Лабораторная работа №8

Математические основы защиты информации и информационной безопасности

Минов Кирилл Вячеславович | НПМмд-02-23

Содержание

1 Цель работы

Реализовать на языке программирования алгоритмы для выполнения арифметических операций с большими целыми числами.

2 Теоретическое введение

Считаем, что число записано в b-ичной системе счисления, b - натуральное число, $b \geq 2$. Натуральное n-разрядное число будем записывать в виде

$$u = u_1 u_2 \dots u_n$$

.При работе с большими целыми числами знак такого числа удобно хранить в отдельной переменной. Например, при умножении двух чисел, знак произведения вычисляется отдельно.

3 Выполнение лабораторной работы

Алгоритм 1 (сложение неотрицательных целых чисел).

Вход. Два неотрицательных числа $u = u_1 u_2 \dots u_n$ и $v = v_1 v_2 \dots v_n$; разрядность чисел n; основание системы счисления b.

Выход. Сумма $w = w_0 w_1 \dots w_n$, где w_0 - цифра переноса - всегда равная 0 или 1.

Код программы (рис. 1).

```java
public class alg1 {
    public static void main(String[] args) {
        a1( uu: 23,  vv: 11,  b: 10);
    }
    static void a1(int uu, int vv, int b) {
        int[] u = getDigits(uu);
        int[] v = getDigits(vv);
        int[] w = new int[Math.max(u.length, v.length) + 1];
        int n = u.length - 1;
        int j = n;
        int k = 0;
        while (j >= 0) {
            w[j + 1] = (u[j] + v[j] + k) % b;
            k = (u[j] + v[j] + k) / b;
            j--;
        }
        w[0] = k;
        printArray(removeZeros(w));
    }
    static int[] getDigits(int number) {
        String strNumber = Integer.toString(number);
        int[] digits = new int[strNumber.length()];
        for (int i = 0; i < strNumber.length(); i++) {
            digits[i] = Character.getNumericValue(strNumber.charAt(i));
        }
        return digits;
    }
    static int[] removeZeros(int[] array) {
        int startIndex = 0;
        while (startIndex < array.length - 1 && array[startIndex] == 0) {...}
        int newSize = array.length - startIndex;
        int[] result = new int[newSize];
        System.arraycopy(array, startIndex, result,  destPos: 0, newSize);
        return result;
    }
    static void printArray(int[] array) {
        for (int i : array) {
            System.out.print(i);
        }
    }
```

Рис. 1: Алгоритм 1 (сложение неотрицательных целых чисел)

Алгоритм 2 (вычитание неотрицательных целых чисел).

Вход. Два неотрицательных числа $u = u_1u_2\dots u_n$ и $v = v_1v_2\dots v_n$, $u > v$; разрядность чисел n; основание системы счисления b.

Выход. Сумма $w = w_1 w_2 \dots w_n = u - v$.

Код программы (рис. 2).

```
def remove_zeros(w):
        z = 0
        while w[z] == 0:
            z = z + 1
        return(w[z:])
#Алгоритм 2
def a2(uu,vv,b):
    u = [int(i) for i in str(uu)]
    v = [int(i) for i in str(vv)]
    l = len(u)
    w = []
    n = len(u) - 1

    j = n
    k = 0
    while j != -1:
        w.append((u[j] - v[j] + k) % b)
        k = (u[j] - v[j] + k) // b
        j = j - 1
    w.reverse()
    return print(''.join(str(i) for i in remove_zeros(w)))

a2(2035, 2000, 10)
35
```

Рис. 2: Алгоритм 2 (вычитание неотрицательных целых чисел)

Алгоритм 3 (умножение неотрицательных целых чисел столбиком).

Вход. Числа $u = u_1 u_2 \dots u_n$ и $v = v_1 v_2 \dots v_m$; основание системы счисления b.

Выход. Произведение $w = uv = w_1 w_2 \dots w_{m+n}$.

Код программы (рис. 3).

```
#Алгоритм 3
def a3(uu,vv,b):
    u = [int(i) for i in str(uu)]
    v = [int(i) for i in str(vv)]
    n = len(u) - 1
    m = len(v) - 1
    j = m
    w = [0] * (len(u) + len(v))
    while j >= 0:
        if v[j] == 0:
            w[j] == 0
            j = j - 1
        else:
            i = n
            k = 0
            while i >= 0:
                t = u[i] * v[j] + w[i + j + 1] + k
                w[i + j + 1] = t % b
                k = t // b
                i = i - 1
            w[j] = k
            j = j - 1
        z = 0
        while w[z] == 0:
            z = z + 1
    return print(''.join(str(i) for i in remove_zeros(w)))
a3(5497, 296, 10)
1627112
```

Рис. 3: Алгоритм 3 (умножение неотрицательных целых чисел столбиком)

Алгоритм 4 (быстрый столбик).

Вход. Числа $u = u_1u_2\dots u_n$ и $v = v_1v_2\dots v_m$; основание системы счисления b.

Выход. Произведение $w = uv = w_1w_2\dots w_{m+n}$.

Код программы (рис. 4).

```
#Алгоритм 4
def a4(uu,vv,b):
    u = [int(i) for i in str(uu)]
    v = [int(i) for i in str(vv)]
    n = len(u) - 1
    m = len(v) - 1
    w = [0] * (len(u) + len(v))

    t = 0
    for s in range(m + n + 2):
        for i in range(s + 1):
            if (n - i < 0) or (m - s + i < 0):
                t = t
            else:
                t = t + u[n - i] * v[m - s + i]
        w[m + n - s + 1] = t % b
        t = t // b
    return print(''.join(str(i) for i in remove_zeros(w)))

a4(5497, 296, 10)
1627112
```

Рис. 4: Алгоритм 4 (быстрый столбик)

Алгоритм 5 (деление многоразрядных целых чисел).

Вход. Числа $u = u_n \dots u_1 u_0$ и $v = v_t \dots v_1 v_0$, $n \geq t \geq 1$, $v_t \neq 0$, разрядность чисел соответственно n и t.

Выход. Частное $q = q_{n-t} \dots q_0$, остаток $r = r_t \dots r_0$.

Код программы (рис. 5).

```python
#Алгоритм 5
def a5(uu,vv,b):
    u = uu
    v = vv

    n = len([int(i) for i in str(uu)]) - 1
    t = len([int(i) for i in str(vv)]) - 1
    q = [0] * (n - t + 1)
    r = [0] * (t + 1)


    while u >= v * b ** (n - t):
        q[n-t] = q[n-t] + 1
        u = u - v * b ** (n - t)

    n = len([int(i) for i in str(u)]) - 1
    t = len([int(i) for i in str(v)]) - 1

    for i in range(n, t, -1):
        u_ = [int(i) for i in str(u)]
        u_.reverse()
        v_ = [int(i) for i in str(v)]
        v_.reverse()

        if u_[i] >= v_[t]:
            q[i-t-1] = b - 1
        else:
            q[i-t-1] = (u_[i] * b + u_[i-1]) // v_[t]

        while q[i-t-1] * (v_[t] * b + v_[t-1]) > u_[i] * b ** 2 + u_[i-1] * b + u_[i-2]:
            q[i-t-1] = q[i-t-1] - 1
        u = u - q[i-t-1] * b ** (i - t - 1) * v

        if u < 0:
            u = u + v * b ** (i-t-1)
            q[i-t-1] = q[i-t-1] - 1

    q.reverse()
    return print('Частное =', ''.join(str(i) for i in remove_zeros(q)), 'Остаток =', u)

a5(389725851, 79116, 10)
```

Рис. 5: Алгоритм 5 (деление многоразрядных целых чисел)

4 Выводы

В ходе выполнения данной лабораторной работы были реализованы алгоритмы для выполнения арифметических операций с большими целыми числами.