Prime Factorization of n!

Filename: fact

Your boss has realized that patterns that emerge in the prime factorization of n! could be the key to accurately predicting the stock market. However, his background in mathematics is poor and he has asked you to write a program that will prime factorize n! so that together you can become rich!

Note that n! is the product of each positive integer in between 1 and n inclusive and that there are exactly 1000 prime numbers less than or equal to 7920.

The Problem:

Given a positive integer *n* (*2 ≤ n ≤ 7920)*, determine the prime factorization of *n!*.

The Input:

The first line of the input file will contain a single integer *t*, (*1 ≤ n ≤ 1000)*, the number of test cases in the file. The following *t* lines will each contain a single positive integer *n* (*2 ≤ n ≤ 7920)*.

The Output:

For each input case, on a single line, output the list of non-zero exponents to each of the primes, in increasing order of the prime numbers, with a space after each value. For example, 5! == $2^3 3^1 5^1$, so for this case the output should be

```
3 1 1
```

Note: there is a space after the last 1. This makes printing the output a bit easier.

Sample Input:

```
3
5
30
31
```

Sample Output:

```
3 1 1
26 14 7 4 2 2 1 1 1 1
26 14 7 4 2 2 1 1 1 1 1
```