2278 Find a number

S is a set of non-negative integers, such that an integer may occur more that once in S, and k is a positive integer. Let m be the largest possible integer, not larger than k, and m is equal to the sum of the members of some subset of S.

The problem of finding m, by given S and k, is a hard one.

Your task will be easier. Your program should find and output an integer p that is within 10% of m. Formally written p should satisfy the inequality: $m * (0.9) < p \leq m$.

Input

Your program should read sets of data from a text file. The first line of the input file contains the number of the data sets. Each data set is presented in the following format:

n
$S[1]$ $S[2]$ $S[3]$ ... $S[n]$
k

The first line contains one positive integer n ($1 \leq n \leq 500$) representing the number of elements in S. The second line contains n non-negative integers, each two consecutive separated by one blank. The third line contains the integer k ($1 \leq k \leq 2000000000$).

There are no blank lines between consecutive sets of data. Input data are correct.

Output

The result of the program is printed on the standard output. For each input data set the program prints a single line containing a single integer p, such that $m * (0.9) < p \leq m$. There should not be any leading blanks at the start of the line.

Sample Input

```
2
5
10 4 5 6 4
24
4
3 8 2 11
10
```

Sample Output

```
22
10
```