

🔍 正则表达式基础 (/courses/90)

# 扩展正则表达式

## 一、实验介绍

### 1.1 实验内容

事实上，一般实验者只需要熟悉基础的正则表达式就足够了。不过有时候为了简化命令操作，了解一些使用范围更广的扩展表达式，会更加方便。本实验将介绍扩展正则表达式的基本知识。

### 1.2 实验知识点

- egrep 命令
- 扩展正则表达式

### 1.3 实验环境

课程使用的实验环境为 Ubuntu Linux 14.04 64 位版本。实验中会用到程序：

- Xfce终端

## 二、实验步骤

接下来开始实验。

### 2.1 简单对比

正规表示法:

```
$ grep -v '^$' regular_express.txt |grep -v '^#'
```

需要使用到管线命令来搜寻两次！那么如果使用延伸型的正规表示法，我们可以简化为：

```
$ egrep -v '^$|^#' regular_express.txt
```

利用支持延伸型正规表示法的 egrep 与特殊字符 “|” 的组功能来区隔两组字符串，如此一来，是否方便很多呢？

此外，grep 默认仅支持基础正则表达式，如果要使用扩展性正则表达式，可以使用 grep - E。  
⑨ 正则表达式基础 (/courses/90)  
grep -E 与 egrep 相当于命令别名关系。

## 2.2 扩展规则

### 2.2.1 + : 表示重复一个或一个以上的前一个 RE 字符

```
$ egrep -n 'go+d' regular_express.txt
```

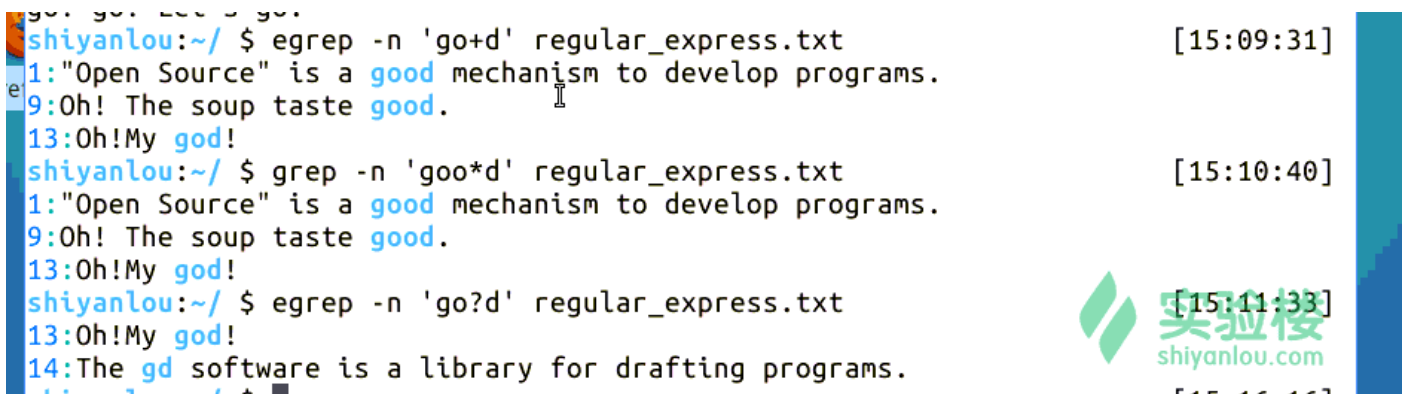
普通写法:

```
$ grep -n 'goo*d' regular_express.txt
```

### 2.2.2 ? : 表示重复零个或一个前一个 RE 字符

```
$ egrep -n 'go?d' regular_express.txt
```

操作截图：



```
shiyancelou:~/ $ egrep -n 'go+d' regular_express.txt [15:09:31]
1:"Open Source" is a good mechanism to develop programs.
9:Oh! The soup taste good.
13:Oh!My god!
shiyancelou:~/ $ grep -n 'goo*d' regular_express.txt [15:10:40]
1:"Open Source" is a good mechanism to develop programs.
9:Oh! The soup taste good.
13:Oh!My god!
shiyancelou:~/ $ egrep -n 'go?d' regular_express.txt [15:11:33]
13:Oh!My god!
14:The gd software is a library for drafting programs.
```

上述三条命令结果如上，发现 'goo\*d' 与 'go+d' 等同，而 'go?d' 结果不同。

### 2.2.3 | :表示用或的方式找出数个字符串

```
$ egrep -n 'gd|good' regular_express.txt
```

### 2.2.4 () : 表示找出群组字符串

```
$ egrep -n 'g(la|oo)d' regular_express.txt
```

也就是搜寻(glad)或 good 这两个字符串

操作截图：

```
shiyancelou:~/ $ egrep -n 'g(la|oo)d' regular_express.txt [15:26:03]
1: Open source is a good mechanism to develop programs.
9: Oh! The soup taste good.
16: The world <Happy> is the same with "glad".
shiyancelou:~/ $
```



## 2.2.5 ()+ : 多个重复群组判别

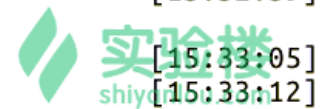
```
$ echo 'AxyzxyzxyzxyzC' | egrep 'A(xyz)+C'
$ echo 'AxyzxyzxyzxyzC' | egrep 'A(xz)+C'
```

也就是要找开头是 A 结尾是 C 中间有一个以上的 'xyz' 或 'xz' 字符串的意思。

结果显示 'A(xyz)+C' 可以匹配, 'A(xz)+C' 没有匹配项。

操作截图：

```
shiyancelou:~/ $ echo 'AxyzxyzxyzxyzC' | egrep 'A(xyz)+C' [15:32:59]
AxyzxyzxyzxyzC
shiyancelou:~/ $ echo 'AxyzxyzxyzxyzC' | egrep 'A(xz)+C' [15:33:05]
shiyancelou:~/ $
```



## 三、实验总结

到这里, 正则表达式的基本知识点就介绍完了, 希望同学们下来能够熟练地运用。

## 四、参考链接

- Shell正则表达式之grep、sed、awk实操笔记 (<http://www.jb51.net/article/54961.htm>)

\*本课程内容, 由作者授权实验楼发布, 未经允许, 禁止转载、下载及非法传播。

上一节：正则表达式运用之 sed工具命令 (/courses/90/labs/781/document)

下一节：挑战：Linux 用户信息编辑 (/courses/90/labs/2788/document)