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

Решение сравнений с помощью алгоритма Евклида

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. _____ А. С. Гераськин

подпись, дата

Саратов 2021

Описание алгоритма

Решение сравнения первой степени с использованием расширенного алгоритма Евклида

Рассмотрим расширенный алгоритм Евклида[8].

Как мы убедились в обычном алгоритме Евклида чтобы найти u, v надо выражать через друг друга остатки r_i, что не удобно.

Данный алгоритм позволяет найти u, v в $d = au + mv$, $НОД(a,m) = d$.

Необходимо работать с системой равенств:

$$\begin{cases} y_1 a + y_2 m = u_3 \\ z_1 a + z_2 m = v_3 \\ t_1 a + t_2 m = t_3 \end{cases}$$

На первом шаге полагаем: $(y_1, y_2, y_3) = (1,0,m)$, $(z_1, z_2, z_3) = (0,1,a)$, $(t_1, t_2, t_3) = (0,0,0)$

На очередном шаге проверяем: если $z_3 = 0$, то искомые значения d, u, v равны y_3, y_2, y_1. Иначе производим деление с остатком: $y_3 = qz_3 + r$,

$(t_1, t_2, t_3) = (v_1, v_2, v_3)$

$(v_1, v_2, v_3) = (u_1 - qv_1, u_2 - qv_2, u_3 - qv_3)$.

$(u_1, u_2, u_3) = (t_1, t_2, t_3)$

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 euclid_extended(int a, int b, int &x, int &y)
{
   int x1, y1;

   if (a == 0)
   {
      x = 0;
      y = 1;
      return b;
   }

   int d = euclid_extended(b % a, a, x1, y1);

   x = y1 - (b / a) * x1;
```

```
    y = x1;

    return d;
}

bool task(int a, int b, int c, int &x0, int &y0, int &g)
{
    g = euclid_extended(abs(a), abs(b), x0, y0);
    if (c % g != 0)
        return false;
    x0 *= c / g;
    y0 *= c / g;
    if (a < 0)
        x0 *= -1;
    if (b < 0)
        y0 *= -1;
    return true;
}

int norm(int x, int m){

    while (x < 0)
        x+= m;

    return x % m;
}

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

    int a, b, m, x, y, g, tmp;

    printf("\nРешение уравнения вида\n  ax = b (mod m)\n");
    printf("можно свести к решению Диофантового уравнения вида\n ax + km = b\n");
    printf("которое в свою очередь может быть решено с помощью расширенного алгоритма Евклида.\n");

    printf("\nВведите коэффициент a: ");
    a = validated_input();

    printf("\nВведите коэффициент b: ");
    b = validated_input();

    printf("\nВведите коэффициент m: ");
    m = validated_input();
    tmp = m;

    printf("\nТаким образом уравнение примет вид:\n %dx = %d (mod %d)\n", a, b, m);
    printf("\nРешим его!\n");

    task(a % tmp, m, b % tmp, x, y, g) ? printf("\nРешение:\n    x = %d\n", (norm(x,m))) : printf("\nРешения
нет!\n");

    return 0;
}
```

Пример запуска программы

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

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


Решение уравнения вида
  ax = b (mod m)
можно свести к решению Диофантового уравнения вида
 ax + km = b
которое в свою очередь может быть решено с помощью расширенного алгоритма Евклида.

Введите коэффициент a: 111

Введите коэффициент b: 75

Введите коэффициент m: 322

Таким образом уравнение примет вид:
 111x = 75 (mod 322)

Решим его!

Решение:
    x = 79
 *  Terminal will be reused by tasks, press any key to close it.
```

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


Решение уравнения вида
  ax = b (mod m)
можно свести к решению Диофантового уравнения вида
 ax + km = b
которое в свою очередь может быть решено с помощью расширенного алгоритма Евклида.

Введите коэффициент a: 9

Введите коэффициент b: 6

Введите коэффициент m: 12

Таким образом уравнение примет вид:
 9x = 6 (mod 12)

Решим его!

Решение:
    x = 10
 *  Terminal will be reused by tasks, press any key to close it.
```