

编写自己的ls程序

本程序用来仿照linux中的ls -l命令来实现的，主要运用的函数有opendir, readdir, lstat等。代码如下：

```
#include <iostream>
#include <vector>
#include <cstdlib>
#include <dirent.h>
#include <sys/types.h>
#include <sys/stat.h>
#include <unistd.h>
#include <cstring>
#include <algorithm>
using namespace std;

void getFileAndDir(vector<string> &, char *);
void displayFileDistribute(vector<string> &);

int main(int argc, char *argv[]){
    vector<string> file;

    if(argc != 2){
        cerr << "usage: ls /XXX" << endl;
        exit(1);
    }

    getFileAndDir(file, argv[1]);
    displayFileDistribute(file);

    /*vector<string>::iterator it = file.begin();
    for(; it != file.end(); ++it){
        //cout << *it << endl;
    }*/

    exit(0);
}

void getFileAndDir(vector<string> &file, char *dirName){
    DIR *dir;
    struct dirent *drt;

    dir = opendir(dirName);
```

```
if(dir == NULL){
    cerr << "Can't opendir " << dirName << endl;
    exit(1);
}

while((drt = readdir(dir)) != NULL){
    //cout << drt->d_name << endl;
    if(!strcmp(drt->d_name, ".") || !strcmp(drt->d_name, "..")){
        continue;
    }
    file.push_back(drt->d_name);
}

closedir(dir);
}

void displayFileDistribute(vector<string> &file){
    vector<string>::iterator it = file.begin();
    struct stat filestat;
    string perm[] = {"---", "--x", "-w-", "-wx", "r--", "r-x", "rw-", "rwx"};
    cout << "总计 " << file.size() << endl;
    sort(file.begin(), file.end());
    for(; it != file.end(); ++it){
        if(lstat((*it).c_str(), &filestat) == -1){
            cerr << "stat filed!" << endl;
            continue;
        }
        cout << filestat.st_ino << " "; //inode

        if(S_ISREG(filestat.st_mode))
            cout << "-";
        else if(S_ISDIR(filestat.st_mode))
            cout << "d";
        else if(S_ISCHR(filestat.st_mode))
            cout << "c";
        else if(S_ISBLK(filestat.st_mode))
            cout << "b";
        else if(S_ISFIFO(filestat.st_mode))
            cout << "p";
        else if(S_ISLNK(filestat.st_mode))
            cout << "l";
        else if(S_ISSOCK(filestat.st_mode))
            cout << "s";
        else
            cout << "X";
    }
}
```

```
//get perm
int i = 3;
unsigned int mask = 0700;
while(i > 0){
    cout << perm[(filestat.st_mode & mask) >> (i - 1) * 3];
    --i;
    mask >>= 3;
}

cout << " " << filestat.st_uid << " ";
cout << filestat.st_gid << " ";
string str = ctime(&filestat.st_atime);
cout << str.substr(0, str.length() - 9) << " ";
cout << *it << endl;
}

}
```

编译运行代码：程序需要接收一个目录参数比如 ./ls /home/wyp

```
总计 10
3145792 -rw----- 3005 3006 Fri Mar 23 10:07 .test.cpp.swp
3145793 -rwxr-xr-x 3005 3006 Tue Apr 3 14:41 2
3145798 -rw-r--r-- 3005 3006 Tue Apr 3 14:41 2.c
3145791 -rw-r--r-- 3005 3006 Wed Mar 21 21:13 cj
3145794 -rw-r--r-- 3005 3006 Tue Apr 3 14:41 cj.c
3145797 -rwxr-xr-x 3005 3006 Tue Apr 3 16:24 ls
3145802 -rw-r--r-- 3005 3006 Tue Apr 3 16:24 ls.cpp
3145795 -rw-r--r-- 3005 3006 Wed Mar 21 21:12 result
3145799 prw-r--r-- 3005 3006 Tue Apr 3 15:32 s
3145796 -rw-r--r-- 3005 3006 Tue Apr 3 14:41 tags
```

本博客文章除特别声明，全部都是原创！
原创文章版权归过往记忆大数据（[过往记忆](#)）所有，未经许可不得转载。
本文链接：[【】（）](#)