

Kawigi quote

Thursday 25/7/2013, a friend of mine ([Goodname](#)) posted in our facebook group (Virtual Contests) a link of [Kawigi](#)'s topcoder profile.

He was amazed by Kawigi's quote as he tested it in [Ideone](#) to find it printing "C++Sucks". We all were amazed by this quote.

But in a few minutes some other friend of mine ([Mosa](#)) commented on the post with this [Ideone](#). Wow! he got it, but the fact that he is kind of greedy, he didn't want to tell us the secret. Now we want to think like Mosa to be able to solve this problem.

Given a string your task is to find the values of `m[0]` and `m[1]` so that when we run this code we get the given string.

Code

```
#include <stdio.h>

int primes [] = {2,3,5,7,11};

double m[2] = {8242465576917890.0 ,494};

int main() {
    m[1]--?m[0]/=primes[((int)m[1]+1)%5],main():printf((char*)m);
}

// this code prints the string "123".
```

Input

The first line of the input file contains an integer `T` ($T \leq 50$) which is the number of test cases to follow.

Each test case will be on a single line consisting of at most 6 alphanumeric characters (A-Za-z0-9).

Output

For each test case, print the values of `m[0]` and `m[1]`, separated by a space.

It's guaranteed that the solution will always exist. If there're multiple solutions, print any.

Example

Input:

```
4
123
abdou
mcMosa
kimo3
```

Output:

8242465576917890.0 494
6966198460426549.0 487
4484456159639948.0 483
21357815651012032.0 488

Special Thanks:

-[Mitch Schwartz](#) for discovering issues with the problem.

-[Mosa Osama](#) for writing the special judge of the problem, and I have to say Mosa's not greedy. ;)