

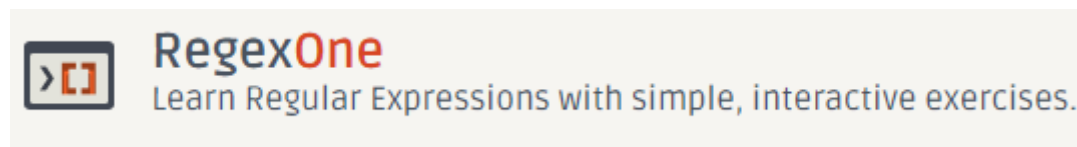
HW4Solution

Author:金之航

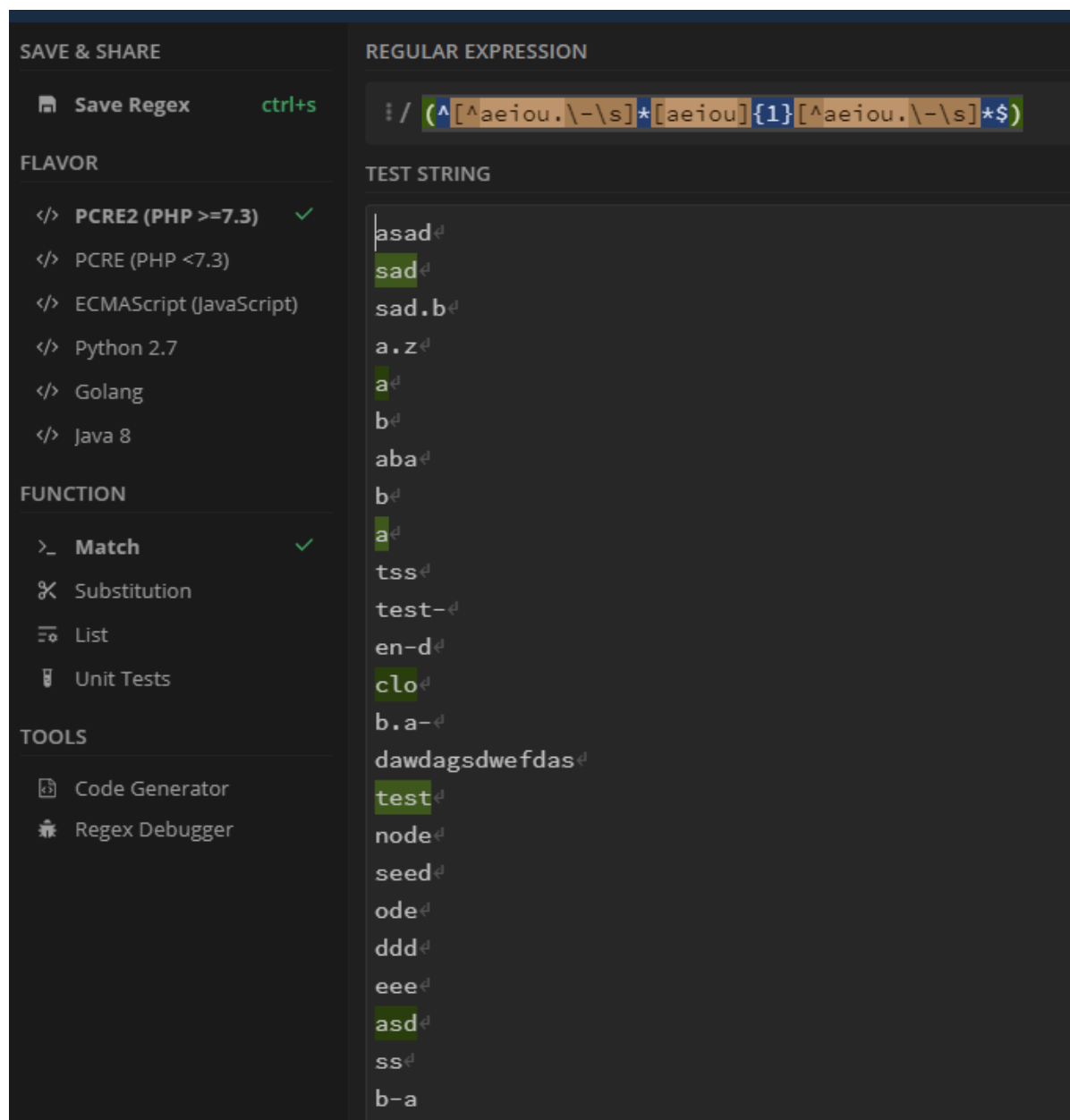
StuID:2183411101

Problem

1.Regex One



regex101 test:

The screenshot shows the regex101 website interface. On the left side, there is a sidebar with several sections: "SAVE & SHARE" with a "Save Regex" button and "ctrl+s" shortcut; "FLAVOR" with a list of options including "PCRE2 (PHP >=7.3)" which is selected and marked with a green checkmark; "FUNCTION" with a list of options including "Match" which is selected and marked with a green checkmark; and "TOOLS" with links to "Code Generator" and "Regex Debugger". The main area on the right is titled "REGULAR EXPRESSION" and contains the input field with the regex: `/([aeiou.\-\\s]*[aeiou]{1}[aeiou.\-\\s]*$)`. Below this is the "TEST STRING" section, which lists various test strings: "asad", "sad", "sad.b", "a.z", "a", "b", "aba", "b", "a", "tss", "test-", "en-d", "clo", "b.a-", "dawdagsdwefdas", "test", "node", "seed", "ode", "ddd", "eee", "asd", "ss", and "b-a". The strings "sad", "a", "a", "clo", "test", "asd", and "ss" are highlighted in green, indicating they match the regex.

2.wordcount

```
xpanse:missing-semester (master %) | echo 'bba' | sed 's/[ab]//'
```

```
xpanse:missing-semester (master %) | echo 'bba' | sed 's/[ab]//g'
```

```
xpanse:missing-semester (master %) | echo 'bbac' | sed 's/[ab]//g'
```

```
xpanse:missing-semester (master %) | echo 'bcbzac' | sed 's/[ab]//g'
```

```
xpanse:missing-semester (master %) | echo 'abcaba' | sed 's/(ab)*//g'
```

```
xpanse:missing-semester (master %) | echo 'abcaba' | sed -E 's/(ab)*//g'
```

```
xpanse:missing-semester (master %) | echo 'abcaba' | sed 's/\\(ab\\+)//g'
```

```
xpanse:missing-semester (master %) | echo 'abcaba' | sed -E 's/(ab)*//g'
```

```
xpanse:missing-semester (master %) | echo 'abcababc' | sed -E 's/(ab|bc)*//g'
```

```
xpanse:missing-semester (master %) | echo 'abcabbc' | sed -E 's/(ab|bc)*//g'
```

```
1 cat /usr/share/dict/words | tr "[:upper:]" "[:lower:]" | grep -E "
  (^([aeiou]\-\\s)*[aeiou]{1}([aeiou]\-\\s)*$)" | sed -E "s/.*([a-z]{1})$/\1/" |
  sort | uniq -c | sort -r | grep -E "[aeiou]{1}$" | head -5
2
3 cat /usr/share/dict/words | tr "[:upper:]" "[:lower:]" | grep -E "
  (^([aeiou]\-\\s)*[aeiou]{1}([aeiou]\-\\s)*$)" | sed -E "s/.*([a-z]{1})$/\1/" |
  sort | uniq -c | sort -n | grep -E "[aeiou]{1}$" | head -5
4
5 tr "[:upper:]" "[:lower:]" #忽略大小写
6 grep -E " (^([aeiou]\-\\s)*[aeiou]{1}([aeiou]\-\\s)*$)" #获取想要的数
据，仅含一个元音字母的单词
7 sed -E "s/.*([a-z]{1})$/\1/" #利用脚本来处理文本文件，只留下元音字母
8 sort | uniq -c #用uniq -c统计元音字符出现次数
9 sort -n | grep -E "[aeiou]{1}$" | head -5 #按大小排列
```

output:

```
→ ~ cat /usr/share/dict/words | tr "[:upper:]" "[:lower:]" | grep -E "([^\aeiou\.\-\s]*[aeiou]{1}[^\aeiou\.\-\s]*$)" | sed -E "s/.*([a-z]{1})$/\1/" | sort | uniq -c | sort -r | grep -E "[aeiou]{1}$" | head -5
```

```
324 e
302 a
190 o
183 i
116 u
```

3.sed

```
1 sed s/REGEX/SUBSTITUTION/ input.txt > input.txt 表达式中后一个 input.txt会首先被
   清空，而且是发生在前的。所以前面一个input.txt在还没有被 sed 处理时已经为空了。在使用正则处
   理文件前最好是首先备份文件。
2
3 sed -i.bak s/REGEX/SUBSTITUTION/ input.txt > input.txt
4 可以自动创建一个后缀为 .bak 的备份文件。
```

4.curl <http://dean.xjtu.edu.cn>

```
1 curl http://dean.xjtu.edu.cn | grep -Eo "(<[a-z]+)( ?.*?)*?>" | sed -E
  "s/(<\w+?)( ?.*?)*?>/\1>/" | sort | uniq -c | sort -n
2
3 curl http://dean.xjtu.edu.cn | grep -Eo "(<[a-z]+)( ?.*?)*?>" | sed -E
  "s/(<\w+?)( ?.*?)*?>/\1>/" | sort | uniq -c | sort -n | awk 'BEGIN { row = 0
    } $2 ~ /^<d.*v>$/ { row +=1 } END { print row }' #随便写的
```

```
→ ~ curl http://dean.xjtu.edu.cn | grep -Eo "(<[a-z]+)( ?.*?)*?>" | sed -E "s/(<\w+?)( ?.*?)*?>/\1>/" |
  sort | uniq -c | sort -n
% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
           Dload  Upload   Total      Spent    Left     Speed
100 42494  100 42494    0     0  2202k      0 --:--:-- --:--:-- --:--:-- 2305k
 1 <form>
 1 <p>
 2 <link>
 3 <input>
 3 <select>
 6 <table>
 8 <div>
 9 <span>
14 <img>
15 <td>
15 <tr>
21 <script>
44 <li>
49 <option>
53 <a>
```

```
1 curl http://dean.xjtu.edu.cn | grep -Eo "(<[a-z]+)( ?.*?)*?>" | sed -E
  "s/(<\w+?)( ?.*?)*?>/\1>/" | sort | uniq -c | sort -n | awk 'BEGIN { row = 0
    } { print $1 } END { print row } ' | paste -sd+ | bc -l #求行数, 可用wc代替
2
3 curl http://dean.xjtu.edu.cn | grep -Eo "(<[a-z]+)( ?.*?)*?>" | sed -E
  "s/(<\w+?)( ?.*?)*?>/\1>/" | sort | uniq -c | sort -n | paste -sd\ | awk
  '{print $8}' #求中位数
4
5 curl http://dean.xjtu.edu.cn | grep -Eo "(<[a-z]+)( ?.*?)*?>" | sed -E
  "s/(<\w+?)( ?.*?)*?>/\1>/" | sort | uniq -c | sort -n | awk 'BEGIN { row = 0
    } { print $1 } END { print row } ' | paste -sd\ | awk '{print ($7+$8)/2}' #
  求平均
6 #...
7
8 sudo yum -y install R #安装R语言
9 curl http://dean.xjtu.edu.cn | grep -Eo "(<[a-z]+)( ?.*?)*?>" | sed -E
  "s/(<\w+?)( ?.*?)*?>/\1>/" | sort | uniq -c | sort -n | awk '{ print $1 }' |
  R --slave -e 'x<-scan(file="stdin", quiet=TRUE);summary(x)'
10 #or
11 curl http://dean.xjtu.edu.cn | grep -Eo "(<[a-z]+)( ?.*?)*?>" | sed -E
  "s/(<\w+?)( ?.*?)*?>/\1>/" | sort | uniq -c | sort -n | awk '{ print $1 }' |
  R --slave -e 'x<-scan(file="stdin",
    quiet=TRUE);min(x);max(x);mean(x);median(x);'
```

```
→ ~ curl http://dean.xjtu.edu.cn | grep -Eo "(<[a-z]+)( ?.*?)*?>" | sed -E "s/(<\w+?)( ?.*?)*?>/\1>/" |
  sort | uniq -c | sort -n | awk '{ print $1 }' | R --slave -e 'x<-scan(file="stdin", quiet=TRUE);summary (
  x)'
```

% Total	% Received	% Xferd	Average Speed	Time	Time	Time	Current	
			Dload	Upload	Total	Spent	Left	Speed
100	42494	100	42494	0	0	1582k	0	--:--:-- --:--:-- --:--:-- 1596k
Min.	1st Qu.	Median	Mean	3rd Qu.	Max.			
1.00	3.00	9.00	16.27	18.00	53.00			

```

→ ~ curl http://dean.xjtu.edu.cn | grep -Eo "<[a-z]+( ?.*?)*?>" | sed -E "s/(<w+?)( ?.*?)*?>/\1>/" |
sort | uniq -c | sort -n | awk '{ print $1 }' | R --slave -e 'x<-scan(file="stdin", quiet=TRUE);min(x);m
ax(x);mean(x);median(x);'
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left   Speed
100 42494  100 42494    0     0  2628k      0 --:--:-- --:--:-- --:--:-- 2766k
[1] 1
[1] 53
[1] 16.26667
[1] 9

```