Double Hashing

Susi heard that prime numbers are a good choice if one searches for adequate sizes of hash tables. Especially if you use double hashing. I.e. a value *a* is hashed to the address *a mod m*, whereas *m* is the size of the hash table. If the address *a mod m* is already occupied one tries the addresses *(a +1*r) mod m*, *(a +2*r) mod m*, and so on. Susi wants to verify the assumption that prime numbers are a good choice for the size of the table. She created a couple of testcases and now it's you task to help her with the analysis of these testcases.

Input

The first line contains the number of testcases. Each testcase consists of the three numbers *m, n* and *r*, whereas $m \leq 1000$ and $r,n<m$. On the next line the *n* values follow.

Output

For each testcase output two lines. The first line contains the number of probes for a hash table of size *m* and the second line the number of probes for a hash table of size *p*. Whereas *p* is the smallest prime number greater or equal to *m*. For the format see the sample. Print a blank line after each testcase.

Example

Input:
2
10 8 4
400 20 53 64 103 54 325 23
8 7 4
16 32 64 128 256 1024 2048

Output:
7 probe(s) made with m=10
6 probe(s) made with p=11

unable to insert elements
0 probe(s) made with p=11