

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение высшего
образования

**«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ Н.Г. ЧЕРНЫШЕВСКОГО»**

Кафедра теоретических основ
компьютерной безопасности и
криптографии

Проверка чисел на простоту с помощью свойств чисел Кармайкла

ЛАБОРАТОРНАЯ РАБОТА

студента 4 курса 431 группы

специальности 10.05.01 Компьютерная безопасность

факультета компьютерных наук и информационных технологий

Серебрякова Алексея Владимировича

Научный руководитель

доцент, к. п. н.

подпись, дата

А. С. Гераськин

Саратов 2022

Описание алгоритма

2. Теорема Кармайкла

Теорема 1 (Кармайкл, 1912). Пусть n — нечетное составное, тогда:

1. если $n \div p^2$, где p — простое, то n — не число Кармайкла;
2. если $n = p_1 \cdot p_2 \cdot \dots \cdot p_k$, где $p_i \neq p_j$ при $i \neq j$, то для того, чтобы n являлось числом Кармайкла необходимо и достаточно, чтобы $(n - 1) \div (p_i - 1)$ ($i = \overline{1, k}$);
3. если $n = p_1 \cdot p_2 \cdot \dots \cdot p_k$, где $p_i \neq p_j$ при $i \neq j$ и n — число Кармайкла, то $k \geq 3$.

Код программы

```
#include <bits/stdc++.h>
#include <iostream>

using namespace std;

int validated_input()
{
    int s = 0;
    while (!(cin >> s))
    {
        cin.clear();
        cin.ignore(numeric_limits<streamsize>::max(), '\n');
        printf("! Неверный ввод. Повторите ввод, начиная с первого неверного элемента.\n");
    }

    return s;
}

int factmod(int n, int p)
{
    // printf("\nПроверка теоремы Вильсона для нашего случая:\n  %d! + 1 (mod %d)\n  ", n, p);
    int res = 1;
    while (n > 1)
    {
        res = (res * ((n / p) % 2 ? p - 1 : 1)) % p;
        for (int i = 2; i <= n % p; ++i)
            res = (res * i) % p;
        n /= p;
    }
    // printf("(p-1)! (mod p) = %d! (mod %d)\n", res % p, p);

    return res % p;
}

bool test(int p)
{
    bool res = (factmod(p - 1, p) + 1) % p == 0;

    return res;
}

bool sv1(int n)
{

```

```

bool res(true);
bool tst;

printf("\nПроверка 1:\n Если n кратно  $p^2$  где p - простое, то n - не число Кармайкла\n");

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

    test(i) ? tst = true : tst = false;

    if (n % (i * i) == 0 && tst){
        res = false;
        printf("[%d]", i);
    }
}

printf(res ? "\nТест 1 пройден\n" : "\nТест 1 не пройден\n");

return res;
}

bool sv2(int n)
{
    printf("\nПроверка 2:\n Если  $n = p_1 * p_2 * \dots * p_k$ ,  $p_i \neq p_j$ ,  $i \neq j$  и при этом  $(n-1)$  кратно  $(p_i - 1) \Leftrightarrow n$  - число Кармайкла\n");
    vector<int> v;
    bool res(true);
    int del(2);

    while (n > 1)
    {
        if (n % del == 0)
        {
            v.push_back(del);

            while (n % del == 0)
                n /= del;
        }

        del++;
    }

    for (auto e : v)
    {
        if ((n - 1) % (e - 1) != 0)
            res = false;
    }

    printf(res ? "\nТест 2 пройден\n" : "\nТест 2 не пройден\n");
    return res;
}

bool sv3(int n)
{
    printf("\nПроверка 2:\n Если  $n = p_1 * p_2 * \dots * p_k$ ,  $p_i \neq p_j$ ,  $i \neq j$  и при этом  $k \geq 3 \Leftrightarrow n$  - число Кармайкла\n");
    vector<int> v;
    int del(2);

    while (n > 1)

```

```

{
    if (n % del == 0)
    {
        v.push_back(del);

        while (n % del == 0)
            n /= del;
    }

    del++;
}

printf(v.size() >= 3 ? "\nТест 3 пройден\n" : "\nТест 3 не пройден\n");
return v.size() >= 3;
}

int main()
{
    setlocale(0, "");
    int n, ntmp;

    printf("\nВведите число n: ");
    n = validated_input();
    ntmp = n;

    printf("\nПроверим число %d на простоту по критерию Вильсона\n", n);

    (test(n)) ?
        printf("\nОтвет: %d - простое число\n", ntmp) :
        printf("\nОтвет: %d - не простое число\n", ntmp);

    printf("\nПроверим число на свойства числа Кармайкла\n");

    if (n % 2 == 0)
    {
        printf("\nЧисло четное - выполнить тест не удастся\n");
        return 0;
    }

    bool t1(sv1(n)), t2(sv2(n)), t3(sv3(n));

    (t1 && t2 && t3) ? printf("\n%d это число Кармайкла\n\n", n) : printf("\n%d это не число Кармайкла\n\n", n);

    return 0;
}

```

Пример запуска программы

• * Executing task: /bin/bash -c ./build/Debug/outDebug

Введите число n: 1105

Проверим число 1105 на простоту по критерию Вильсона

Ответ: 1105 - не простое число

Проверим число на свойства числа Кармайкла

Проверка 1:

Если n кратно p^2 где p - простое, то n - не число Кармайкла

Тест 1 пройден

Проверка 2:

Если $n = p_1 p_2 \dots p_k$, $p_i \neq p_j$, $i \neq j$ и при этом $(n-1)$ кратно $(p_i - 1) \Leftrightarrow n$ - число Кармайкла

Тест 2 пройден

Проверка 2:

Если $n = p_1 p_2 \dots p_k$, $p_i \neq p_j$, $i \neq j$ и при этом $k \geq 3 \Leftrightarrow n$ - число Кармайкла

Тест 3 пройден

1105 это число Кармайкла

* Terminal will be reused by tasks, press any key to close it.

• * Executing task: /bin/bash -c ./build/Debug/outDebug

Введите число n: 132165421

Проверим число 132165421 на простоту по критерию Вильсона

Ответ: 132165421 - не простое число

Проверим число на свойства числа Кармайкла

Проверка 1:

Если n кратно p^2 где p - простое, то n - не число Кармайкла

Тест 1 пройден

Проверка 2:

Если $n = p_1 p_2 \dots p_k$, $p_i \neq p_j$, $i \neq j$ и при этом $(n-1)$ кратно $(p_i - 1) \Leftrightarrow n$ - число Кармайкла

Тест 2 пройден

Проверка 2:

Если $n = p_1 p_2 \dots p_k$, $p_i \neq p_j$, $i \neq j$ и при этом $k \geq 3 \Leftrightarrow n$ - число Кармайкла

Тест 3 не пройден

132165421 это не число Кармайкла

* Terminal will be reused by tasks, press any key to close it.