

code/io/statcheck.c

```

1  #include "csapp.h"
2
3  int main (int argc, char **argv)
4  {
5      struct stat stat;
6      char *type, *readok;
7
8      Stat(argv[1], &stat);
9      if (S_ISREG(stat.st_mode))    /* Determine file type */
10         type = "regular";
11     else if (S_ISDIR(stat.st_mode))
12         type = "directory";
13     else
14         type = "other";
15     if ((stat.st_mode & S_IRUSR)) /* Check read access */
16         readok = "yes";
17     else
18         readok = "no";
19
20     printf("type: %s, read: %s\n", type, readok);
21     exit(0);
22 }

```

*code/io/statcheck.c*图 10-10 查询和处理一个文件的 `st_mode` 位

10.7 读取目录内容

应用程序可以用 `readdir` 系列函数来读取目录的内容。

```

#include <sys/types.h>
#include <dirent.h>

DIR *opendir(const char *name);

```

返回：若成功，则为处理的指针；若出错，则为 `NULL`。

函数 `opendir` 以路径名为参数，返回指向目录流(directory stream)的指针。流是对条目有序列表的抽象，在这里是指目录项的列表。

```

#include <dirent.h>

struct dirent *readdir(DIR *dirp);

```

返回：若成功，则为指向下一个目录项的指针；若没有更多的目录项或出错，则为 `NULL`。

每次对 `readdir` 的调用返回的都是指向流 `dirp` 中下一个目录项的指针，或者，如果没有更多目录项则返回 `NULL`。每个目录项都是一个结构，其形式如下：

```

struct dirent {
    ino_t d_ino;        /* inode number */
    char d_name[256];   /* Filename */
};

```

虽然有些 Linux 版本包含了其他的结构成员，但是只有这两个对所有系统来说都是标