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);

bool fermat\_factorization(unsigned long long n);

void 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;

fermat\_factorization(N); cout << endl;

break;

case 3:

cout << "Введите число: "; cin >> N;

combined\_method(N); cout << 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;

}

}

bool fermat\_factorization(unsigned long long n) {

unsigned long long a = ceil(sqrt(n)); // находим первое число, ближайшее к sqrt(n)

unsigned long long b2 = a\*a - n; // b^2 = a^2 - n

unsigned long long b = sqrt(b2);

unsigned count = 0; // ограничение на количество итераций

while (b \* b != b2) { // если b не целое, то увеличиваем a и пересчитываем b^2

a++;

b2 = a\*a - n;

b = sqrt(b2);

count++;

if (count > 100) { // ограничение на количество итераций

return false;

}

}

cout << a - b << " " << a + b << endl;

return true;

}

void combined\_method(unsigned long long n){

bool is\_prime = isPrime(n);

if (is\_prime) {

cout << n << " is a prime number." << endl;

return;

}

dividing(n);

cout << "\n\nМетод Ферма: " << endl;

unsigned long long k = ceil(sqrt(n));

while (true) {

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

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

cout << (k - r) << (k + r) << endl;

return;

}

k++;

}

}

**Вывод:** в ходе лабораторной работы я научился шифровать и сжимать информацию.