Ram is a bright boy who is very much interested in number theory. He was studying about factorials of numbers, and got some interesting idea.

Being a brilliant coder, he started writing a program and implemented the following routines :

- $fact(n)$ — This function returns the value of $n!$, where $n \geq 0$

  eg. fact(5) returns 120

- $count(n)$ — This function returns the number of terms in the prime factorisation of n, where $n \geq 0$.

  eg. count(24) returns 4 (because, $24 = 2 * 2 * 2 * 3$). The prime factorisation of 24 contains 4 terms

- $func(n)$ — This function is explained below.

Ram wrote the function "`func`" as follows:

```
int func(int $n$)
{
int ans = 0;
for(int $i=0$; ; $i++$)
{
if( count( fact( $i$ ) ) $\le n$)
ans++;
else
return ans;
}
}
```

The above procedure takes too much time to execute. Help Ram by writing a more efficient solution that does the same function as "`func`" does.

Input

The first line of input gives the number of test cases t.

The next t lines contains a positive integer, representing n ($1 \leq t \leq 1000$, $1 \leq n \leq 10000000$).

Output

Print $func(n)$ for the given n, on a line by itself.

Note: Consider 1 as a prime number.

Sample Input

```
4
1
2
3
4
```

Sample Output

```
3
4
4
5
```