

# Лабораторная работа № 7 по курсу Дискретный Анализ. Динамическое программирование

Выполнил студент группы М8О-307Б-21 МАИ *Друхольский Александр*.

## Условие

Задача:

Задано целое число  $n$ . Необходимо найти количество натуральных (без нуля) чисел, которые меньше  $n$  по значению и меньше  $n$  лексикографически (если сравнивать два числа как строки), а так же делятся на  $m$  без остатка.

## Метод решения

Для решения задачи необходимо подумать, как её можно разбить на более мелкие задачи. В этом нам поможет условие, которое говорит, что числа должны быть меньше исходного лексикографически. В таком случае мы можем свести задачу к просчёту количества таких чисел для числа, являющегося префиксом исходного числа. Таким образом

1. Считаем числа с 0 по  $i$  позицию ( $\text{Calculate}(i)$ ). Это будет  $\text{DP}(i) = \text{DP}(i-1) + \text{Calculate}(i)$ .
2. Для числа с 0 по  $i+1$  позицию решением будет  $\text{DP}(i) + \text{Calculate}(i+1)$  и так далее

Для каждого префикса ( $\text{upEdge}$ ), например 4-х значного, будем использовать число - степень десятки ( $\text{downEdge}$ ) (например 1000). Находим ближайшие числа к  $\text{upEdge}$ ,  $\text{downEdge}$  кратные  $m$ . За константу высчитываем количество чисел, удовлетворяющих условию и переходим к следующему префиксу. В итоге, учитывая, что искать количество чисел для каждого префикса мы будем за  $O(1)$ , общая сложность алгоритма  $O(s)$ , где  $s = \text{length}(n)$ , или  $O(\lg(n))$ . Но учитывая, что число  $n$  ограничено, максимальная длина числа - 19 символов, то алгоритм для любого числа сработает максимум за 19 итераций.

## Описание программы

Основные моменты:

1. *unsigned long long DP(unsigned long long n, long long m, long long len)* - основная функция. Вызывает DP для следующего префикса и прибавляет к получившемуся числу значение функции *CountInRange*.
2. *unsigned long long CountInRange(unsigned long long upEdge, long long m, unsigned long long downEdge)* - высчитывает количество чисел, которые делятся на  $m$  в промежутке.

## Дневник отладки

WA - 1 Отправил неверную версию кода с лишними выводами.

WA - 7 Ошибка при работе с большими числами

WA - 18,20 Ошибка в логике алгоритма. Иногда верхняя граница при подсчёте в диапазоне была меньше нижней.

## Тест производительности

Так как время работы алгоритма зависит только от длины первого числа, то произведём тесты для разных длин этого числа.

№	Длина числа	Время, с
1	1	1.4e-05
2	2	2.3e-05
3	3	1.9e-05
4	7	2.4e-05
5	10	2.1e-05
6	14	1.8e-05
7	16	2.5e-05
8	18	2.4e-05

Время для разных чисел не отличается настолько, что мы могли это определить, так как разница во времени выполнения одной или 18 итераций

## Выводы

Я смог разработать алгоритм для решения задачи методом динамического программирования. Наивным методом это задача решалась бы значительно дольше по времени.