### ls

#### 规格说明

Usage：ls [-adilR] [name...]

Options：

-a: 显示隐藏文件(文件名以.开头的文件)

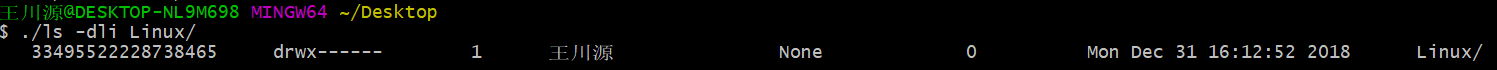
-d: 显示目录本身信息，不列出目录条目

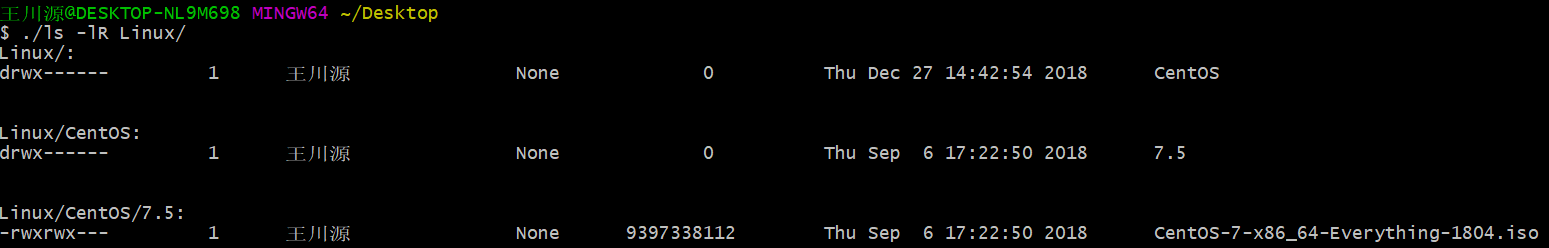
-i：显示文件的inode number

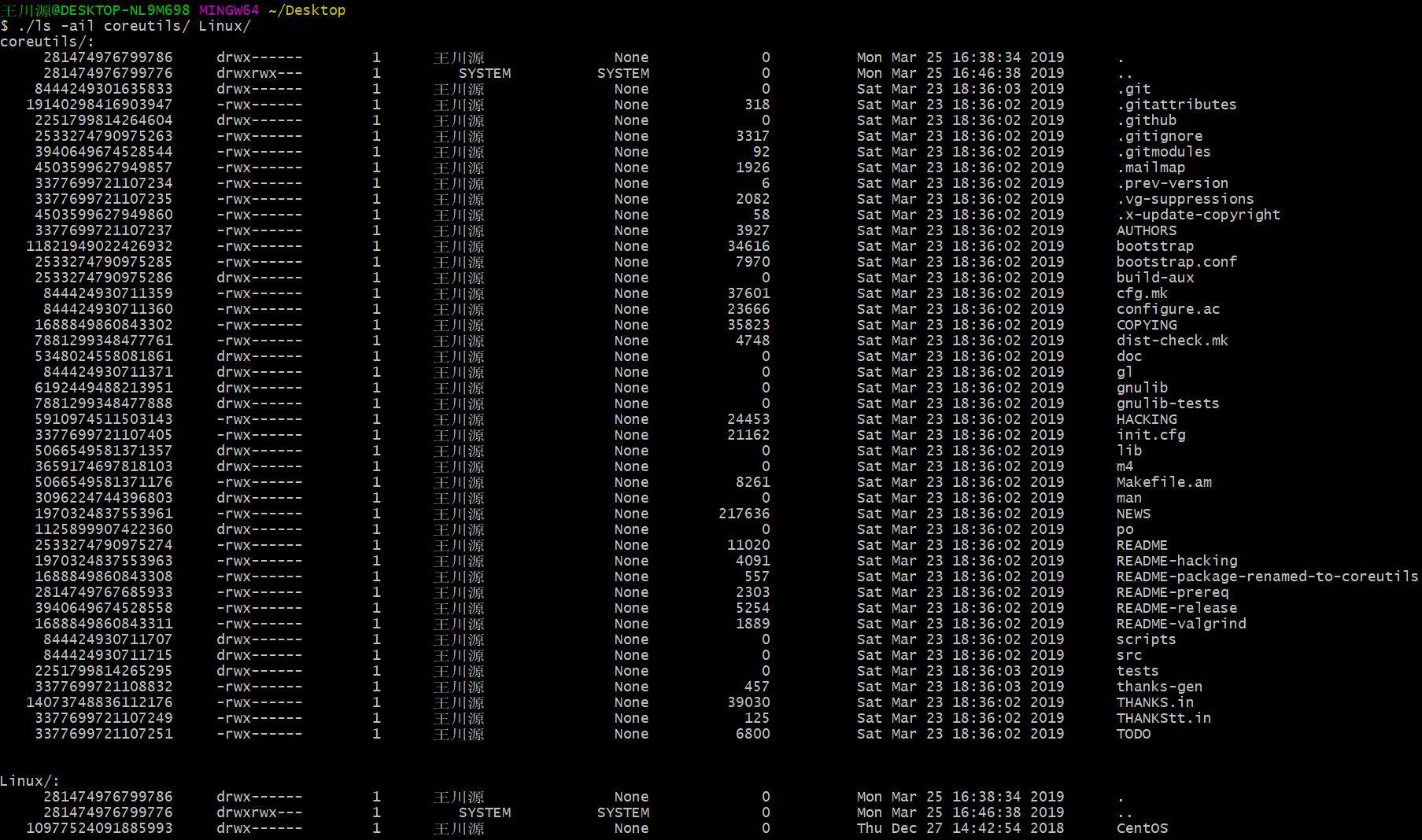
-l：显示文件的详细信息

-R：递归显示目录子条目信息

Example：







#### 代码设计

解析命令行参数，以’-’开头判定为Option，其余判定为Filename；

根据输入的Option，确定mode，设置对应标识；

对于每一个输入的Filename，判定为目录还是非目录文件，根据标识格式化输出。

#### 源码对比

源码有对于软链接处理，采用pending\_dirs以及active\_dir\_set两个数据结构来处理递归ls时软链接引起的死循环问题。

pending\_dirs是pending\_dir的链表，如果-R递归处理，则将子目录加入pending\_dirs；pending\_dir中通过realname来标识是否是软链接。

active\_dir\_set存放正处理的目录，如果子目录是父目录软链接，则inode相同，无法存入set，以此避免死循环。

### wc

#### 规格说明

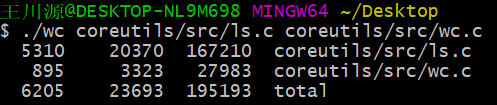
Usage：wc [filename...]

Output：

Entry format：line-count word-count byte-count filename

include total when the number of arguments is greater than one

Example：



#### 代码设计

对于每一个输入的Filename，统计行数，单词数，字节数，并输出，若输入参数大于1，输出合计。

统计步骤:

1. fgets读取1024个字符(char)
2. 对于每一个字符，字节数加1
3. 对于每一个字符，若为’\n’，行数加1
4. 对于每一个字符，若为’\t’或’\n‘或’ ’(空白字符)且之前读取连续非空白字符，单词数加1

#### 源码对比

源码计数逻辑在bool wc (int fd, char const \*file\_x, struct fstatus \*fstatus, off\_t current\_pos)中。

通过getc读取单个字符，并在switch中进行统计:

‘\n’: lines++

‘\r’, ‘\f’, ‘\t’, ‘ ‘, ‘\v’等goto到word\_separator中进行单词的判断，通过省略break语句和使用goto语句减少了条件判断的分支，使复杂情况简化。