**西安电子科技大学网信院**

**信息安全基础与密码学**

**综合实验**

**实 验 报 告（一）**

**Fermat素性检测算法**

**班级：2218031**

**姓名：周仙辉**

**学号：22009201377**

**日期：2024-10-20**

一、实验目的（包括实验环境、实现目标等等）

实验环境：Windows-wsl2-ubuntu22.04，c++ 11

目标：实现大数Fermat素性检测算法

二、方案设计

（包括背景、原理、必要的公式、图表、算法步骤等等）

背景和原理：

* Fermat 小定理：给定素数p,aEZ,则有aP-1=1(modp)。
* 如果有一个整数a,(a,m)=1,使得am-1=1(modm),则m是素数或者伪随机数。

算法步骤：

给定奇整数m≥3和安全参数k

(1) 随机选取整数a,2≤a≤m-2;

(2) 计算g=(a,m),如果g=1,转(3);否则,跳出,m为合数;

(3)计算r=am-1(modm),如果r=1,m可能是素数,转(1);否则,跳出,m为合数;

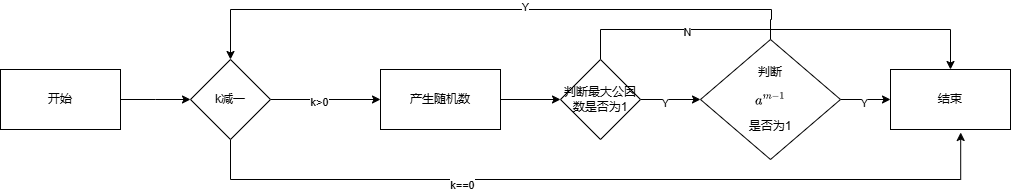
(4)重复上述过程k次,如果每次得到m可能为素数/则m为素数的概率为1-张

三、方案实现

（包括算法流程图、主要函数的介绍、算法实现的主要代码等等）

算法流程：

算法流程图：



主要函数介绍：

void gcd (const mpz\_t x,const mpz\_t y,mpz\_t res)

通过欧几里得算法获取x,y的最大公因数，存在res中

void qmi(const mpz\_t x,const mpz\_t y,mpz\_t res)

通过快速幂算法，将mody存储在res中。

void randinit()

初始化随机数生成器

主要代码：

std::pair<bool,double> fpt (mpz\_t m, mpz\_t k){

double rate = 1;

mpz\_t a,res;

mpz\_init(a),mpz\_init(res);

mpz\_mod\_ui(a,m,2);

if( mpz\_cmp\_ui(m,3) < 0 || mpz\_cmp\_ui(a,0) == 0) return {false,1};

for(int i=0; mpz\_cmp\_ui(k,i) > 0 ;i++){

// （1）随机数 a ; 2 <= a <= m-2

mpz\_urandomm(a,state,m);

mpz\_sub\_ui(m,m,4);

mpz\_mod(a,a,m);

mpz\_add\_ui(a,a,2);

mpz\_add\_ui(m,m,4);

// (2) 最大公约数

gcd(a,m,res);

if(mpz\_cmp\_ui(res,1) != 0){

mpz\_clear(res);

mpz\_clear(a);

return {false,0};

}

// (3) 幂运算

qmi(a,m,res);

if(mpz\_cmp\_ui(res,1) != 0) {

mpz\_clear(res);

mpz\_clear(a);

return {false,0};

}

else rate \*= 0.5;

}

mpz\_clear(a),mpz\_clear(res);

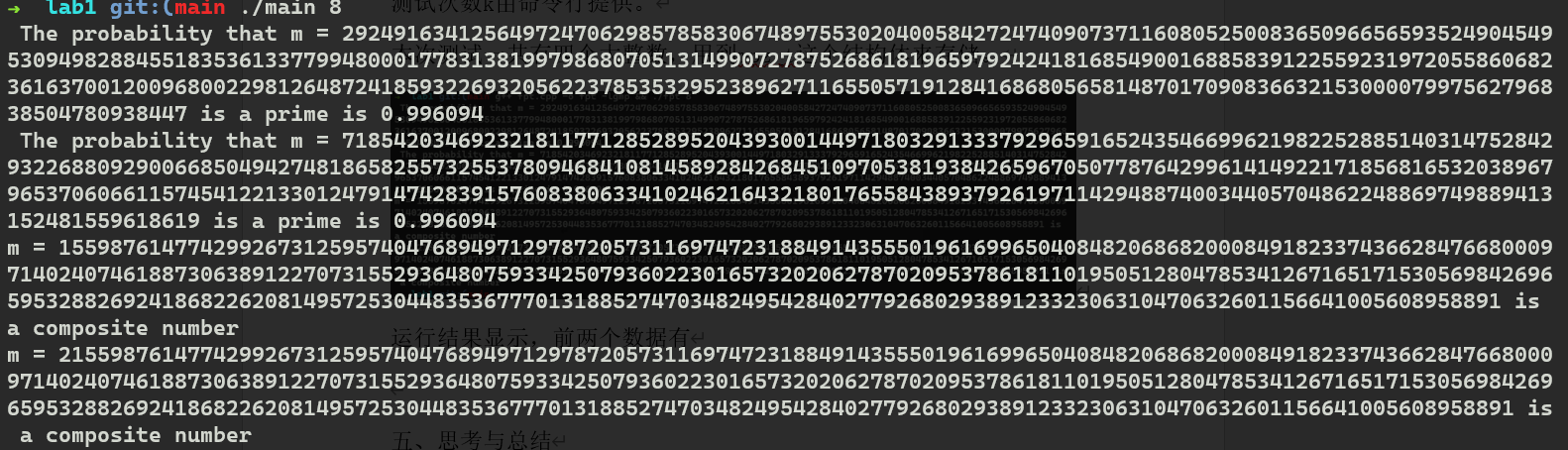
return {true,1-rate};

}

四、数据分析(包括算法测试数据的分析，运行结果截图等等)

测试次数k由命令行提供。

本次测试一共有四个大整数，用到mpz\_t这个结构体来存储。



运行结果显示，前两个数据有0.996094的概率为素数，后两个数据则为合数。

五、思考与总结

1. 如果有一个整数𝒂，(𝒂,𝒎)=𝟏，使得𝒂𝒎−𝟏≡𝟏 𝒎𝒐𝒅 𝒎 则𝒎一定是一个素数吗？为什么？（请简述并举例说明，不能只简单回答“是”或“不是”）

不是，有可能是合数。例如：4^2 ≡ 1mod3 ,但4是合数。

1. Fermat素性检测中都用到了哪些运算？分别实现什么功能？请简述。

随机数生成：实现了寻找 的随机数，然后才能执行其他功能。

模运算： 实现gcd

幂模运算：计算

1. 你还了解哪种素性检测算法？请简述，并分析其与Fermat素性检测算法的区别与联系。

Miller-Rabin 素性测试

对于一个奇数 n，如果 n 是素数，随机选择一个 a 满足 ，则以下条件之一必须成立：

* 存在某个r使得

其中 d 是 n−1 的一个奇数因子，即（将n−1分解为2的幂次和一个奇数因子）。

区别与联系：

都有用到费马小定理。但是Miller-Rabin的筛选更加严格，准确性更高。

1. 实验过程中还遇到了什么问题，如何解决的？通过该实验有何收获？

问题

C++下自己实现的大整数，运行效率太低。使用gmp可以提高效率。

收获

理解了fermat素性检测。复习了幂模算法，欧几里得算法。

**实验报告提交说明**：

1. 实验报告同时提交word文档与源代码（.c或者.py）。
2. 实验报告与源代码命名规则：实验1-学号-姓名，例如：实验1-20009200400-张三。
3. 请于截止日期前在西电智课平台（学在西电）提交相关文档，逾期未提交，该部分成绩记为0分，周知。