

Лабораторная работа № 6 по курсу дискретного анализа: калькулятор

Выполнил студент группы 08-208 МАИ *Сизонов Артём*.

Условие

Кратко описывается задача:

1. Общая постановка задачи: Необходимо разработать программную библиотеку на языке C или C++, реализующую простейшие арифметические действия и проверку условий над целыми неотрицательными числами. На основании этой библиотеки, нужно составить программу, выполняющую вычисления над парами десятичных чисел и выводящую результат на стандартный файл вывода.

Метод решения

Выполнение программы состоит из трёх этапов: считывания входных данных, выполнения арифметической операции и вывода ответа.

Считывание входных данных. Входные данные считываются, как строки. Первые две входные строчки каждого теста преобразуются в длинные числа (класс TNumber). Внутреннее длинные числа представляют собой вектор чисел. Самый младший разряд числа находится в 0-м индексе вектора.

Выполнение арифметических операций.

1. Сложение. Сложение двух длинных чисел происходит самым очевидным образом. Идём с самых младших разрядов двух чисел и поразрядно складываем цифры в разрядах. Результат такого сложения, взятый по модулю системы счисления, записывается в соответствующий индекс ответа. Также стоит учитывать случаи, когда нужно прибавить единицу к следующему разряду.

2. Вычитание. Вычитание происходит также самым очевидным образом. Идём с самых младших разрядов. Вычитаем из цифры разряда уменьшаемого цифру разряда вычитаемого. Результат такого вычитания записывается в соответствующий индекс ответа. Также стоит учитывать случаи, когда нужно занять единицу у следующего разряда.

3. Умножение. Умножение также происходит в столбик. Для каждого i -го разряда первого числа проходим все разряды второго числа, умножая цифры в соответствующих разрядах. Результат, взятый по модулю системы исчисления, прибавляем к $(i + j)$ -му индексу ответа. Учитываем случаи, когда нужно прибавить некоторые числа к следующему разряду.

4. Возведение в степень. Используется бинарное возведение в степень, основанное на равенствах: $a^n = (a^2)^{n/2}$, если n – чётное и $a^n = a \cdot (a^2)^{n/2}$, если n – нечётное.

5. Деление. В первую очередь нормализуем заданные числа. Пусть n – длина делителя, а $n + m$ – длина делимого. Начиная со старших разрядов, делим число из $n + 1$ старших разрядов (u_p) на делитель (v). Подбираем максимальное число q_s , для которого: $u_p - q_s \cdot v \geq 0$. Вычитаем из числа u_p число $q_s \cdot v$. Заменяем старшие индексы делимого на u_p . Записываем q_s в соответствующий индекс ответа. Сдвигаемся на один разряд.

6. Сравнение. Числа сравниваются, пробегаая от меньшего разряда к большему.

Описание программы

Программа написана одним файлом.

TNumber – класс, являющийся представлением длинного числа.

Для класса TNumber переопределены операции сложения, умножения, вычитания, деления, возведения в степень и сравнения.

`void PrintNumber(TNumber&)` – функция, выводящая длинное число.

`void ReadNumber(TNumber&)` – функция, считывающая длинное число.

Дневник отладки

Посылка 1: Status: Compilation error. Решение: Исправлены ошибки компиляции: неиспользуемые переменные.

Посылка 2: Status: Time limit exceeded at test 10pow.t. Проблема: В реализации возведения в степень делались лишние копирования.

Посылка 3: Status: Time limit exceeded at test 10pow.t. Проблема: В реализации возведения в степень делались лишние копирования. Доработки не исправили ошибку

Посылка 4: Status: Wrong answer at test 2plus.t. Проблема: Допущена ошибка в алгоритме сложения (выход за пределы вектора).

Посылка 5: Status: Time limit exceeded at test 10pow.t. Проблема: В реализации возведения в степень делались лишние копирования. Доработки не исправили ошибку

Посылка 6: Status: Wrong answer at test 2plus.t. Проблема: Допущена ошибка в алгоритме сложения (выход за пределы вектора). Доработки не исправили ошибку

Посылка 7: Status: Wrong answer at test 3minus.t. Проблема: Допущена ошибка в алгоритме вычитания (выход за пределы вектора).

Посылка 8: Status: Time limit exceeded at test 6pow.t. Проблема: В реализации возведения в степень делались лишние копирования. Доработки не исправили ошибку

Посылка 9: Status: Wrong answer at test 3minus.t. Проблема: Допущена ошибка в алгоритме вычитания (выход за пределы вектора). Доработки не исправили ошибку

Посылка 10: Status: Wrong answer at test 3minus.t. Проблема: Допущена ошибка в алгоритме вычитания (выход за пределы вектора). Доработки не исправили ошибку

Посылка 11: Status: Wrong answer at test 5div.t. Проблема: Допущена ошибка в алгоритме деления (производились удаления лишних ведущих нулей).

Посылка 12: Status: Wrong answer at test 5div.t. Проблема: Допущена ошибка в алгоритме деления (производились удаления лишних ведущих нулей). Доработки не исправили ошибку

вили ошибку

Посылка 13: *Status:* Wrong answer at test 5div.t. Проблема: Допущена ошибка в алгоритме деления (производились удаления лишних ведущих нулей). Доработки не исправили ошибку

Посылка 14: *Status:* Wrong answer at test 5div.t. Проблема: Допущена ошибка в алгоритме деления (производились удаления лишних ведущих нулей). Доработки не исправили ошибку

Посылка 15: *Status:* Wrong answer at test 5div.t. Проблема: Допущена ошибка в алгоритме деления (производились удаления лишних ведущих нулей). Доработки не исправили ошибку

Посылка 16: *Status:* ОК.

Тест производительности

```
bsb@dell: ~/Рабочий стол/Информатика/Дискретный анализ/Лабораторные/LabN6 125x37
bsb@dell:~/Рабочий стол/Информатика/Дискретный анализ/Лабораторные/LabN6$ time cat plus_minus_100 | ./a.out > ans
real    0m0.117s
user    0m0.104s
sys     0m0.004s
bsb@dell:~/Рабочий стол/Информатика/Дискретный анализ/Лабораторные/LabN6$ time cat plus_minus_1000 | ./a.out > ans
real    0m0.575s
user    0m0.560s
sys     0m0.024s
bsb@dell:~/Рабочий стол/Информатика/Дискретный анализ/Лабораторные/LabN6$ time cat plus_minus_2000 | ./a.out > ans
real    0m1.099s
user    0m1.088s
sys     0m0.036s
bsb@dell:~/Рабочий стол/Информатика/Дискретный анализ/Лабораторные/LabN6$ time cat mul_1000 | ./a.out > ans
real    0m1.244s
user    0m1.236s
sys     0m0.016s
bsb@dell:~/Рабочий стол/Информатика/Дискретный анализ/Лабораторные/LabN6$ time cat mul_2000 | ./a.out > ans
real    0m2.990s
user    0m2.964s
sys     0m0.052s
bsb@dell:~/Рабочий стол/Информатика/Дискретный анализ/Лабораторные/LabN6$ time cat mul_4000 | ./a.out > ans
real    0m8.369s
user    0m8.296s
sys     0m0.092s
bsb@dell:~/Рабочий стол/Информатика/Дискретный анализ/Лабораторные/LabN6$ python3 full_tester.py > mul_8000
bsb@dell:~/Рабочий стол/Информатика/Дискретный анализ/Лабораторные/LabN6$ time cat mul_8000 | ./a.out > ans
real    0m25.907s
user    0m25.860s
sys     0m0.168s
bsb@dell:~/Рабочий стол/Информатика/Дискретный анализ/Лабораторные/LabN6$
```

Недочёты

Выявленных недочётов нет.

Выводы

Длинная арифметика применяется в случаях, когда необходимо производить вычислительные операции над числами, разрядность которых превышает длину машинного слова вычислительной машины. Такие вычислительные операции реализуются программно, а длинное число представляется в виде вектора чисел. Пусть n – длина числа, являющегося левым операндом, а m – длина числа, являющегося правым операндом. Тогда сложности: сложения – $O(\max(n, m))$, вычитания – $O(\max(n, m))$, умножения – $O(n \cdot m)$, деления – $O((n - m) \cdot m)$, сравнения – $O(n)$. Встроенные библиотеки для работы с длинными числами присутствуют в таких языках программирования, как Ruby, Java, Python.