## 费马素性检验

14061075 修闽珂

1. 算法原理:

算法的原理运用了费马小定理。

判断一个大奇整数n为素数。随机选取整数b，b在0到n之间，计算b与n的最大公因数d。如果d>1则n不是素数。如果d = 1， 计算b^(n-1)mod n,看看同余式b^(n-1) = 1(mod n)。如果不成立，n不是素数；如果成立，则n是合数的可能性小于0.5，即n是素数的可能性为1-0.5。重复以上步骤直到重复安全次数k次，此时n是素数的可能性为1-0.5^k，可以认定n是素数了。

1. 测试样例

|  |  |
| --- | --- |
| 51 | not. the time is:0.0 |
| 967 | yes. the time is:0.0 |
| 698594 | not. the time is:0.0 |
| 34528 | not. the time is:0.0 |
| 69 | not. the time is:0.0 |
| 70695 | not. the time is:0.0 |
| 530148 | not. the time is:0.0 |
| 360924 | not. the time is:0.0 |
| 604701 | not. the time is:0.0 |
| 98 | not. the time is:0.0 |

1. 心得体会

这个程序就是按照算法照搬就可以了。但是有一个很大的问题就是可运算的位数并不大。可能一方面是电脑的机能问题，也有可能是编程语言的效率问题，更有可能的是代码算法的优化。我尚且做不到超长位数的加减乘除算法，或者更好的优化的长整数的运算方法，这个肯能会影响我的代码的执行效率。经大神指点迷津我实现了模重复平方求模的算法，算法的效率大大提高。改进后的算法能够计算几千位的数字

1. 算法流程

程序1：

以写入方法打开文件input

随机生成大数和小数的个数

生成小数，直接在0-100内调用randint函数

生成大数，先生成大数的位数，然后每一位都是由0-9的数字构成（首位不能是0）

关掉文件

程序2：

只读打开input，写入打开output

安全次数为10

如果读入的数字是1或者0，直接认定不是质数

如果是10以内的数字直接判断

如果是较大的数，就照搬算法原理中的算法

关掉文件