ЛАБОРАТОРНАЯ РАБОТА №8
Целочисленная арифметика многократной точности

В данной работе рассмотрим алгоритмы для выполнения арифметических операций с большими целыми числами. Будем считать, что число записано в *b*-ичной системе счисления, *b* — натуральное число, $b \geq 2$. Натуральное *b*-разрядное число будем записывать в виде:

$$\mathbf{u = u_1 u_2 \ldots u_n.}$$

Алгоритм 1 (сложение неотрицательных целых чисел)

Вход. Два неотрицательных числа $\mathbf{u = u_1u_2...u_n}$ и $\mathbf{v = v_1v_2...v_n}$; разрядность чисел *n*; основание системы счисления *b*.

Выход. Сумма $\mathbf{w = w_0w_1...w_n}$, где $\mathbf{w_0}$ — цифра переноса — всегда равная 0 либо 1.

1. Присвоить $\mathbf{j := n, k := 0}$ (j идет по разрядам, k следит за переносом).
2. Присвоить $\mathbf{w_j = (u_j + v_j + k) \ (mod\ b)}$, где $\mathbf{w_j}$ — наименьший неотрицательный вычет в данном классе вычетов;
   $\mathbf{k = \lfloor (u_j + v_j + k) / b \rfloor}$.
3. Присвоить $\mathbf{j := j - 1}$. Если $\mathbf{j > 0}$, то возвращаемся на шаг 2; если $\mathbf{j = 0}$, то присвоить $\mathbf{w_0 := k}$ и результат: **w**.

Алгоритм 2 (вычитание неотрицательных целых чисел)

Вход. Два неотрицательных числа $\mathbf{u = u_1 u_2 \ldots u_n}$ и $\mathbf{v = v_1 v_2 \ldots v_n}$, $\mathbf{u > v}$; разрядность чисел *n*; основание системы счисления *b*.

Выход. Разность $\mathbf{w = w_1 w_2 \ldots w_n = u - v}$.

1. Присвоить $\mathbf{j := n, k := 0}$ (k — заем из старшего разряда).
2. Присвоить $\mathbf{w_j = (u_j - v_j + k) \pmod b}$, где $\mathbf{w_j}$ — наименьший неотрицательный вычет в данном классе вычетов;
   $\mathbf{k = \lfloor (u_j - v_j + k) / b \rfloor}$.
3. Присвоить $\mathbf{j := j - 1}$. Если $\mathbf{j > 0}$, то возвращаемся на шаг 2; если $\mathbf{j = 0}$, то результат: $\mathbf{w}$.

Алгоритм 3 (умножение неотрицательных целых чисел столбиком)

Вход. Числа $\mathbf{u = u_1 u_2 \ldots u_n}$, $\mathbf{v = v_1 v_2 \ldots v_m}$; основание системы счисления b.

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

1. Выполнить присвоения:
   $\mathbf{w_{m+1} := 0, w_m := 0, \ldots, w_1 := 0}$, $\mathbf{j := m}$
   (j перемещается по номерам разрядов числа **v** от младших к старшим).
2. Если $\mathbf{v_j = 0}$, то присвоить $\mathbf{w_j := 0}$ и перейти на шаг 6.
3. Присвоить $\mathbf{i := n, k := 0}$ (Значение **i** идет по номерам разрядов числа **u**, **k** отвечает за перенос).
4. Присвоить
   $\mathbf{t := u_i \cdot v_j + w_{i+j} + k, w_{i+j} := t \ (mod\ b), k :=} \lfloor \mathbf{t / b} \rfloor$,
   где $\mathbf{w_{i+j}}$ — наименьший неотрицательный вычет в данном классе вычетов.
5. Присвоить $\mathbf{i := i - 1}$. Если $\mathbf{i > 0}$, то возвращаемся на шаг 4, иначе присвоить $\mathbf{w_j := k}$.
6. Присвоить $\mathbf{j := j - 1}$. Если $\mathbf{j > 0}$, то вернуться на шаг 2. Если $\mathbf{j = 0}$, то результат: **w**.

Алгоритм 4 (быстрый столбик)

Вход. Числа $\mathbf{u = u_1u_2\ldots u_n}$, $\mathbf{v = v_1v_2\ldots v_m}$; основание системы счисления *b*.
Выход. Произведение $\mathbf{w = uv = w_1w_2\ldots w_{n+m}}$.

1. Присвоить $\mathbf{t := 0}$.
2. Для **s от 0 до n + m − 1 с шагом 1** выполнить шаги 3 и 4.
3. Для $\mathbf{t = 0}$ до $\mathbf{n}$ выполнить присвоение
   $\mathbf{t := t + u_{n-s+t} \cdot v_{m-t+s}}$.
4. Вычислить
   $\mathbf{w_{s+1} := t \pmod b,\ t := \lfloor t / b \rfloor}$,
   где $\mathbf{w_{s+1}}$ — наименьший неотрицательный вычет по модулю *b*.
   Результат: **w**.

Алгоритм 5 (деление многоразрядных целых чисел)

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

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

1. Для $\mathbf{j}$ от 0 до $\mathbf{n - t}$ присвоить $\mathbf{q_j := 0}$.

2. Пока $\mathbf{u \geq v\, b^{n-t}}$, выполнять:
   $\mathbf{q_{n-t} := q_{n-t} + 1,\ u := u - v\, b^{n-t}}$.

3. Для $\mathbf{i := n, n - 1, \ldots, t + 1}$ выполнять пункты 3.1 – 3.4:
   3.1. Если $\mathbf{u_i \geq v_t}$, то присвоить
   $\mathbf{q_{i-t-1} := b - 1}$,
   иначе присвоить
   $\mathbf{q_{i-t-1} := \lfloor (u_i b + u_{i-1}) / v_t \rfloor}$.

   3.2. Пока
   $\mathbf{q_{i-t-1}(v_t b + v_{t-1}) > u_i b^2 + u_{i-1} b + u_{i-2}}$,
   выполнять
   $\mathbf{q_{i-t-1} := q_{i-t-1} - 1}$.

   3.3. Присвоить
   $\mathbf{u := u - q_{i-t-1} v\, b^{i-t-1}}$.

   3.4. Если $\mathbf{u < 0}$, то присвоить
   $\mathbf{u := u + v\, b^{i-t-1},\ q_{i-t-1} := q_{i-t-1} - 1}$.

4. $\mathbf{r := u}$.

Примеры результатов вычисления:

Сложение:

u = 123456789012345

v = 987654321098765

w = 1111111110111110

Вычитание:

u = 987654321098765

v = 123456789012345

w = 864197532086420

Умножение столбиком:

u = 123456789012345

v = 678901234567890

w = 83810234134616553086419753208605

Быстрое умножение:

u = 123456789012345

v = 678901234567890

w = 838102341346165530864197532086050

Деление:

u = 1000000000000000

v = 250000000000000

q = 4

r = 0

Деление (Дополнительный Пример):

u = 1234567890123456789012345

v = 123456789012345

q = 10000000001000000000

r = 12345

Выводы

В ходе данной лабораторной работы было рассмотрено 5 алгоритмов, обспечивающих более высокую производительность машинного сложения, вычитание, умножения и деления больших чисел. Данные алгоритмы были программно реализованы на языке ***Julia***. Данные программы могут быть использованы для работы с числами любой счетной системы (в частности десятичной) и превышающими размер стандартных типов данных.