

Problem A: MODEX

Many well-known cryptographic operations require modular exponentiation. That is, given integers x , y and n , compute $x^y \bmod n$. In this question, you are tasked to program an efficient way to execute this calculation.

Input

The input consists of a line containing the number c of datasets, followed by c datasets, followed by a line containing the number 0.

Each dataset consists of a single line containing three positive integers, x , y , and n , separated by blanks. You can assume that $1 < x, n < 2^{15} = 32768$, and $0 < y < 2^{31} = 2147483648$.

Output

The output consists of one line for each dataset. The i^{th} line contains a single positive integer z such that

$$z = x^y \bmod n$$

for the numbers x , y , z given in the i^{th} input dataset.

Example

Sample Input	Sample Output
3	3
2 3 5	1
2 10 11	2
3 8 7	
0	