**Java Loops II**

<https://www.hackerrank.com/challenges/java-loops/problem>

We use the integers *a*, *b*, and *n* to create the following series:

(a + 20 . b), (a + 20 . b + 21 . b), . . . , (a + 20 . b + 21 . b + . . . + 2n-1 . b)

You are given *q* queries in the form of *a*, *b*, and *n*. For each query, print the series corresponding to the given *a*, *b*, and *n* values as a single line of *n* space-separated integers.

**Input Format**

The first line contains an integer, *q*, denoting the number of queries.  
Each line *i* of the *q* subsequent lines contains three space-separated integers describing the respective *ai*, *bi*, and *ni* values for that query.

**Constraints**

* *0 <= q <= 500*
* *0 <= a, b <= 50*
* *1 <= n <= 15*

**Output Format**

For each query, print the corresponding series on a new line. Each series must be printed in order as a single line of *n* space-separated integers.

**Sample Input**

2

0 2 10

5 3 5

**Sample Output**

2 6 14 30 62 126 254 510 1022 2046

8 14 26 50 98

**Explanation**

We have two queries:

1. We use *a = 0*, *b = 2*, and *n = 10* to produce some series s0, s1, . . . , sn-1:

* s0 = 0 + 1 . 2 = 2
* s1 = 0 + 1 . 2 + 2 . 2 = 6
* s2 = 0 + 1 . 2 + 2 . 2 + 4 . 2 = 14

... and so on.

Once we hit *n = 10*, we print the first ten terms as a single line of space-separated integers.

1. We use *a = 5*, *b = 3*, and *n = 5* to produce some series s0, s1, . . . , sn-1:

* s0 = 5 + 1 . 3 = 8
* s1 = 5 + 1 . 3 + 2 . 3 = 14
* s2 = 5 + 1 . 3 + 2 . 3 + 4 . 3 = 26
* s3 = 5 + 1 . 3 + 2 . 3 + 4 . 3 + 8 . 3 = 50
* s4 = 5 + 1 . 3 + 2 . 3 + 4 . 3 + 8 . 3 + 16 . 3 = 98

... and so on.

We then print each element of our series as a single line of space-separated values.