Once upon a time, there was an NWERC judge with a tendency to create slightly too hard problems. As a result, his problems were never solved. As you can image, this made our judge somewhat frustrated. This year, this frustration has culminated, and he has decided that rather than spending a lot of time constructing a well-crafted problem, he will simply write some insanely hard problem statement and just generate some random input and output les. After all, why bother having proper test data if nobody is going to try the problem anyway?

Thus, the judge generates a testcase by simply letting the input be a random number, and letting the output be another random number. Formally, to generate the data set with T test cases, the judge generates $2T$ random numbers $x_1, \ldots, x_{2T}$ between 0 and 10 000, and then writes T, followed by the sequence $x_1, x_3, x_5, \ldots, x_{2T-1}$ to the input le, and the sequence $x_2, x_4, x_6, \ldots, x_{2T}$ to the output le.

The random number generator the judge uses is quite simple. He picks three numbers x_1, a, and b between 0 and 10 000 (inclusive), and then for i from 2 to $2T$ lets $x_i = (a \cdot x_{i-1} + b) \bmod 10001$.

You may have thought that such a poorly designed problem would not be used in a contest of such high standards as NWERC. Well, you were wrong.

Input

On the rst line one positive number: the number of testcases, at most 100. After that per testcase:

- One line containing an integer n ($0 \leq n \leq 10000$): an input testcase.

The input le is guaranteed to be generated by the process described above.

Output

Per testcase:

- One line with an integer giving the answer for the testcase.

If there is more than one output le consistent with the input le, any one of these is acceptable.

Sample Input

```
3
17
822
3014
```

Sample Output

```
9727
1918
4110
```