

code/src/csapp.c

```

1  ssize_t rio_readn(int fd, void *usrbuf, size_t n)
2  {
3      size_t nleft = n;
4      ssize_t nread;
5      char *bufp = usrbuf;
6
7      while (nleft > 0) {
8          if ((nread = read(fd, bufp, nleft)) < 0) {
9              if (errno == EINTR) /* Interrupted by sig handler return */
10                 nread = 0;      /* and call read() again */
11              else
12                 return -1;      /* errno set by read() */
13          }
14          else if (nread == 0)
15              break;             /* EOF */
16          nleft -= nread;
17          bufp += nread;
18      }
19      return (n - nleft);        /* Return >= 0 */
20  }

```

*code/src/csapp.c**code/src/csapp.c*

```

1  ssize_t rio_writen(int fd, void *usrbuf, size_t n)
2  {
3      size_t nleft = n;
4      ssize_t nwritten;
5      char *bufp = usrbuf;
6
7      while (nleft > 0) {
8          if ((nwritten = write(fd, bufp, nleft)) <= 0) {
9              if (errno == EINTR) /* Interrupted by sig handler return */
10                 nwritten = 0;    /* and call write() again */
11              else
12                 return -1;      /* errno set by write() */
13          }
14          nleft -= nwritten;
15          bufp += nwritten;
16      }
17      return n;
18  }

```

code/src/csapp.c

图 10-4 rio_readn 和 rio_writen 函数

每打开一个描述符，都会调用一次 `rio_readinitb` 函数。它将描述符 `fd` 和地址 `rp` 处的一个类型为 `rio_t` 的读缓冲区联系起来。

`rio_readlineb` 函数从文件 `rp` 读出下一个文本行(包括结尾的换行符)，将它复制到内存位置 `usrbuf`，并且用 `NULL`(零)字符来结束这个文本行。`rio_readlineb` 函数最多读 `maxlen-1` 个字节，余下的一个字符留给结尾的 `NULL` 字符。超过 `maxlen-1` 字节的文