48 Биективный S-блок

Задача. Боб строит S-блок, который выполняет преобразование 16-разрядных слов с помощью следующей функции языка Си:

```
u16 S(u16 x) {
  static const u16 S0[256] = {...};
  static const u16 S1[256] = {...};
  register u32 y0 = S0[x & 255];
  register u32 y1 = S1[x >> 8];
  return (y0 + 1) * (y1 + 1) % 65537 - 1;
}
```

Найти заполнение массивов `S0` и `S1`, при котором S-блок будет биективным.

Решение. Использованная операция умножения — это операция мультипликативной группы $\mathbb{F}^*_{65537}$. Элементы группы кодируются числами от 0 до $2^{16}-1$. Перед умножением к коду добавляется единица, а после умножения она вычитается.

Число 3 — образующий $\mathbb{F}^*_{65537}$:

$$3^{65536/2} \not\equiv 1 \pmod{65537}.$$

Поэтому любой элемент группы можно представить в виде

$$3^{x_0+256x_1} = 3^{x_0} \cdot 3^{256x_1} \bmod 65537, \quad x_i = 0, 1, \ldots, 255.$$

В массив `S0` запишем значения $3^{x_0} \bmod 65537 - 1$, а в массив `S1` — значения $3^{256x_1} \bmod 65537 - 1$.

Массив `S0` не содержит числа 65535 (иначе $3^{x_0} \equiv 1 \bmod 65537 \Rightarrow \operatorname{ord} 3 \leq x_0$ и 3 не является образующим). Поэтому при умножении в последнем операторе функции переполнения не произойдет. □

Комментарий. Наш читатель hellman прислал подтвержающий решение скрипт системы компьютерной алгебры SAGE:

```
from sage.all import *
from itertools import product

R = IntegerModRing(65537)
g = R.multiplicative_generator() # ex. 3

S0 = []
S1 = []
for x in xrange(256):
    S0.append(int(g**x) - 1)
    S1.append(int(g**(256*x)) - 1)

S = []
for x, y in product(range(256), repeat=2):
    y0 = S0[x]
    y1 = S1[y]
    res = ((y0 + 1) * (y1 + 1)) % 65537 - 1;
    S.append(res)
print sorted(S) == range(2**16)
```