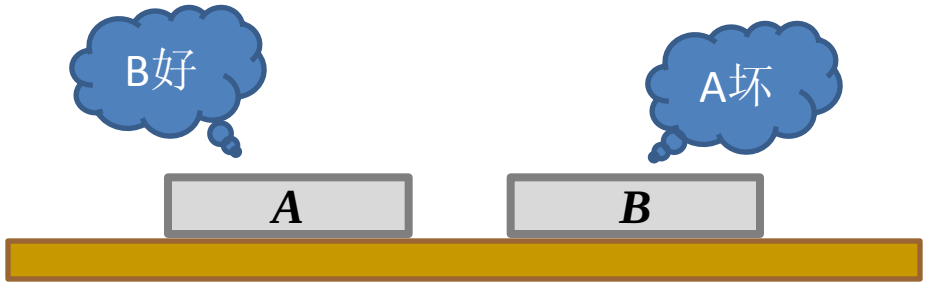
芯片测试

一次测试过程

测试方法：将2片芯片（A和B）置于测试台上，互相进行测试，测试报告是“好”或“坏”，只取其一.



假设：好芯片的报告一定是正确的，坏芯片的报告是不确定的（可能会出错）

测试结果分析

A 报告	*B* 报告	结论
*B*是好的	*A*是好的	*A*, *B* 都好或 *A*, *B* 都坏
*B*是好的	*A*是坏的	至少一片是坏的
*B*是坏的	*A*是好的	至少一片是坏的
*B*是坏的	*A*是坏的	至少一片是坏的

问题

输入：
n **片芯片，其中好芯片至少比坏芯片多 1片.**

问题：

设计一种测试方法，通过测试从 *n* 片芯片中挑出 1 片好芯片.

要求：使用最少的测试次数.

判定芯片A的好坏

问题：给定芯片A，判定A的好坏

方法：用其他 n-1片芯片对 A 测试 .

n=7：好芯片数 ≥ 4.

A 好，6个报告中至少 3 个报“好”

A 坏，6个报告中至少 4 个报“坏”

n是奇数：好芯片数 ≥ $(n+1)/2$.

A 好，至少有 $(n-1)/2$个报“好”

A 坏，至少有 $(n+1)/2$个报告“坏”

结论：至少一半报“好”，A是好芯片，

超过一半报“坏”，A是坏芯片.

判定芯片A的好坏

n=8：好芯片数 ≥ 5.

A 好，7个报告中至少 4 个报“好”

A 坏，7个报告中至少 5 个报“坏”

n是偶数：好芯片数 ≥ $n/2+1$.

A 好，至少有 $n/2$个报告“好”

A 坏，至少有 $n/2+1$个报告“坏”

结论：n-1 份报告中，

至少一半报“好”，则 A 为好芯片

超过一半报“坏”，则 A为坏芯片

蛮力算法

测试方法：任取 1片测试，如果是好芯片，测试结束；如果是坏芯片，抛弃，再从剩下芯片中任取 1片测试，直到得到 1片好芯片.

时间估计：

第1片坏芯片，最多测试 n-2次，

第2片坏芯片，最多测试 n-3次，

…

总计 $\Theta(n^2)$

分治算法设计思想

假设 n为偶数，将 n片芯片两两一组做测试淘汰，剩下芯片构成子问题，进入下一轮分组淘汰.

淘汰规则：

"好, 好" ⇒ 任留 1片，进入下轮

其他情况 ⇒ 全部抛弃

递归截止条件：$n \leq 3$

3 片芯片，1 次测试可得到好芯片.

1 或 2 片芯片，不再需要测试.

分治算法的正确性

命题1 当 n 是偶数时，在上述淘汰规则下，经过一轮淘汰，剩下的好芯片比坏芯片至少多1片.

证 设A, B都好的芯片 i 组, A与B一好一坏 j 组，A与B 都坏的 k 组. 淘汰后好芯片至少 i片，坏芯片至多 k片 .

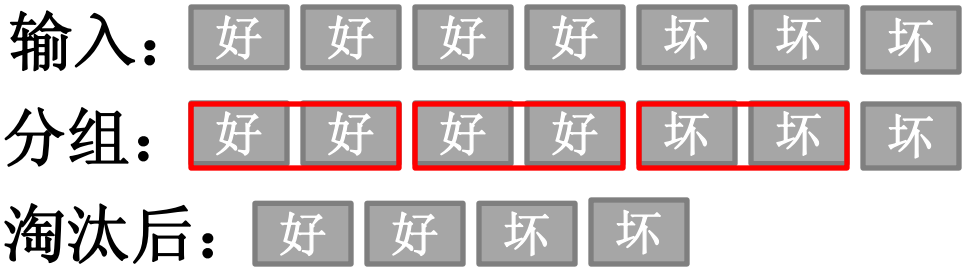
$2i + 2j + 2k = n$ 初始芯片总数

$2i + j > 2k + j$ 初始 好芯片多于坏芯片

$\Rightarrow$ $i > k$

*n*为奇数时的特殊处理

当 *n* 是奇数时，可能出问题



处理办法：当 *n* 为奇数时，增加一轮对轮空芯片的单独测试.

如果该芯片为好芯片，算法结束；
如果是坏芯片，则淘汰该芯片.

伪码描述

算法 Test(n)

1. $k \leftarrow n$
2. while k >3 do
3. 　　将芯片分成$\lfloor k/2 \rfloor$组　// 轮空处理
4. 　　for i =1 to $\lfloor k/2 \rfloor$ do
5. 　　　　if 2片好 then 则任取1片留下
6. 　　　　else 2片同时丢掉
7. 　　k←剩下的芯片数
8. if k =3 then
9. 　　任取2片芯片测试
10. 　　if 1好1坏 then 取没测的芯片
11. 　　else 任取1片被测芯片
12 if k =2 or 1 then 任取1片

分组淘汰

递归结束

时间复杂度分析

设输入规模为n

每轮淘汰后，芯片数至少减半

测试次数(含轮空处理)：$O(n)$

时间复杂度：

$$W(n) = W(n/2) + O(n)$$

$$W(3) = 1,\ W(2) = W(1) = 0$$

解得 $W(n) = O(n)$

小结

- 芯片测试的分治算法

  如何保证子问题与原问题性质相同:
  增加额外处理

  额外处理的工作量不改变函数的阶

  时间复杂度为$O(n)$