**Special equations**

**Time Limit: 2000/1000 MS (Java/Others)    Memory Limit: 32768/32768 K (Java/Others)  
TSpecial Judge**

**Problem Description**

　　Let f(x) = anxn +...+ a1x +a0, in which ai (0 <= i <= n) are all known integers. We call f(x) 0 (mod m) congruence equation. If m is a composite, we can factor m into powers of primes and solve every such single equation after which we merge them using the Chinese Reminder Theorem. In this problem, you are asked to solve a much simpler version of such equations, with m to be prime's square.

**Input**

　　The first line is the number of equations T, T<=50.  
　　Then comes T lines, each line starts with an integer deg (1<=deg<=4), meaning that f(x)'s degree is deg. Then follows deg integers, representing an to a0 (0 < abs(an) <= 100; abs(ai) <= 10000 when deg >= 3, otherwise abs(ai) <= 100000000, i<n). The last integer is prime pri (pri<=10000).   
　　Remember, your task is to solve f(x) 0 (mod pri\*pri)

**Output**

　　For each equation f(x) 0 (mod pri\*pri), first output the case number, then output anyone of x if there are many x fitting the equation, else output "No solution!"

**Sample Input**

4

2 1 1 -5 7

1 5 -2995 9929

2 1 -96255532 8930 9811

4 14 5458 7754 4946 -2210 9601

**Sample Output**

Case #1: No solution!

Case #2: 599

Case #3: 96255626

Case #4: No solution!