Just Primes

This problem is really simple, or is it? Given a positive integer **N**, calculate the **minimum** number of **distinct** primes needed such that their sum equals to **N**. A prime number is a natural number greater than **1** that cannot be formed by multiplying two smaller natural numbers. The first few prime numbers are **2, 3, 5, 7, 11, 13, 17, 19, 23, 29, ...** and so on.

Input

The first line contains an integer **T**, denoting the number of test cases. Each of the next subsequent **T** lines contain a positive integer **N**.

Constraints

- **1 ≤ T ≤ 50,000**
- **1 ≤ N ≤ 50,000**

Output

For each test case, first print the case number followed by the minimum number of distinct primes such that their sum equals to **N**. If **N** cannot be represented by a summation of distinct primes, then print the case number followed by **-1**. Refer to the sample input/output for more clarity of the format.

Sample Input

```
10
1
2
3
10
27
100
1000
4572
4991
49999
```

Sample Output

```
Case 1: -1
Case 2: 1
Case 3: 1
Case 4: 2
Case 5: 3
Case 6: 2
Case 7: 2
Case 8: 2
Case 9: 3
```

Case 10: 1

Challenge

Too easy? Try the harder version here - Just Primes II