C++ Variadics

Create a template function named *reversed_binary_value*. It must take an arbitrary number of *bool* values as template parameters. These booleans represent binary digits in reverse order. Your function must return an integer corresponding to the binary value of the digits represented by the booleans. For example: *reversed_binary_value<0,0,1>()* should return 4.

Input Format

The first line contains an integer, T, the number of test cases. Each of the T subsequent lines contains a test case. A test case is described as 2 space-separated integers, X and Y, respectively.

- X is the value to compare against.
- Y represents the range to compare: $64 \times Y$ to $64 \times Y + 63$.

Constraints

- $0 \leq X \leq 65535$
- $0 \leq Y \leq 1023$
- The number of template parameters passed to *reversed_binary_value* will be ≤ 16.

Output Format

Each line of output contains 64 binary characters (i.e., 0's and 1's). Each character represents one value in the range. The *first* character corresponds to the *first* value in the range. The *last* character corresponds to the *last* value in the range. The character is 1 if the value in the range matches X; otherwise, the character is 0.

Sample Input

```
2
65 1
10 0
```

Sample Output

```
0100000000000000000000000000000000000000000000000000000000000000
0000000000100000000000000000000000000000000000000000000000000000
```

Explanation

The second character on the first line is a 1, because the second value in the range $64..127$ is 65 and X is 65.

The eleventh character on the second line is a 1, because the eleventh value in the range $0..63$ is 10 and X is 10.

All other characters are zero, because the corresponding values in the range do not match X.