



实验报告

计算机科学与技术

23020007034-郭长挺

<https://github.com/luoshenzjtywxw/-.git>

1 Shell

1.1 练习

阅读 man ls，然后使用 ls 命令进行如下操作：

所有文件（包括隐藏文件）

文件打印以人类可以理解的格式输出（例如，使用 454M 而不是 454279954）

文件以最近访问顺序排序

以彩色文本显示输出结果

```
穹灵@Atlantis MINGW64 ~/Desktop/aaa
$ ls -a
./  ../  .a.txt.un~  a.txt  a.txt~  b.txt  bbb/  '新建 文本文档.sh'

穹灵@Atlantis MINGW64 ~/Desktop/aaa
$ ls -t
b.txt  a.txt  a.txt~  '新建 文本文档.sh'  bbb/

穹灵@Atlantis MINGW64 ~/Desktop/aaa
$ ls -lh a.txt
-rw-r--r-- 1 穹灵 197609 4.0K Sep  5 19:24 a.txt

穹灵@Atlantis MINGW64 ~/Desktop/aaa
$ ls -color
total 6
-rw-r--r-- 1 穹灵  0 Sep  5 15:46 '新建 文本文档.sh'
drwxr-xr-x 1 穹灵  0 Sep  3 20:23 bbb/
-rw-r--r-- 1 穹灵 15 Sep  5 19:45 b.txt
-rw-r--r-- 1 穹灵 106 Sep  5 17:17 a.txt~
-rw-r--r-- 1 穹灵 4076 Sep  5 19:24 a.txt
```

编写两个 bash 函数 marco 和 polo 执行下面的操作。每当你执行 marco 时，当前的工作目录应当以某种形式保存，当执行 polo 时，无论现在处在什么目录下，都应当 cd 回到当时执行 marco 的目录。为了方便 debug，你可以把代码写在单独的文件 marco.sh 中，并通过 source marco.sh 命令，（重新）加载函数。

这就是 Vim 的界面开始看起来像一个程序语言的时候。Vim 的编辑命令也被称为 “动词”，因为动词可以施动于名词。

i 进入插入模式

但是对于操纵/编辑文本，不单想用退格键完成

O / o 在之上/之下插入行

d{移动命令} 删除 {移动命令}

例如，dw 删除词，d\$ 删除到行尾，d0 删除到行头。

c{移动命令} 改变 {移动命令}

例如，cw 改变词

比如 d{移动命令} 再 i

x 删除字符（等同于 dl）

s 替换字符（等同于 xi）

可视化模式 + 操作

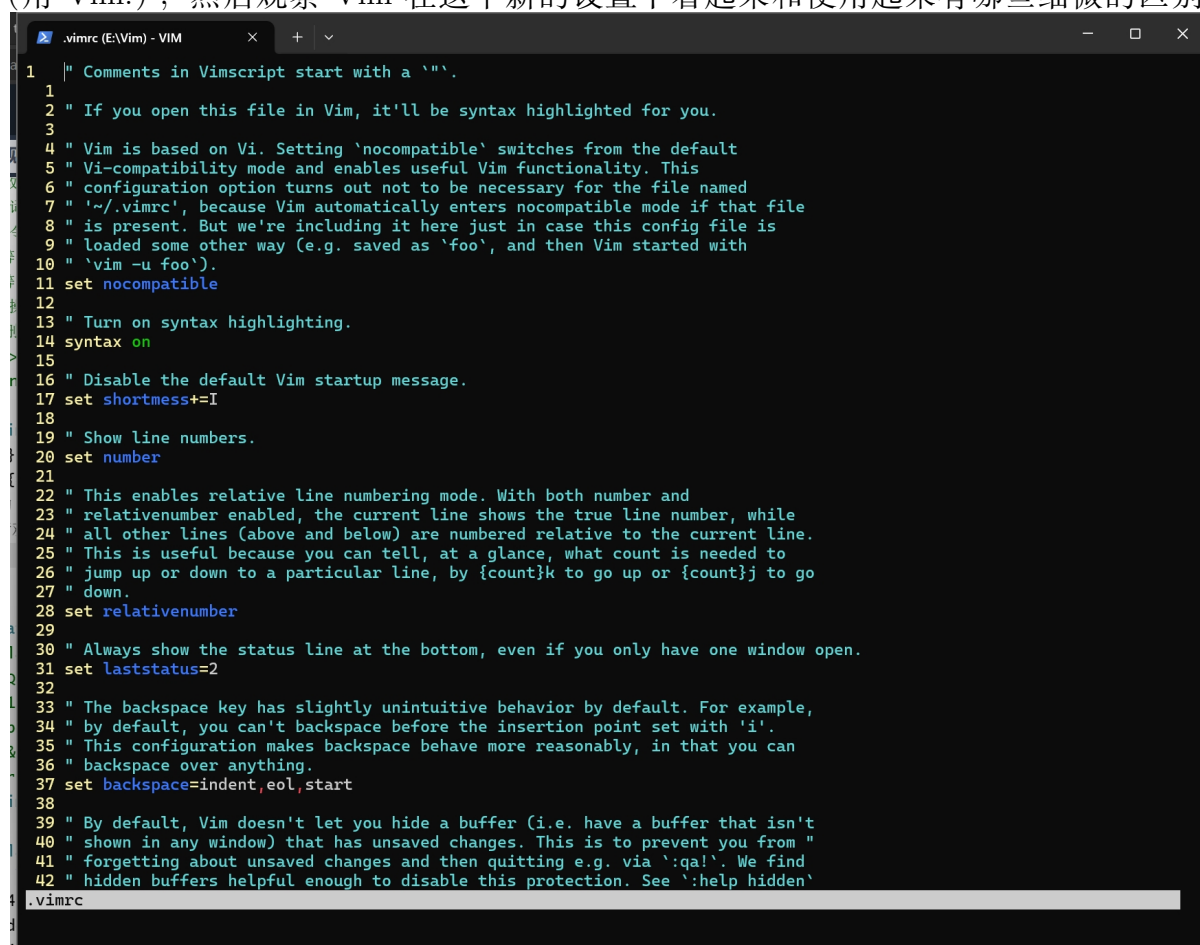
选中文字，d 删除 或者 c 改变

u 撤销，<C-r> 重做

y 复制 / “yank”（其他一些命令比如 d 也会复制）

p 粘贴

下载我们提供的 vimrc，然后把它保存到 ~/.vimrc。通读这个注释详细的文件（用 Vim!），然后观察 Vim 在这个新的设置下看起来和使用起来有哪些细微的区别。



```
1  " Comments in Vimscript start with a '"'.
2  " If you open this file in Vim, it'll be syntax highlighted for you.
3
4  " Vim is based on Vi. Setting 'nocompatible' switches from the default
5  " Vi-compatibility mode and enables useful Vim functionality. This
6  " configuration option turns out not to be necessary for the file named
7  " '~/.vimrc', because Vim automatically enters nocompatible mode if that file
8  " is present. But we're including it here just in case this config file is
9  " loaded some other way (e.g. saved as 'foo', and then Vim started with
10 " 'vim -u foo').
11 set nocompatible
12
13 " Turn on syntax highlighting.
14 syntax on
15
16 " Disable the default Vim startup message.
17 set shortmess+=I
18
19 " Show line numbers.
20 set number
21
22 " This enables relative line numbering mode. With both number and
23 " relativenumber enabled, the current line shows the true line number, while
24 " all other lines (above and below) are numbered relative to the current line.
25 " This is useful because you can tell, at a glance, what count is needed to
26 " jump up or down to a particular line, by {count}k to go up or {count}j to go
27 " down.
28 set relativenumber
29
30 " Always show the status line at the bottom, even if you only have one window open.
31 set laststatus=2
32
33 " The backspace key has slightly unintuitive behavior by default. For example,
34 " by default, you can't backspace before the insertion point set with 'i'.
35 " This configuration makes backspace behave more reasonably, in that you can
36 " backspace over anything.
37 set backspace=indent,eol,start
38
39 " By default, Vim doesn't let you hide a buffer (i.e. have a buffer that isn't
40 " shown in any window) that has unsaved changes. This is to prevent you from "
41 " forgetting about unsaved changes and then quitting e.g. via `:qa!`. We find
42 " hidden buffers helpful enough to disable this protection. See `:help hidden`
.vimrc
```

每一行前方有序号，可以更方便的编辑，语句有不同的颜色，如果错误使用上下左右会有提醒，可以使用鼠标功能。搜索可以不区分大小写

2.2 心得

vim 是一个非常好用的文本编辑器, 可以免去我们操作鼠标, 加快我们的工作效率, 我们也可以为自己的 vim 自定义各种设置, 更加个性化.

3 数据整理

3.1 练习

Exercise 1: Matching Characters

Task	Text	
Match	abcdefg	✓
Match	abcde	✓
Match	abc	✓

[Continue >](#)

Solve the above task to continue on to the next problem, or read the [Solution](#).

Exercise 1½: Matching Digits

Task	Text	
Match	abc123xyz	✓
Match	define "123"	✓
Match	var g = 123;	✓

[Continue >](#)

Solve the above task to continue on to the next problem, or read the [Solution](#).

Exercise 2: Matching With Wildcards

Task	Text	
Match	cat.	✓
Match	896.	✓
Match	?=+.	✓
Skip	abc1	

[Continue >](#)

Solve the above task to continue on to the next problem, or read the [Solution](#).

Exercise 3: Matching Characters

Task	Text	
Match	can	✓
Match	man	✓
Match	fan	✓
Skip	dan	
Skip	ran	
Skip	pan	

[Continue ›](#)

Solve the above task to continue on to the next problem, or read the [Solution](#).

Exercise 4: Excluding Characters

Task	Text	
Match	hog	✓
Match	dog	✓
Skip	bog	

[Continue ›](#)

Solve the above task to continue on to the next problem, or read the [Solution](#).

Exercise 5: Matching Character Ranges

Task	Text	
Match	Ana	✓
Match	Bob	✓
Match	Cpc	✓
Skip	aax	
Skip	bby	
Skip	ccz	

[Continue ›](#)

Solve the above task to continue on to the next problem, or read the [Solution](#).

Exercise 6: Matching Repeated Characters

Task	Text
------	------

Match	wazzzzzup
-------	-----------



Match	wazzzup
-------	---------



Skip	wazup
------	-------

[Continue >](#)

Solve the above task to continue on to the next problem, or read the [Solution](#).

Exercise 7: Matching Repeated Characters

Task	Text
------	------

Match	aaaabcc
-------	---------



Match	aabbbbbc
-------	----------



Match	aacc
-------	------



Skip	a
------	---

[Continue >](#)

Solve the above task to continue on to the next problem, or read the [Solution](#).

Exercise 8: Matching Optional Characters

Task	Text
------	------

Match	1 file found?
-------	---------------



Match	2 files found?
-------	----------------



Match	24 files found?
-------	-----------------



Skip	No files found.
------	-----------------

[Continue >](#)

Solve the above task to continue on to the next problem, or read the [Solution](#).

Exercise 9: Matching Whitespaces

Task	Text
------	------

Match	1. abc
-------	--------



Match	2. abc
-------	--------



Match	3. abc
-------	--------



Skip	4.abc
------	-------

[Continue >](#)

Solve the above task to continue on to the next problem, or read the [Solution](#).

Exercise 10: Matching Lines

Task	Text	
Match	Mission: successful	✓
Skip	Last Mission: unsuccessful	
Skip	Next Mission: successful upon capture of target	
<input type="text" value="^Mission:\s*successful\$"/>		Continue >

Solve the above task to continue on to the next problem, or read the [Solution](#).

Exercise 11: Matching Groups

Task	Text	Capture Groups
Capture	file_record_transcript.pdf	<input type="text" value="file_record_transcript"/> ✓
Capture	file_07241999.pdf	<input type="text" value="file_07241999"/> ✓
Skip	testfile_fake.pdf.tmp	
<input type="text" value="^(\\w+)\\.pdf\$"/>		Continue >

Solve the above task to continue on to the next problem, or read the [Solution](#).

3.2 心得

正则表达式可以极大提高我们处理数据的能力, 具有强大的搜索能力, 对于编程、数据分析和系统管理等领域都具有重要的意义