15 FNV

Задача. Функции хэширования FNV (`http://isthe.com/chongo/tech/comp/fnv/`) обрабатывает сообщение `msg` из `size` октетов следующим образом:

```
uint32 fnv32(const uint8* msg, size_t size)
{
  uint32 hash = 2166136261;
  while (size--)
    hash ^= *msg++,
    hash *= 16777619;
  return hash;
}
```

Найдите сообщение `msg`, которое имеет нулевое хэш-значение.

Решение. Конечно, задачу можно решить «в лоб» путем перебора порядка 2^{32} сообщений. Предложим менее тривиальное решение, которое можно применить к функциям семейства FNV с длинами хэш-значений 64, 128, 256, 512 и 1024. Решение «в лоб» на таких длинах уже не пройдет.

Мультипликативные операции функции FNV ведутся в кольце $\mathbb{Z}_{2^{32}} = \{0, 1, \ldots, 2^{32} - 1\}$. Мультипликативная группа $\mathbb{Z}^*_{2^{32}}$ этого кольца состоит из нечетных чисел и имеет порядок 2^{31}. По теореме Лагранжа все элементы $a \in \mathbb{Z}^*_{2^{32}}$ имеют порядок, который делит 2^{31}. Напомним, что порядок a — это минимальное натуральное e такое, что $a^e \equiv 1 \pmod{2^{32}}$. Порядок можно найти, проверив значения e вида 2^i, $i = 0, 1, \ldots, 31$.

Множитель 16777619 функции FNV имеет порядок 2^{30} и половину элементов $a \in \mathbb{Z}^*_{2^{32}}$ можно представить как $16777619^x \bmod 2^{32}$. Числа 2166136261, 2166136263 так представить нельзя, а вот число 2166136265 можно:

$$2166136265 = 16777619^{695367386} \bmod 2^{32}.$$

Для нахождения искомого представления можно использовать логарифмирование по методу Поллига – Хэллмана. Логарифмирование будет быстрым, поскольку порядок основания 16777619 является степенью 2, т. е. не содержит больших простых делителей.

Рассмотрим обработку сообщения `msg` следующей структуры (умножения выполняются по модулю 2^{32}):

1. Первый октет равняется 12. После его обработки

$$\texttt{hash} = (2166136261 \oplus 12) \cdot 16777619 = 2166136265 \cdot 16777619.$$

2. Следущие $2^{30} - 695367386 - 1$ октетов нулевые. После их обработки

$$\begin{aligned}\texttt{hash} &= 2166136265 \cdot 16777619^{2^{30}-695367386} \\ &= 16777619^{695367386} \cdot 16777619^{2^{30}-695367386} \\ &= 16777619^{2^{30}} = 1.\end{aligned}$$

3. Последний октет равняется 1. После его обработки

$$\texttt{hash} = (1 \oplus 1) \cdot 16777619 = 0,$$

чего и требовалось добиться.

□