Computer Science G12

Term 2, Test 2. Wed Jan 10 2018

Name :

Note: Write your full name in capitals.

Submission

Write your code in a text file (extension `txt`) and submitted as attachment via email.

Problems

Note: The reference to the `print` function is the one available in the HTML Live Editor, which you can use.

1. Implement a counter object in Javascript such that the following code prints `32`. **Constraints**: The only instance properties are the methods `inc()` and getCount()‘.

   ```
   var i = new Counter(30)
   i.inc()
   i.inc()
   print("i:"+i)
   ```

2. You are provided a `complex.js` script file and a `complex-test.html` HTML test file. You’ll need to modify them to satisfy the constraints stated below.

   Both files can be downloaded from http://msantos.sdf.org/G12/Term2 and are also attached in the appendix section below in this document.

   Implement the necessary code so that the test code runs as expected.

Appendix

Complex HTML template file: complex-test.html

```
<!doctype html>
<html>
  <!-- HTML template page: complex-test.html
       http://msantos.sdf.org/G12/Term2/complex-test.html
  -->
```

```
  <head>
      <title>Complex test bed</title>
      <meta charset="utf-8">
      <script>
        //Helper function
        function print(x){
          document.body.innerHTML += x + "<br>";
        }
      </script>
      <style>
        body { font-family: monospace ; }
      </style>
  </head>

  <body>


<script src="file://PATH/complex.js"></script>
    <script>

var z = new Complex([2,2],"z")
var v = new Complex([-2,1],"v")
var w = new Complex([-7,4],"w")
var i = new Complex([0,1],"v")
print(z) //prints z(2,2)
print(v) //  v(-2,1)
print(w) //  w(-7,4)
print(i) //  i

var zt = z.t()   // z†(2,-2)
var vt = v.t()   // v†(-2,-1)
var it = i.t()   // i†

print( z.norm() )    // 2 sqrt(2) ~ 2.8284
print( v.norm() )    // sqrt(5) ~ 2.2361
print( i.norm() )    // 1

var z_v = z.add(v)   // (z+v)
var wxz_v = w.mult(z.add(v)) // w(z+v)

print( z.mult(i) )   //  zi(-2,2)
print( i.mult(i) )   //  -1

print( w.phase() )   // angle with x-axis
    </script>
  </body>
```

```
</html>
```

where `PATH` depends on wehter you run on Windows or Mac:

a. **Mac**: `PATH=/Users/'your-use-name'/Desktop`
b. **Windows**: `PATH=/C:/Users/'your-use-name'/Desktop`

and where the script file `complex.js`, which can be downloaded from http://msantos.sdf.org/G12/Term2/, contains the complex template library that can be found in the appendix below.

Complex Template Library: complex.js

```
//Complex Numbers Library

/*The complex number constructor

Input:
        vec:: array of 2 numbers
    name :: string

"Output": (not really a return value, mind you!)
        Object with properties:
            x, y :: real & imaginary components
        name :: string, label of our complex number
        If the input vec variable is [0,1], the
                name **must** be `i`
            If not given any input name, it needs to get a default one of `z`
*/
function Complex(){
}

Complex.prototype.toString = function(){

}

Complex.prototype.mult = function(B){

}
```