aaaaaaaaaA heH heH nuN

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

Vasily Tadokorov is a stringologist. He thinks a string is fragrant if it can be divided into two parts: `nunhehheh` as the prefix and a number of (excluding 0) `a` as the suffix. For example, `nunhehhehaaaaaa` is fragrant, but `nunhehheh` and `nunhehhehoooaaa` are not fragrant.

Today Vasily Tadokorov has some strings consisting of lowercase English letters. For each string, he wants to know how many subsequences of this string are fragrant. A string a is a subsequence of a string b if a can be obtained from b by deletion of several (including 0) characters.

The above is a problem of string that Vasily came up with. As we know, a problem usually has several examples for better understanding. However, Vasily gets buried in making some fragrant examples. After 2000 years, he finally makes two perfect examples as follows.

Example 1:

- Input: `nunhehhehahaahahahahahahaahaahahahahha`
- Output: 114514

Example 2:

- Input: `nunhehhehhehhahaahahahaahaahahaaaahaa`
- Output: 1919810

Vasily is not clever enough. He doesn't want to work for another 2000 years, so he asks you for help. He gives you T tasks, each of which contains an integer n, and you should construct a string consisting of only lowercase English letters that has **exactly** n fragrant subsequences.

Input

The first line contains an integer T ($1 \leq T \leq 1000$), denoting the number of tasks.

Each of the next T lines contains an integer n ($0 \leq n \leq 10^9$).

Output

For each test case, output one string consisting of only lowercase English letters in a single line indicating the answer. You need to ensure that the sum of the length over all the output strings does not exceed 10^6. It can be proved that a solution always exists. If there are multiple solutions, print any.

Example

standard input	standard output
2 114514 1919810	nunhehhehahaahahahahahahaahaahahahahha nunhehhehhehhahaahahahaahaahahaaaahaa