

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

Реализация алгоритма полиномиального деления PDF

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

PDF. Полиномиальное деление над полем (Polynomial Division over a Field)

Вход: $p_1(x) = \sum_{0 \leq i \leq m} c_i x^i$ и $p_2(x) = \sum_{0 \leq i \leq n} d_i x^i$ над полем, $m \geq n \geq 0$ и $d_n \neq 0$. (Этот алгоритм будет работать и над областью целостности J при условии, что d_n обратим в J .)

Выход: $q(x) = \sum_{0 \leq i \leq m-n} q_i x^i$ и $\sum_{0 \leq i \leq n-1} r_i x^i$, обладающие свойством евклидовости (теорема 3.1.1).

1. [Основной цикл] Для k от $m-n$ до 0 выполнять $\{q_k := c_{n+k}/d_n$; для j от $n+k-1$ до k выполнять $c_j := c_j - q_k d_{j-k}\}$.
2. [Выход] Вернуть q_i , $i = 0, 1, 2, \dots, m-n$, коэффициенты полинома $q(x)$, вычисленного на шаге 1, и r_i , $i = 0, 1, 2, \dots, n-1$, коэффициенты полинома $r(x)$, где $r_i = c_i$ (c_i также вычисляются на шаге 1).

Код программы

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

using namespace std;

typedef vector<int> vect;
typedef pair<vect, vect> vect_pair;

vect_pair v;

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;
}

void pdf(vect a, vect b)
{
    int a_len(a.size() - 1), b_len(b.size() - 1);
    vect q(a_len - b_len + 1);

    for (int i = 0; i <= a_len - b_len; i++)
    {
        q[i] = a[i] / b[0];

        for (int j = i; j <= b_len + i; j++)
```

```

        a[j] = a[j] - q[i] * b[j - i];
    }

    v.first = q;
    v.second = a;
}

vect get()
{
    vect s;
    int n;

    printf("\nВведите степень многочлена:\n ");
    n = validated_input();

    printf("\nВведите коэффициенты многочлена:\n ");
    for (int i = 0; i < n + 1; i++)
    {
        printf("\n [X ^ %d] --> ", n - i);
        s.push_back(validated_input());
    }

    return s;
}

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

    vect poly1, poly2;

    poly1 = get();
    poly2 = get();

    pdf(poly1, poly2);

    printf("\nЧастное:\n ");
    for(auto e: v.first)
        printf("{%3d} ", e);

    printf("\nОстаток:\n ");
    for(auto e: v.second)
        printf("{%3d} ", e);

    return 0;
}

```

Пример запуска программы

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

Введите степень многочлена:

4

Введите коэффициенты многочлена:

[X ^ 4] --> 2

[X ^ 3] --> -10

[X ^ 2] --> 23

[X ^ 1] --> -22

[X ^ 0] --> -3

Введите степень многочлена:

2

Введите коэффициенты многочлена:

[X ^ 2] --> 1

[X ^ 1] --> -3

[X ^ 0] --> 5

Частное:

{ 2} { -4} { 1}

Остаток:

{ 0} { 0} { 0} { 1} { -8} * Terminal will be reused by tasks, press any key to close it.

$$\begin{array}{r|l} 2x^4 - 10x^3 + 23x^2 - 22x - 3 & x^2 - 3x + 5 \\ \hline 2x^4 - 6x^3 + 10x^2 & 2x^2 - 4x + 1 \\ \hline -4x^3 + 13x^2 - 22x - 3 & \\ -4x^3 + 12x^2 - 20x & \\ \hline & x^2 - 2x - 3 \\ & x^2 - 3x + 5 \\ \hline & x - 8 \end{array}$$

```

Консоль отладки Microsoft Visual Studio

Введите степень 1-го многочлена: 3

Введите коэффициенты:

Коэффициент при x^3: 4
Коэффициент при x^2: 0
Коэффициент при x^1: 2
Коэффициент при x^0: -11

Введите степень 2-го многочлена: 1

Введите коэффициенты:

Коэффициент при x^1: 1
Коэффициент при x^0: 5

Коэффициенты при частном:
4 -20 102
Коэффициенты при остатке:
0 -521

C:\Users\HP\source\repos\ААиТЧ 05-01\Debug\ААиТЧ 05-01.exe
(процесс 11120) завершил работу с кодом 0.
Чтобы автоматически закрывать консоль при остановке отладки
, включите параметр "Сервис" ->"Параметры" ->"Отладка" -> "
Автоматически закрыть консоль при остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно...

```

$4x^3$	$+0x^2$	$+2x$	-11		x	$+5$
$4x^3$	$+20x^2$				$4x^2$	$-20x + 102$
	$-20x^2$	$+2x$	-11			
	$-20x^2$	$-100x$				
		$102x$	-11			
		$102x$	$+510$			
			-521			