Problem 2. Polynomial evaluation

Given a polynomial $P(x)$ of degree n

$$P(x) = a_0 + a_1 x + \cdots + a_n x^n = \sum_{i=0}^{n} a_i x^i,$$

we want to evaluate the polynomial at several points $x_0, x_1, \cdots, x_{m-1}$.

Input format

On the first line, a nonnegative integer n, which is the degree of the polynomial.

On the second line, $n+1$ numbers $a_0, a_1, \cdots, a_n$ separated by space, which are the coefficients of the polynomial.

On the third line, a nonnegative integer m.

Then m lines follow, the i-th of which is a number $x_i (i = 0, 1, \cdots, m-1)$.

Output format

m lines, the i-th of which ($i = 0, 1, \cdots, m-1$) is a number $P(x_i)$, **rounded to three decimal places**.

Example

Input

```
2
-0.5 1 2.5
5
0
-6.6
1000
-1
32
```

Output

```
-0.500
101.800
2500999.500
1.000
2591.500
```

Notes

It is guaranteed that $n \leqslant 30$. An array is enough to store the coefficients. Do not use heap memory.

Your program will not be compiled and linked against the math library, so do not use the functions in `<math.h>`.

The evaluation of $P\left(x_i\right)$ at a given point x_i should be done using only one loop without call to standard library functions. Think about how to do this efficiently.