

类型判断问题:

用三种方式, 甄别数据是数组还是对象?

暗示全局变量问题:

```
var a = 10;
function test (a) {
    console.log(a);
    delete a;
    console.log(a);
}
test(20);
delete a;
console.log(a);
```

```
var a = 10;
function test (a) {
    console.log(a);
    delete a;
    console.log(a);
}
test(20);
delete a;
console.log(a);
```

预编译问题:

```
var a = 10;
if (function c() {}) {
    function f () {
        return typeof c + a;
    }
}
function test (a, b) {
```

```
        console.log(a);
        var a = 20;
        console.log(d);
        if (b) {
            function d () {
                console.log(b() + a);
            };
        }
        console.log(d);
        d();
    }
    test(20, f);
}

test(20, f);
```

this 问题:

1.

```
var foo = '123';
function print() {
    this.foo = '234';
    console.log(foo);
};
print();
new print();
```
2.

```
var name = '222';
var a = {
    name: '111',
    say: function () {
        console.log(this.name);
    }
}
a.say();
```

```
var b = {  
  name: '333',  
  say: function (fun) {  
    fun();  
  }  
}  
b.say(a.say);  
b.say = a.say;  
b.say();
```

闭包问题:

```
1.  var arr = [];  
    function test () {  
      for (var i = 0; i < length; i++) {  
        (function (index) {  
          arr[i] = function () {console.log(i)}  
        })(i);  
      }  
    }  
}
```

```
2.  
function foo() {  
  var i = 0;  
  return function() {  
    console.log(i++);  
  }  
}
```

```
var f1 = foo(),  
    f2 = foo();  
f1();  
f1();  
f2();
```

3.

```
function fun(n,o) {  
  console.log(o);  
  return {  
    fun:function(m) {  
      return fun(m,n);  
    }  
  };  
}  
  
var a = fun(0); a.fun(1);  a.fun(2);  a.fun(3);  
var b = fun(0).fun(1).fun(2).fun(3);  
var c = fun(0).fun(1);  c.fun(2);  c.fun(3);
```

算法:

1. 20 个随机数字，放到 3 个数组中 arr1, arr2, arr3 保证 3 个数组的和相差最小。

瀑布流

未完待续...