Project Euler #122: Efficient exponentiation

This problem is a programming version of Problem 122 from projecteuler.net

The most naive way of computing n^{15} requires fourteen multiplications:

$n \times n \times \cdots \times n = n^{15}$

But using a "binary" method you can compute it in six multiplications:

$n \times n = n^{2}$
$n^{2} \times n^{2} = n^{4}$
$n^{4} \times n^{4} = n^{8}$
$n^{8} \times n^{4} = n^{12}$
$n^{12} \times n^{2} = n^{14}$
$n^{14} \times n = n^{15}$

However it is yet possible to compute it in only five multiplications:

$n \times n = n^{2}$
$n^{2} \times n = n^{3}$
$n^{3} \times n^{3} = n^{6}$
$n^{6} \times n^{6} = n^{12}$
$n^{12} \times n^{3} = n^{15}$

We shall define $m(k)$ to be the minimum number of multiplications to compute n^k. For example $m(15) = 5$.

For a given k, compute $m(k)$, and also output the sequence of multiplications needed to compute n^k. See the sample output for more details.

Input Format

The first line of input contains T, the number of test cases.

Each test case consists of a single line containing a single integer, k.

Constraints

$1 \le T \le 500$
$2 \le k$

Input file #1: $k \le 111$.
Input file #2: $k \le 275$.

Output Format

For each test case, first output $m(k)$ in a single line. Then output $m(k)$ lines, each of the form `n^a * n^b = n^c`, where `a`, `b` and `c` are natural numbers. You may also output `n` instead of `n^1`. Use the `*` (asterisk/star) symbol, not the letter `x` or something else.

The sequence of multiplications must be valid. Any valid sequence will be accepted.

Sample Input

```
2
2
15
```

Sample Output

```
1
n^1 * n^1 = n^2
5
n * n = n^2
n^2 * n = n^3
n^3 * n^3 = n^6
n^6 * n^6 = n^12
n^12 * n^3 = n^15
```

Explanation

The second case, $k = 15$, is the example given in the problem statement.

The sample output illustrates that you can use `n` instead of `n^1`.