A. Bits

time limit per test

1 second

memory limit per test

256 megabytes

input

standard input

output

standard output

Let's denote as https://espresso.codeforces.com/6500127ba81ee2a9ffdacf23b301df518aa9cd65.png the number of bits set ('1' bits) in the binary representation of the non-negative integer *x*.

You are given multiple queries consisting of pairs of integers *l* and *r*. For each query, find the *x*, such that *l* ≤ *x* ≤ *r*, and https://espresso.codeforces.com/6500127ba81ee2a9ffdacf23b301df518aa9cd65.png is maximum possible. If there are multiple such numbers find the smallest of them.

**Input**

The first line contains integer *n* — the number of queries (1 ≤ *n* ≤ 10000).

Each of the following *n* lines contain two integers *li*, *ri* — the arguments for the corresponding query (0 ≤ *li* ≤ *ri* ≤ 1018).

**Output**

For each query print the answer in a separate line.

**Examples**

**input**

**Copy**

3  
1 2  
2 4  
1 10

**output**

**Copy**

1  
3  
7

**Note**

The binary representations of numbers from 1 to 10 are listed below:

110 = 12

210 = 102

310 = 112

410 = 1002

510 = 1012

610 = 1102

710 = 1112

810 = 10002

910 = 10012

1010 = 10102

TUTORIAL

Let us define function *f*(*L*, *R*), that gives answer to the query. It looks follows:

* if *L* = *R* then *f*(*L*, *R*) = *L*;
* else if 2*b* ≤ *L*, where *b* — maximum integer such 2*b* ≤ *R*, then *f*(*L*, *R*) = *f*(*L* - 2*b*, *R* - 2*b*) + 2*b*;
* else if 2*b*+ 1 - 1 ≤ *R* then *f*(*L*, *R*) = 2*b*+ 1 - 1;
* else *f*(*L*, *R*) = 2*b* - 1.

Total complexity is *O*(*logR*) per query.

CODE:

#include<bits/stdc++.h>

#define int long long int

using namespace std;

int f(int l,int r)

{

int a=1;

while(a<<1 <= r)

a<<=1;

if(l==r)

return l;

else if(l>=a)

return f(l-a,r-a)+a;

else if((a<<1)-1<=r)

return (a<<1)-1;

else

return a-1;

}

main()

{

int t;

cin>>t;

while(t--)

{

int l,r;

cin>>l>>r;

cout<<f(l,r)<<"\n";

}

}