

1. 排序

写常见的快速排序等排序算法即可

2. 编写一个函数 king，实现猴子选大王的功能。

新猴王的选择方法是：让 N 只候选猴子围成一圈（最多100只猴子），从某位置起顺序编号为 1 ~ N 号。从第1号开始报数，每轮从1报到3，凡报到3的猴子即退出圈子，接着又从紧邻的下一只猴子开始同样的报数。如此不断循环，最后剩下的一只猴子就选为猴王。

在右侧编辑器中的Begin-End之间补充代码，输出 N（数据由平台提供，已获取，直接使用即可）只猴子中谁是猴子大王的。具体要求如下：

让 N 只候选猴子围成一圈（最多100只猴子），从某位置起顺序编号为 1 ~ N 号；

从第 1 号开始报数，每轮从 1 报到 3，凡报到 3 的猴子即退出圈子，接着又从紧邻的下一只猴子开始同样的报数；如此不断循环，最后剩下的一只猴子就选为猴王。

测试输入：5

预期输出：4号猴子是大王。

测试输入：18

预期输出：14号猴子是大王。

```
1 #include <iostream>
2 using namespace std;
3
4 // 函数king: 猴子选大王
5 // 参数: a-猴子数组n-1个猴子分别占据下标为~n-1的位置, n-数组长度
6 // 返回值: 新猴王的下标序号
7 int king(int a[], int n);
8
9 int main()
10 {
11     int n, a[1000], i;    // 定义变量及数组, n-猴子数量, a-猴子数组
12     cin >> n;           // 输入猴子数量, n>0
13
14     // 初始化猴子数组, n 个猴子分别占据 n 个位置
15     a[0] = 0; // 0号位置没有猴子
16     for(i = 1; i <= n; i++)
17         a[i] = i;
18
19     // 选大王啦
20     i = king(a, n);
21     cout << i << "号猴子是大王。" << endl;
22     return 0;
23 }
24
25 int king(int a[], int n)
26 {
27     // 请在此添加代码, 实现函数king
28     /***** Begin *****/
29     int i, j, t=0;
30     for(i=0; i<=n; i++)
31         a[i]=1;
32     for(i=1; i<=n; i++)
33     {
34         j=1;
35         while(j<=3)
36         {
37             t=(t+1)%n;
38             if(a[t]==1)j++;
39         }
```

```
40     a[t]=0;
41 }
42 return t;
43 /***** End *****/
44 }
```

数学公式法：

```
1 #include <stdio.h>
2 int main()
3 {
4     int n, m, i, s = 0;
5     printf ("N M = ");
6     scanf ("%d%d", &n, &m);
7     for (i = 2; i <= n; i++)
8     {
9         s = (s + m) % i;
10    }
11    printf ("\nThe winner is %d\n", s+1);
12 }
```