

## 浙江大学城市学院实验报告

课程名称 跨平台脚本开发技术

实验项目名称 实验八 错误处理与正则表达式

学生姓名 吴成洋 专业班级 软件工程 1404 学号 31401417

实验成绩          指导老师（签名）                      日期                 

注意：

- 务请保存好各自的源代码，以备后用。
- 请把作业保存为 pdf 上传到 BB 平台，请务必在截止日期前提交。

### 实验目的：

掌握 JS 中的错误处理与正则表达式。

### 实验内容：

1. 学习 JavaScript 调试：把以下代码存为 debug.js，用不同的调试方式学习如何调试。

```
function numberToString(n, base) {  
  var result = "", sign = "";  
  if (n < 0) {  
    sign = "-";  
    n = -n;  
  }  
  do {  
    result = String(n % base) + result;  
    n /= base;  
  } while (n > 0);  
  return sign + result;  
}  
  
console.log(numberToString(13, 10));  
  
// → 1.5e-3231.3e-3221.3e-3211.3e-3201.3e-3191.3e-3181.3...
```

- 1) 用 console.log
- 2) 在 node.js 和 chromedevtools 中设置断点调试;
- 3) 添加 debugger 语句, 在 node.js 和 chromedevtools 中观察。

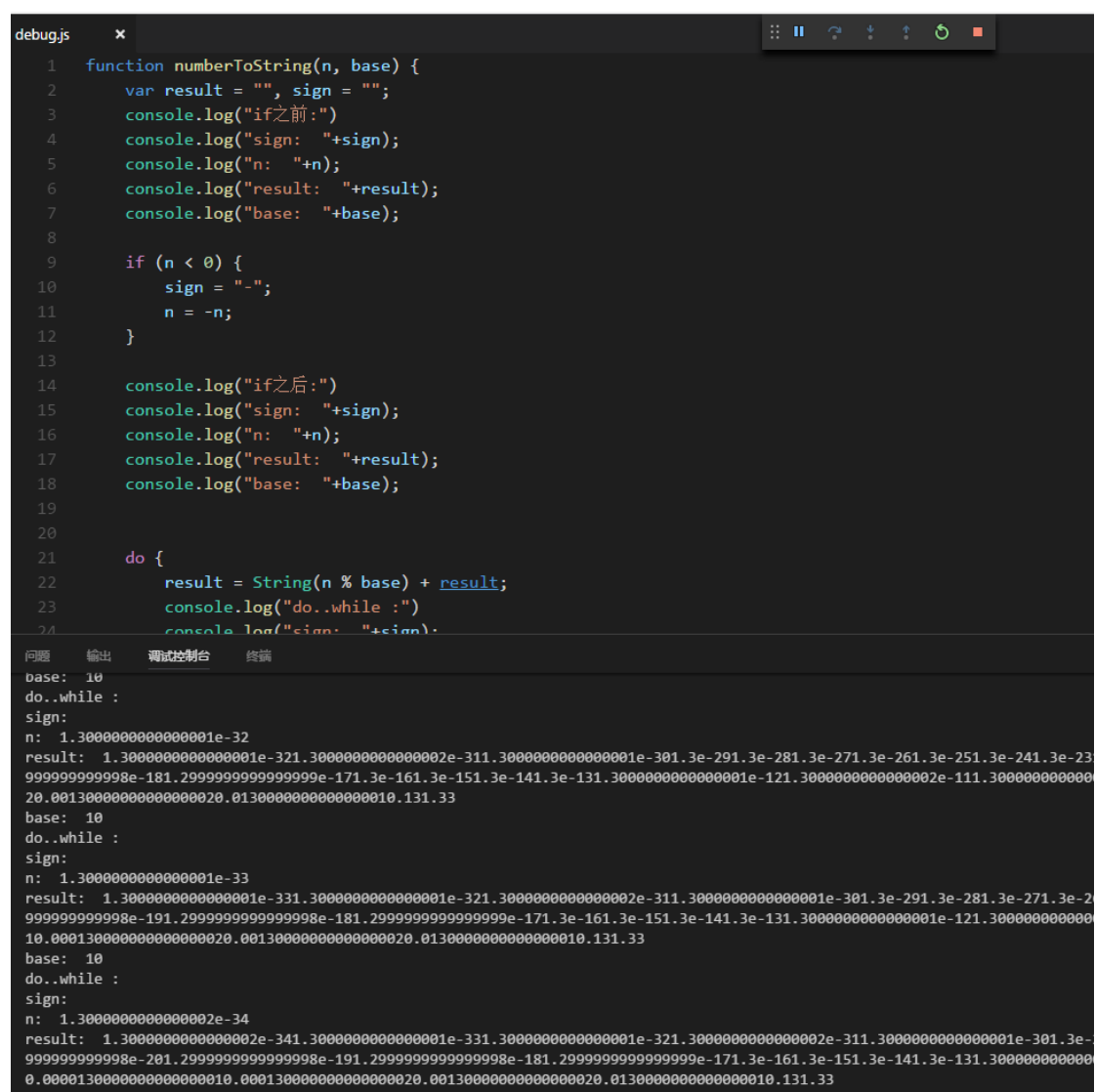
2. 教材 P115 习题 8.11.1、8.11.2

3. 教材 P133 习题 9.20.1、9.20.2、9.20.3

## 实验步骤:

1、

1.1、



```
debugjs x
1 function numberToString(n, base) {
2   var result = "", sign = "";
3   console.log("if之前:")
4   console.log("sign: "+sign);
5   console.log("n: "+n);
6   console.log("result: "+result);
7   console.log("base: "+base);
8
9   if (n < 0) {
10    sign = "-";
11    n = -n;
12  }
13
14  console.log("if之后:")
15  console.log("sign: "+sign);
16  console.log("n: "+n);
17  console.log("result: "+result);
18  console.log("base: "+base);
19
20
21  do {
22    result = String(n % base) + result;
23    console.log("do..while :")
24    console.log("sign: "+sign);
25  } while (n > 0);
26
27  return result;
28 }
```

问题 输出 调试控制台 终端

base: 10  
do..while :  
sign:  
n: 1.3000000000000001e-32  
result: 1.3000000000000001e-321.3000000000000002e-311.3000000000000001e-301.3e-291.3e-281.3e-271.3e-261.3e-251.3e-241.3e-23  
99999999998e-181.2999999999999999e-171.3e-161.3e-151.3e-141.3e-131.3000000000000001e-121.3000000000000002e-111.300000000000  
20.001300000000000020.013000000000000010.131.33  
base: 10  
do..while :  
sign:  
n: 1.3000000000000001e-33  
result: 1.3000000000000001e-331.3000000000000001e-321.3000000000000002e-311.3000000000000001e-301.3e-291.3e-281.3e-271.3e-2  
99999999998e-191.2999999999999999e-181.2999999999999999e-171.3e-161.3e-151.3e-141.3e-131.3000000000000001e-121.300000000000  
10.000130000000000020.001300000000000020.013000000000000010.131.33  
base: 10  
do..while :  
sign:  
n: 1.3000000000000002e-34  
result: 1.3000000000000002e-341.3000000000000001e-331.3000000000000001e-321.3000000000000002e-311.3000000000000001e-301.3e-  
99999999998e-201.2999999999999999e-191.2999999999999999e-181.2999999999999999e-171.3e-161.3e-151.3e-141.3e-131.300000000000  
0.00001300000000000010.000130000000000020.001300000000000020.013000000000000010.131.33

代码如下:

```
function numberToString(n, base) {  
  var result = "", sign = "";
```

```
console.log("if 之前:")
console.log("sign: "+sign);
console.log("n: "+n);
console.log("result: "+result);
console.log("base: "+base);

if (n < 0) {
    sign = "-";
    n = -n;
}

console.log("if 之后:")
console.log("sign: "+sign);
console.log("n: "+n);
console.log("result: "+result);
console.log("base: "+base);

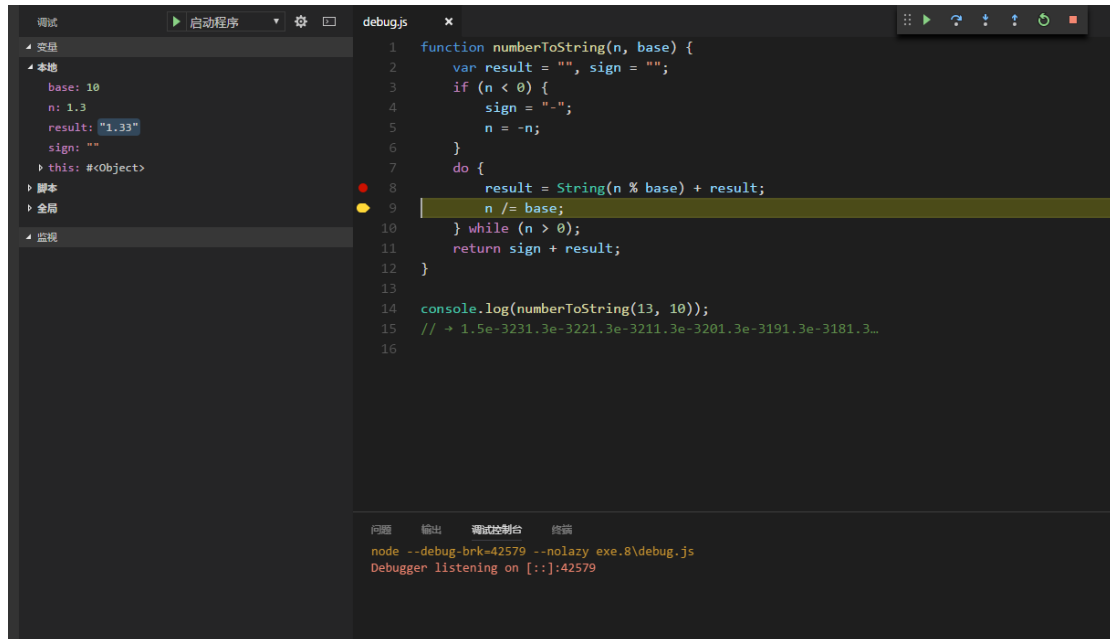
do {
    result = String(n % base) + result;
    console.log("do..while :")
    console.log("sign: "+sign);
    console.log("n: "+n);
    console.log("result: "+result);
    console.log("base: "+base);
    n /= base;
} while (n > 0);

console.log("do..while 之后:")
console.log("sign: "+sign);
console.log("n: "+n);
console.log("result: "+result);
console.log("base: "+base);

return sign + result;
}

console.log(numberToString(13, 10));
// → 1.5e-3231.3e-3221.3e-3211.3e-3201.3e-3191.3e-3181.3...
```

## 1.2、



## 1.3、

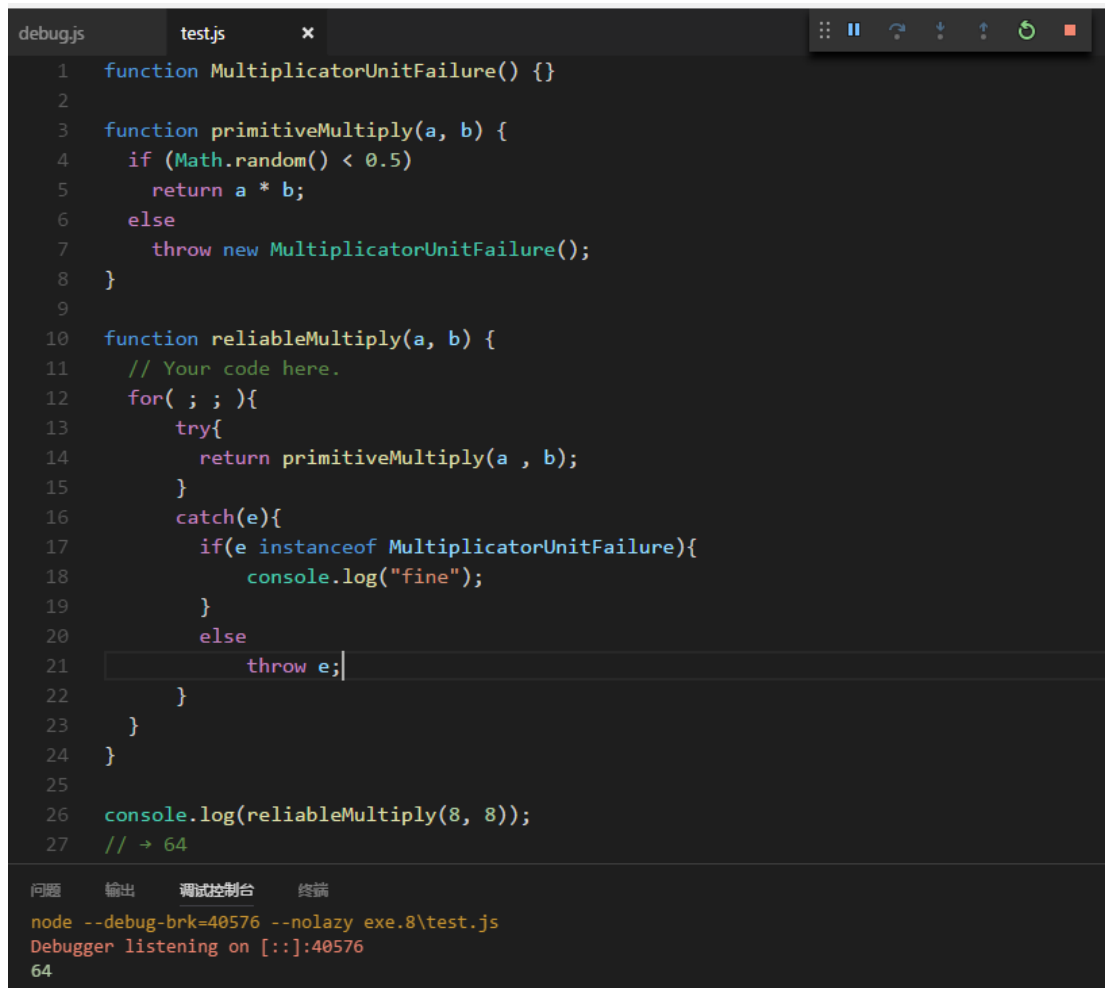
```
debug.js  x
1  function numberToString(n, base) {
2      var result = "", sign = "";
3      if (n < 0) {
4          sign = "-";
5          n = -n;
6      }
7      do {
8          result = String(n % base) + result;
9          debugger;
10         n /= base;
11     } while (n > 0);
12     return sign + result;
13 }
14
15 console.log(numberToString(13, 10));
16 // → 1.5e-3231.3e-3221.3e-3211.3e-3201.3e-3191.3e-3181.3...
17

问题  输出  调试控制台  终端

16 // → 1.5e-3231.3e-3221.3e-3211.3e-3201.3e-3191.3e-3181.3...
17
debug> help
Commands: run (r), cont (c), next (n), step (s), out (o), backtrace (bt), setBreakpoint (sb), clearBreakpoint (cb),
watch, unwatch, watchers, repl, exec, restart, kill, list, scripts, breakOnException, breakpoints, version
debug> sb
[Function: bound ]
debug> n
break in D:\CYann_Work\JS\exe.8\debug.js:9
   7   do {
   8       result = String(n % base) + result;
>  9       debugger;
   9       n /= base;
  10   } while (n > 0);
debug> n
break in D:\CYann_Work\JS\exe.8\debug.js:10
   8       result = String(n % base) + result;
   9       debugger;
>10       n /= base;
  11   } while (n > 0);
debug> n
break in D:\CYann_Work\JS\exe.8\debug.js:11
   9       debugger;
  10       n /= base;
>11   } while (n > 0);
  12   return sign + result;
  13 }
debug> repl
Press Ctrl + C to leave debug repl
> result
'3'
>
```

2、

2.1、



```
1 function MultiplicatorUnitFailure() {}
2
3 function primitiveMultiply(a, b) {
4   if (Math.random() < 0.5)
5     return a * b;
6   else
7     throw new MultiplicatorUnitFailure();
8 }
9
10 function reliableMultiply(a, b) {
11   // Your code here.
12   for( ; ; ){
13     try{
14       return primitiveMultiply(a , b);
15     }
16     catch(e){
17       if(e instanceof MultiplicatorUnitFailure){
18         console.log("fine");
19       }
20       else
21         throw e;
22     }
23   }
24 }
25
26 console.log(reliableMultiply(8, 8));
27 // → 64
```

问题 输出 调试控制台 终端

```
node --debug-brk=40576 --nolazy exe.8\test.js
Debugger listening on [::]:40576
64
```

## 2.2、



```
debug.js  test.js  test2.js  x  :: || ↶ ↷ ↻
```

```
19  var locked = box.locked;
20  if(!locked)
21      return body();
22
23  box.unlock();
24  try{
25      return body();
26  }finally{
27      box.lock();
28  }
29 }
30
31 withBoxUnlocked(function() {
32     box.content.push("gold piece");
33 });
34
35 try {
36     withBoxUnlocked(function() {
37         throw new Error("Pirates on the horizon! Abort!");
38     });
39 } catch (e) {
40     console.log("Error raised:", e);
41 }
42 console.log(box.locked);
43 // → true
```

问题 输出 调试控制台 终端

```
node --debug-brk=18594 --nolazy exe.8\test2.js
Debugger listening on [::]:18594
Error raised: Error: Pirates on the horizon! Abort!
    at d:\CYann_Work\JS\exe.8\test2.js:37:11
    at withBoxUnlocked (d:\CYann_Work\JS\exe.8\test2.js:25:14)
    at Object.<anonymous> (d:\CYann_Work\JS\exe.8\test2.js:36:3)
    at Module._compile (module.js:570:32)
    at Object.Module._extensions..js (module.js:579:10)
    at Module.load (module.js:487:32)
    at tryModuleLoad (module.js:446:12)
    at Function.Module._load (module.js:438:3)
    at Timeout.Module.runMain [as _onTimeout] (module.js:604:10)
    at ontimeout (timers.js:365:14)
true
```

代码如下：

```
var box = {
  locked:true ,
  unlock:function(){
    this.locked = false;
  },
  lock:function(){
    this.locked = true;
  } ,
  _content:[] ,
  get content(){
```

```
        if(this.locked)
            throw new Error ("locked");
        return this._content;
    }
};

function withBoxUnlocked(body) {
    // Your code here.
    var locked = box.locked;
    if(!locked)
        return body();

    box.unlock();
    try{
        return body();
    }finally{
        box.lock();
    }
}

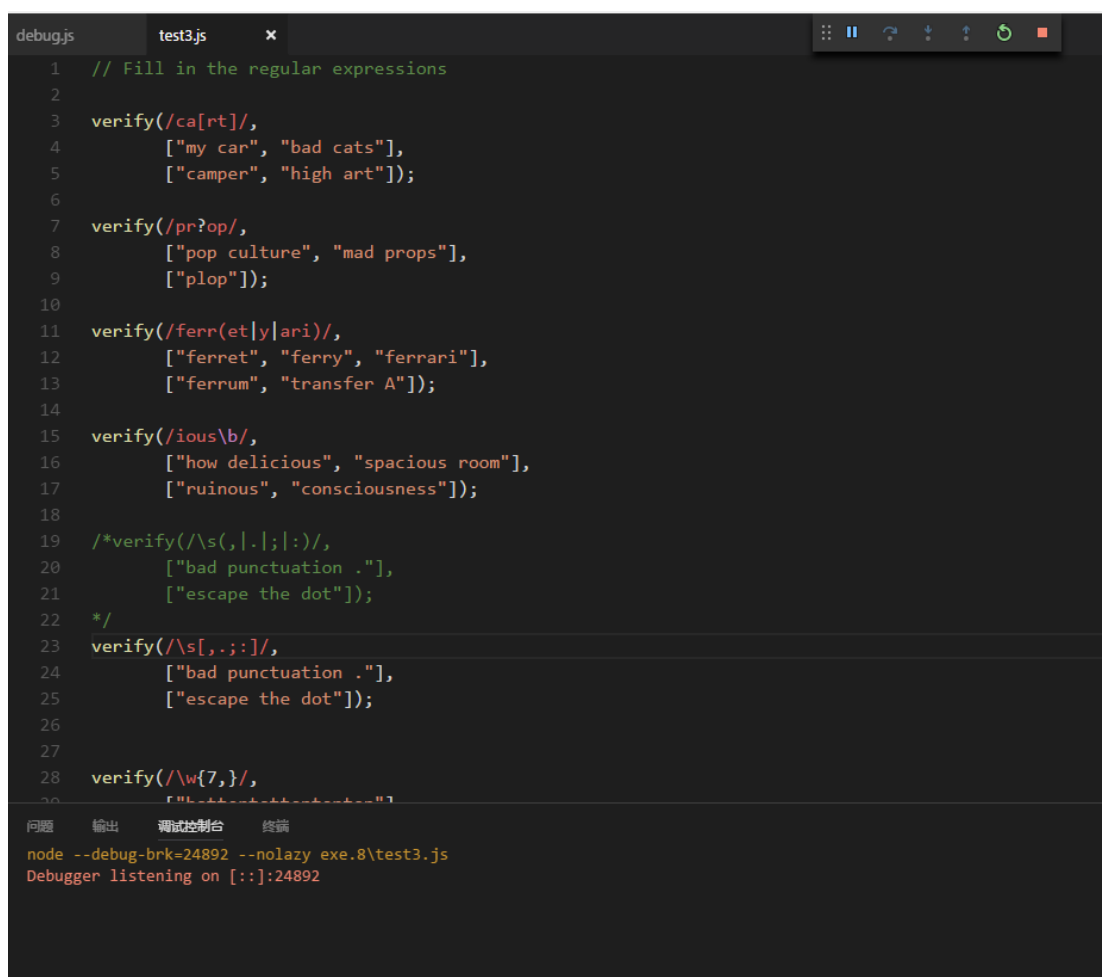
withBoxUnlocked(function() {
    box.content.push("gold piece");
});

try {
    withBoxUnlocked(function() {
        throw new Error("Pirates on the horizon! Abort!");
    });
} catch (e) {
    console.log("Error raised:", e);
}
console.log(box.locked);
// → true
```

3、

3.1、





```
1 // Fill in the regular expressions
2
3 verify(/ca[rt]/,
4     ["my car", "bad cats"],
5     ["camper", "high art"]);
6
7 verify(/pr?op/,
8     ["pop culture", "mad props"],
9     ["plop"]);
10
11 verify(/ferr(et|y|ari)/,
12     ["ferret", "ferry", "ferrari"],
13     ["ferrum", "transfer A"]);
14
15 verify(/ious\b/,
16     ["how delicious", "spacious room"],
17     ["ruinous", "consciousness"]);
18
19 /*verify(/\s(,|.|;|:)/,
20     ["bad punctuation ."],
21     ["escape the dot"]);
22 */
23 verify(/\s[,.;:]/,
24     ["bad punctuation ."],
25     ["escape the dot"]);
26
27
28 verify(/\w{7,}/,
```

问题 输出 调试控制台 终端

node --debug-brk=24892 --nolazy exe.8\test3.js  
Debugger listening on [::]:24892

代码如下：

```
// Fill in the regular expressions

verify(/ca[rt]/,
    ["my car", "bad cats"],
    ["camper", "high art"]);

verify(/pr?op/,
    ["pop culture", "mad props"],
    ["plop"]);

verify(/ferr(et|y|ari)/,
    ["ferret", "ferry", "ferrari"],
    ["ferrum", "transfer A"]);

verify(/ious\b/,
    ["how delicious", "spacious room"],
    ["ruinous", "consciousness"]);
```

```
/*verify(/\s(,|.|;|:)/,
    ["bad punctuation ."],
    ["escape the dot"]);
*/
verify(/\s[,.+:]/,
    ["bad punctuation ."],
    ["escape the dot"]);

verify(/\w{7,}/,
    ["hottentottententen"],
    ["no", "hotten totten tenten"]);

verify(/\b[a-df-z]+\b/i,
    ["red platypus", "wobbling nest"],
    ["earth bed", "learning ape"]);

function verify(regex, yes, no) {
    // Ignore unfinished exercises
    if (regex.source == "...") return;
    yes.forEach(function(s) {
        if (!regex.test(s))
            console.log("Failure to match '" + s + "'");
    });
    no.forEach(function(s) {
        if (regex.test(s))
            console.log("Unexpected match for '" + s + "'");
    });
}
```

### 3.2、

```
test4.js  x
1  var text = "'I'm the cook,' he said, 'it's my job.'";
2  // Change this call.
3  console.log(text.replace(/(^|\w)'|'(\w|$)/i, '$1"$2'));
4  // → "I'm the cook," he said, "it's my job."
5

问题  输出  调试控制台  终端
node --debug-brk=23640 --nolazy exe.8\test4.js
Debugger listening on [::]:23640
"I'm the cook," he said, "it's my job."
```

### 3.3、

```
test4.js  test5.js  x
1  // Fill in this regular expression.
2  var number = /^(+|-)(\d+(\.d*)?|\.\d+)([eE](+|-)\d+)?$/;
3
4  // Tests:
5  ["1", "-1", "+15", "1.55", ".5", "5.", "1.3e2", "1E-4",
6   "1e+12"].forEach(function(s) {
7     if (!number.test(s))
8       console.log("Failed to match '" + s + "'");
9   });
10 ["1a", "+-1", "1.2.3", "1+1", "1e4.5", ".5.", "1f5",
11  "."].forEach(function(s) {
12     if (number.test(s))
13       console.log("Incorrectly accepted '" + s + "'");
14   });

问题  输出  调试控制台  终端
node --debug-brk=44348 --nolazy exe.8\test5.js
Debugger listening on [::]:44348
```