南京信息工程大学 实验（实习）报告

实验名称 RSA加密 实验（实习）日期 2023/11/14 指导教师 高光勇

系 计算机 专业 奇安信 年级 2021 班次 1 姓名 朱宸扬 学号 202183760012

**一、实验目的**

**熟悉RSA加解密算法的运行过程，编写实现RSA算法程序。**

**二、实验要求**

**(1) 实现判定互素、筛选素参数算法**

**(2) 实现RSA加/解密算法**

**三、实验内容 (要求写出算法原理、实验核心代码及运行结果)**

**在90001~100000随机找了素数p,q**

**模逆用的欧几里得**

**最大公因数用的辗转相除法**

import random  
list = []  
for i in range(90001,100001):  
 flag = 0  
 for j in range(2,i-1):  
 if i%j==0:  
 flag=1  
 break  
 if flag == 0:  
 list.append(i)  
  
random\_primes = random.sample(list, 2)  
p = random\_primes[0]  
q = random\_primes[1]  
print("p:",p)  
print("q:",q)  
  
r = p\*q  
  
Euler = (p-1)\*(q-1)  
print("Euler:",Euler)  
  
def gcd(a, b):  
 while b:  
 a, b = b, a % b  
 return a  
  
list\_gcd = []  
for i in range(90001,100001):  
 if gcd(i, Euler) == 1:  
 list\_gcd.append(i)  
  
e = random.choice(list\_gcd)  
print("e:",e)  
  
def mod\_inverse(a, m):  
 m0, x0, x1 = m, 0, 1  
 while a > 1:  
 q = a // m  
 m, a = a % m, m  
 x0, x1 = x1 - q \* x0, x0  
 return x1 + m0 if x1 < 0 else x1  
  
d = mod\_inverse(e,Euler)  
print("d:",d)  
import random  
  
  
# 生成随机明文  
plaintext = random.getrandbits(25) # 25bit大小密文  
# 使用公钥进行加密  
ciphertext = pow(plaintext, e, r) # 其中e是公钥指数，n是模数  
print("明文:", plaintext)  
print("密文:", ciphertext)  
decr = pow(ciphertext,d,r)  
print("解密:",decr)

**p: 91807**

**q: 99961**

**Euler: 9176927760**

**e: 98411**

**d: 7820404931**

**明文: 14761884**

**密文: 587285661**

**解密: 14761884**

**四、实验心得**

RSA算法的核心是利用明文和密文幂e,d次方后的互相转化，欧拉函数是一个找e的辅助概念,保证了e和欧拉互素后，才能找到d，使得ed (modr)=1

实验后，我加深了对这一过程印象和理解