A young schoolboy would like to calculate the sum

$$S_k(n) = \sum_{i=1}^{n} i^k$$

for some fixed natural k and different natural n. He observed that calculating i^k for all i ($1 \le i \le n$) and summing up results is a too slow way to do it, because the number of required arithmetical operations increases as n increases. Fortunately, there is another method which takes only a constant number of operations regardless of n. It is possible to show that the sum $S_k(n)$ is equal to some polynomial of degree $k+1$ in the variable n with rational coefficients, i.e.,

$$S_k(n) = \frac{1}{M}\left(a_{k+1}n^{k+1} + a_k n^k + \ldots + a_1 n + a_0\right)$$

for some integer numbers $M, a_{k+1}, a_k, \ldots, a_1, a_0$.

We require that integer M be positive and as small as possible. Under this condition the entire set of such numbers (i.e. $M, a_{k+1}, a_k, \ldots, a_1, a_0$) will be unique for the given k. You have to write a program to find such set of coefficients to help the schoolboy make his calculations quicker.

Input

The input file contains several datasets, each of them containing a single integer k ($0 \le k \le 20$).

The first line of the input contains the number of datasets, and it's followed by a blank line. There's also a blank line between datasets.

Output

For each dataset, write integer numbers $M, a_{k+1}, a_k, \ldots, a_1, a_0$ to the output file in the given order. Numbers should be separated by one space. Remember that you should write the answer with the smallest positive M possible.

Print a blank line between consecutive outputs.

Sample input

```
1

2
```

Sample Output

```
6 2 3 1 0
```