Bases Conversion (23)

Your professor has just introduced you to the different bases in which a number can be written and wants to test your understanding. His favourite number is 23 (this should not surprise you too much... it is a prime number!) and for this reason his favourite bases are 2 and 3.

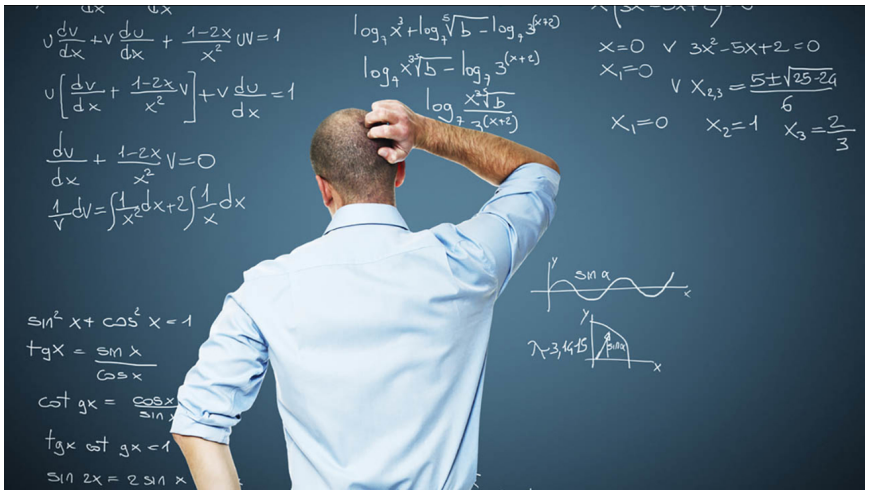

Figure 1: A professor figuring out how to keep students busy.

A number is called **special** if and only if the sum of the digits of its base *two* representation is the same as the sum of the digits of its base *three* representation.

Consider the number $6_{(10)}$ (the subscript denotes the basis) $= 110_{(2)} = 20_{(3)}$. The sum of the digits in base two is $1 + 1 + 0 = 2$, which is exactly the sum of the digits in base three: $2 + 0 = 2$. Thus, number $6_{(10)}$ is special. On the contrary, the number $9_{(10)} = 1001_{(2)} = 100_{(3)}$ is not special (the sums of digits are 2 and 1, respectively).

In order to keep the class busy without much work on his side, your professor has come up with a boring homework assignment. He has given you a list of T numbers $N_1, \ldots, N_T$ and for each of those he wants you to compute the number of special numbers from 1 to N_i (extremes included). Automate this tedious task by writing a program!

☞ Among the attachments of this task you may find a template file `23.*` with a sample incomplete implementation.

Input

The first line contains the integer T. The second line contains T integers N_i.

Output

You need to write a single line with T integers, where the i-th indicates the amount of special numbers in the range between 1 and N_i.

Constraints

- $1 \leq T \leq 10\,000$.
- $1 \leq N_i \leq 100\,000\,000$ for each $i = 0 \ldots T - 1$.

Scoring

Your program will be tested against several test cases grouped in subtasks. In order to obtain the score of a subtask, your program needs to correctly solve all of its test cases.

– **Subtask 1** (0 points) Examples.

– **Subtask 2** (30 points) The sum of all N_i does not exceed $1\,000\,000$.

– **Subtask 3** (35 points) $N_i \leq 1\,000\,000$ for each $i = 0 \ldots T - 1$.

– **Subtask 4** (35 points) No additional limitations.

Examples

input	output
4 1 5 6 2	1 1 2 1
1 10	4

Explanation

The representations in base two and three of the numbers between 1 and 10 are contained in the following table. Special numbers have been highlighted in yellow.

Base 10	Base 2	Base 3	Sum in base 2	Sum in base 3
1	1	1	1	1
2	10	2	1	2
3	11	10	2	1
4	100	11	1	2
5	101	12	2	3
6	110	20	2	2
7	111	21	3	3
8	1000	22	1	4
9	1001	100	2	1
10	1010	101	2	2

In the **first sample case** for the ranges up to 1, 2 and 5 there is only a single special number ($1_{(10)}$). If we consider numbers up to 6, there are two special numbers ($1_{(10)}$ and $6_{(10)}$).

In the **second sample case**, in the range up to $10_{(10)}$ there are four special numbers (the ones highlighted in the table above).