结束。

这样的话，当有一个客户端主动连接（connect()），Linux 内核就自动完成[TCP 三次握手](http://blog.csdn.net/tennysonsky/article/details/45622395)，将建立好的链接自动存储到队列中，如此重复。



## 三次握手的连接队列

这里详细的介绍一下 listen() 函数的第二个参数（ backlog）的作用：告诉内核连接队列的长度。

内核为任何一个给定的监听套接口维护一个队列，该队列由两部分构成，分别是完成连接接队列、未完成连接队列：

1、未完成连接队列（incomplete connection queue），当服务器每收到客户端的一个SYN分节，就会将该客户端放入未完成连接队列，而服务器套接口处于 SYN\_RCVD 状态。  
2、已完成连接队列（completed connection queue），当客户端和服务器彻底完成三次握手过程，客户端将从未完成连接队列升级成已完成连接队列，并从未完成连接队列中清空该客户端，这些套接口处于 ESTABLISHED 状态。

**backlog 参数**历史上被定义为上面两个队列的大小之和，大多数实现默认值为 5

# 四、accept()函数

accept()函数功能是，从连接队列头部取出一个已经完成的连接，如果这个队列没有已经完成的连接，accept()函数就会阻塞

如果，服务器不能及时调用 accept() 取走队列中已完成的连接，队列满掉后会怎样呢？

Unix服务器不会对再对建立新连接的syn进行应答，所以客户端的 connect 就会返回 ETIMEDOUT。  
  
Linux 则会延时连接，写程序时服务器的 listen() 的第二个参数最好还是根据需要填写，写太大不好，浪费资源，写太小也不好，延时建立连接。