

22章 文本处理

作者 bluetea

网站<https://github.com/bluetea>

1. 下载文件

```
1 require "open-uri"
2
3 url = "http://www.cnbeta.com"
4 filename = "cnbeta.html"
5 File.open(filename, "w") do |f|
6   html = open(url).read
7   p html.encoding
8   f.write html
9 end
```

```
getpage.rb
1 htmlfile = "cnbeta.html"
2 textfile = "cnbeta.txt"
3
4 html = File.read(htmlfile)
5 File.open(textfile, "w") do |f|
6   in_header = true #确认当匹配h1标签后, 不会再进行标签匹配, 增加效率
7   html.each_line do |line|
8     if in_header && /<title>cnBeta.COM_中文业界资讯站/ !~ line #!~如果匹配不到返回true|
9       next
10    else
11      in_header = false
12    end
13    break if /<footer class/ =~ line #如果匹配到footer则不再读取后面的东西
14    f.write line #将非h1之上的的标签写入到cnbeta.txt文件内
15  end
16 end
17
```

2. 只留正文部分

3. 删除标签

```
1 require "cgi/util"
2 htmlfile = "cnbeta.html"
3 textfile = "cnbeta.txt"
4
5 html = File.read(htmlfile)
6 File.open(textfile, "w") do |f|
7   in_header = true #确认当匹配h1标签后, 不会再进行标签匹配, 增加效率
8   html.each_line do |line|
9     if in_header && /<title>cnBeta.COM_中文业界资讯站/ !~ line #!~如果匹配不到返回true
10      next
11    else
12      in_header = false
13    end
14    break if /<footer class/ =~ line #如果匹配到footer则不再读取后面的东西
15    line.gsub!(/<[^\>]+\>/, '') #删除所有的标签
16    esc_line = CGI.unescapeHTML(line) #这个方法可以将html标签的&amp, &lt等字符转义为&, <等
17    f.write esc_line
18  end
19 end
20
```

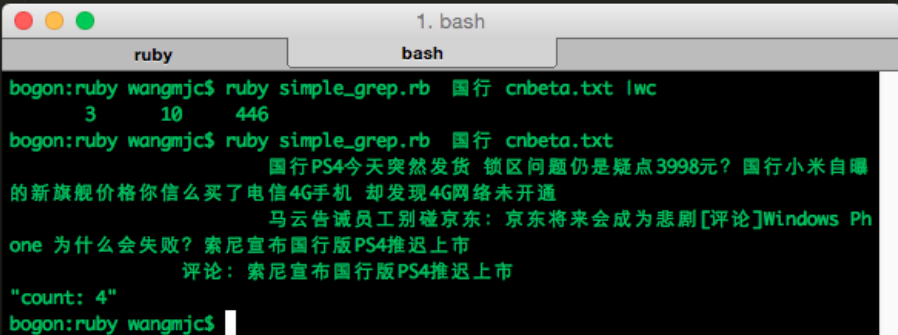
任务2 扩展simple_grep.rb

计算国行在文本中的出现次数

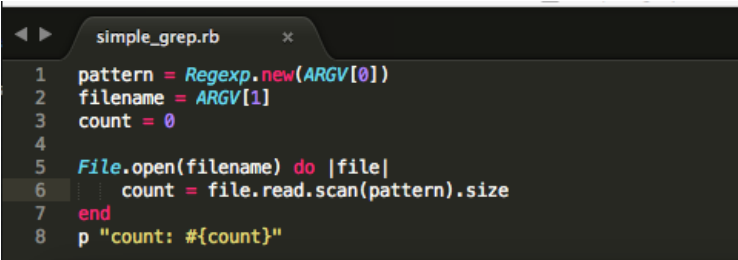
但是这种如果一行中出现两次, 那么就无法准确计算, 改造如下:

或者这样

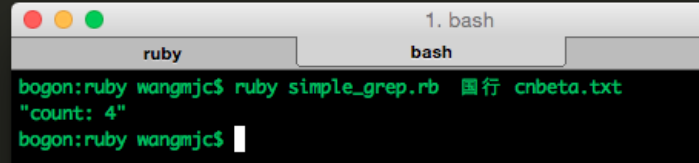
```
1 pattern = Regexp.new(ARGV[0])
2 filename = ARGV[1]
3 count = 0
4
5 File.open(filename) do |file|
6   file.each_line do |line|
7     if pattern =~ line
8       count += line.scan(pattern).size
9       print line
10    end
11  end
12
13
```



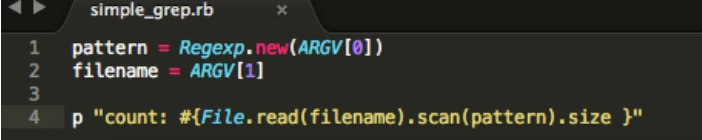
但是如果仅仅是为了计算出出现次数的话，可以更简单的改造



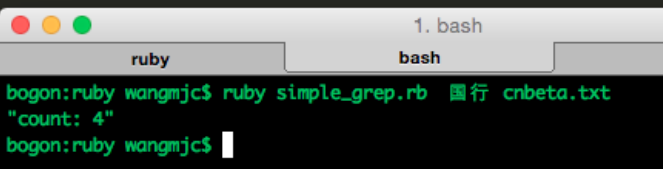
```
1 pattern = Regexp.new(ARGV[0])
2 filename = ARGV[1]
3 count = 0
4
5 File.open(filename) do |file|
6   count = file.read.scan(pattern).size
7 end
8 p "count: #{count}"
```



或者更简单



```
1 pattern = Regexp.new(ARGV[0])
2 filename = ARGV[1]
3
4 p "count: #{File.read(filename).scan(pattern).size}"
```



继续改造，让打印包含国行这行的时候，将国行以一种形式高亮显示

```
1 pattern = Regexp.new(ARGV[0])
2 filename = ARGV[1]
3 count = 0
4
5 File.open(filename) do |file|
6   file.each_line do |line|
7     if pattern =~ line
8       count += line.scan(pattern).size
9       print line.gsub(pattern){|i| "<<#{i}>>"}
10    end
11  end
12 end
13 p "count: #{count}"
```



```
1. bash
ruby      bash
bogon:ruby wangmjc$ ruby simple_grep.rb 国行 cnbeta.txt
<<国行>>PS4今天突然发货 锁区问题仍是疑点3998元? <<国行>>小米
自曝的新旗舰价格你信么买了电信4G手机 却发现4G网络未开通
马云告诫员工别碰京东: 京东将来会成为悲剧[评论]Windows Phone
为什么会失败? 索尼宣布<<国行>>版PS4推迟上市
评论: 索尼宣布<<国行>>版PS4推迟上市
"count: 4"
bogon:ruby wangmjc$
```

继续改造，只显示匹配字符的前后个10个字符