Министерство образования Республики Беларусь

Учреждение образования

«Брестский Государственный технический университет»

Кафедра ИИТ

**Лабораторная работа №4**

По дисциплине «Криптографические методы защиты информации»

Тема: «Факторизация составного числа»

**Выполнил:**

Студент 2 курса

Группы ИИ-21

Литвинюк Т. В.

**Проверил:**

Хацкевич М. В.

Брест 2023

**Цель:** освоить алгоритмы факторизации составного числа.

**Ход работы:**



#include <iostream>

#include <cmath>

#include "../Libs/bignumber.h"

using namespace pr0crustes;

using namespace std;

bool isPrime(unsigned long long N);

void dividing(unsigned long long N);

unsigned long long gcd(unsigned long long a, unsigned long long b);

pair<unsigned long long, unsigned long long> fermat\_factorization(unsigned long long n);

pair<unsigned long long, unsigned long long> combined\_method(unsigned long long n);

int main() {

short menu;

unsigned long long N;

while (true){

cout << "1. Метод пробного деления.\n" <<

"2. Метод Ферма (без операции деления).\n" <<

"3. Комбинированный метод.\n" <<

"4. Выход из программы.\n> ";

cin >> menu;

switch(menu){

case 1:{

cout << "Введите число: "; cin >> N;

dividing(N); cout << endl << endl;

break;

}

case 2:{

cout << "Введите число: "; cin >> N;

pair<unsigned long long, unsigned long long> a = fermat\_factorization(N);

cout << a.first << " " << a.second << endl;

break;

}

case 3:{

cout << "Введите число: "; cin >> N;

pair<unsigned long long, unsigned long long> a = combined\_method(N);

cout << a.first << " " << a.second << endl;

break;

}

case 4:{

return 0;

break;

}

}

}

}

bool isPrime(unsigned long long n){

bool is\_prime = true;

for (unsigned long long i = 2; i <= sqrt(n); i += 2) {

if (n % i == 0) {

is\_prime = false;

break;

}

}

return is\_prime;

}

void dividing(unsigned long long n){

if (n % 2 == 0) {

cout << "2 ";

n /= 2;

}

for (int i = 3; i <= sqrt(n); i += 2) {

while (n % i == 0) {

cout << i << " ";

n /= i;

}

}

if (n > 2) {

cout << n;

}

}

unsigned long long gcd(unsigned long long a, unsigned long long b) {

if (b == 0) return a;

return gcd(b, a % b);

}

pair<unsigned long long, unsigned long long> fermat\_factorization(unsigned long long n) {

if (n % 2 == 0)

return make\_pair(2, n / 2);

unsigned long long k = sqrt(n);

while (true) {

unsigned long long r = sqrt(k \* k - n);

if (r \* r == k \* k - n) {

return make\_pair(k - r, k + r);

}

k++;

}

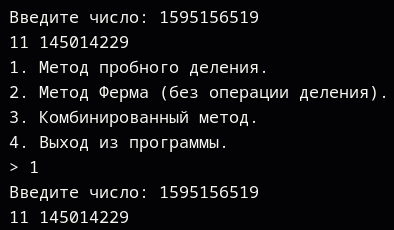
}

pair<unsigned long long, unsigned long long> combined\_method(unsigned long long n){

if (isPrime(n))

return make\_pair(1, n);

return fermat\_factorization(n);

}

}

Метод Ферма занимает гораздо больше времени на разложение чисел на простые множители.

1595156519÷11=145014229, числа 11 и 145014229 — простые, значит, программа отработала правильно.

**Вывод:** в ходе лабораторной работы освоил алгоритмы факторизации составных чисел.