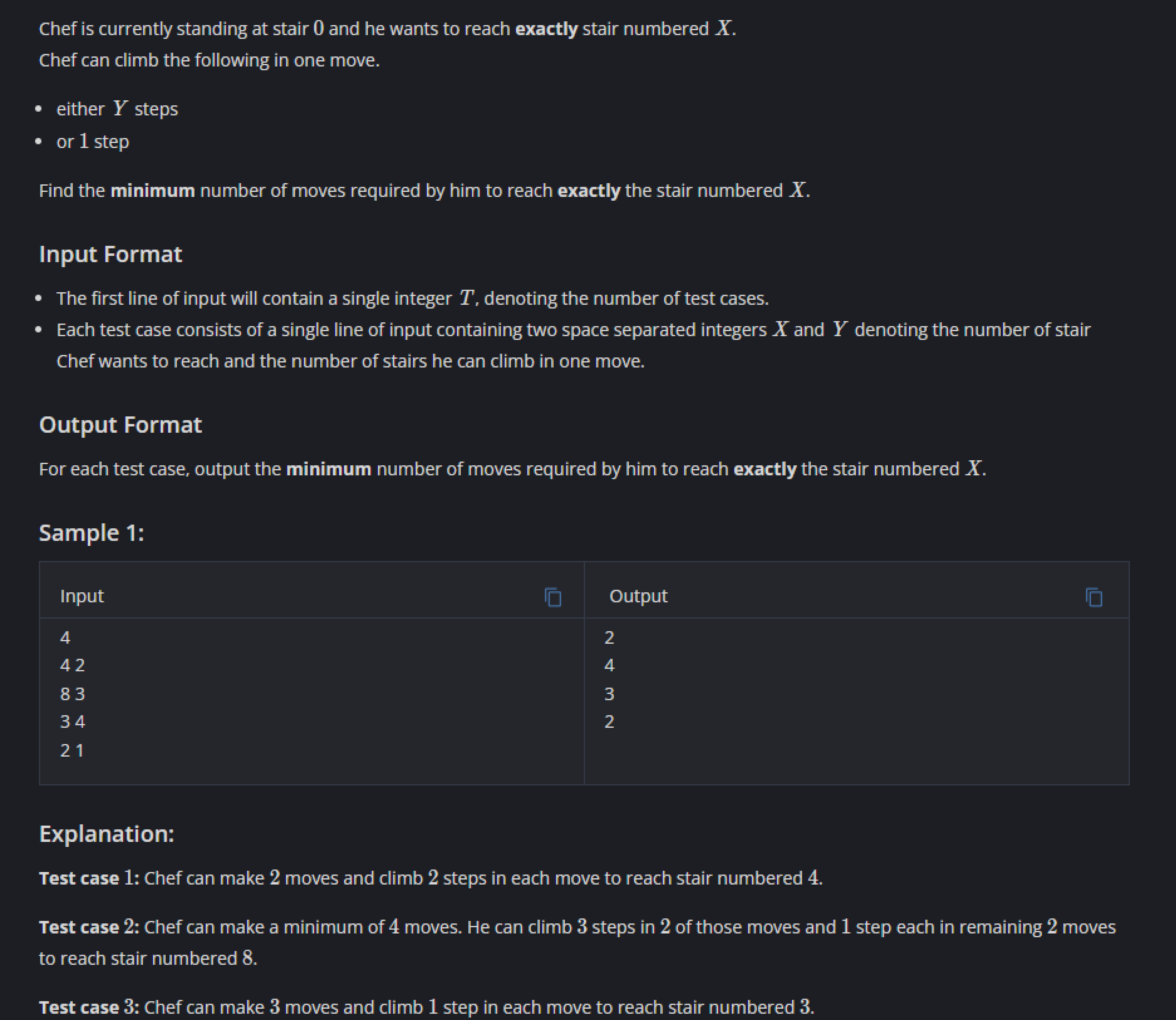


Chef is currently standing at stair 0 and he wants to reach **exactly** stair numbered X.
Chef can climb the following in one move.

- either Y steps
- or 1 step

Find the **minimum** number of moves required by him to reach **exactly** the stair numbered X.

Input Format

- The first line of input will contain a single integer T, denoting the number of test cases.
- Each test case consists of a single line of input containing two space separated integers X and Y denoting the number of stair Chef wants to reach and the number of stairs he can climb in one move.

Output Format

For each test case, output the **minimum** number of moves required by him to reach **exactly** the stair numbered X.

Sample 1:

Input	Output

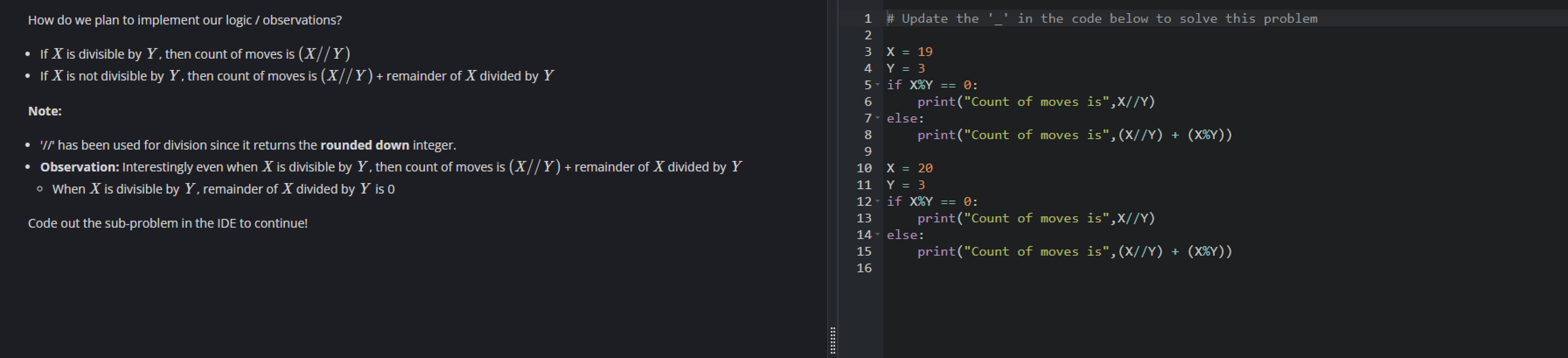
4
4 2
8 3
3 4
2 1
	2
	4
	3
	2

Explanation:

Test case 1: Chef can make 2 moves and climb 2 steps in each move to reach stair numbered 4.

Test case 2: Chef can make a minimum of 4 moves. He can climb 3 steps in 2 of those moves and 1 step each in remaining 2 moves to reach stair numbered 8.

Test case 3: Chef can make 3 moves and climb 1 step in each move to reach stair numbered 3.



How do we plan to implement our logic / observations?

- If X is divisible by Y, then count of moves is $(X//Y)$
- If X is not divisible by Y, then count of moves is $(X//Y)$ + remainder of X divided by Y

Note:

- '//' has been used for division since it returns the **rounded down** integer.
- **Observation:** Interestingly even when X is divisible by Y, then count of moves is $(X//Y)$ + remainder of X divided by Y
    - When X is divisible by Y, remainder of X divided by Y is 0

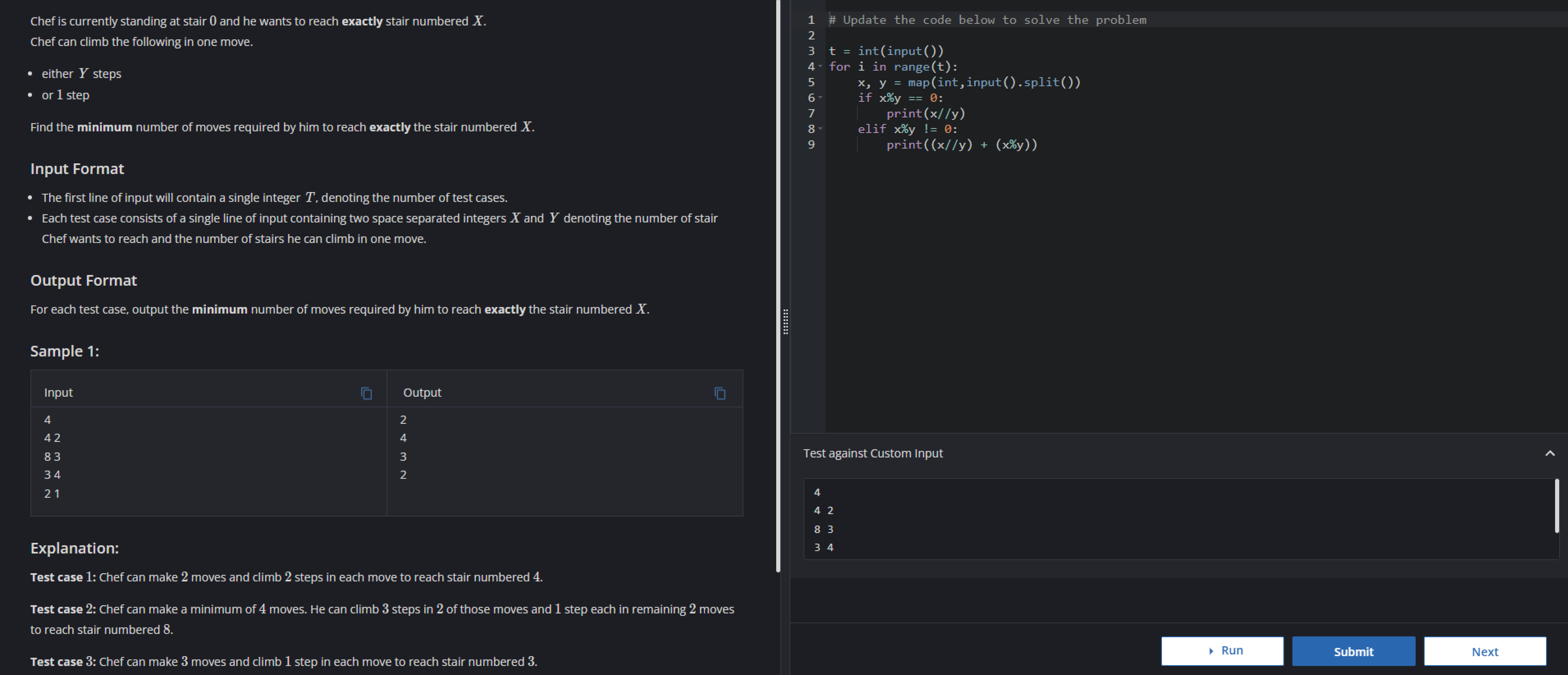
Code out the sub-problem in the IDE to continue!

```
# Update the '_' in the code below to solve this problem

X = 19
Y = 3
if X%Y == 0:
    print("Count of moves is",X//Y)
else:
    print("Count of moves is",(X//Y) + (X%Y))

X = 20
Y = 3
if X%Y == 0:
    print("Count of moves is",X//Y)
else:
    print("Count of moves is",(X//Y) + (X%Y))

```



Chef is currently standing at stair 0 and he wants to reach **exactly** stair numbered X.
Chef can climb the following in one move.

- either Y steps
- or 1 step

Find the **minimum** number of moves required by him to reach **exactly** the stair numbered X.

Input Format

- The first line of input will contain a single integer T, denoting the number of test cases.
- Each test case consists of a single line of input containing two space separated integers X and Y denoting the number of stair Chef wants to reach and the number of stairs he can climb in one move.

Output Format

For each test case, output the **minimum** number of moves required by him to reach **exactly** the stair numbered X.

Sample 1:

Input

```
4
4 2
8 3
3 4
2 1
```

Output

```
2
4
3
2
```

Explanation:

Test case 1**:** Chef can make 2 moves and climb 2 steps in each move to reach stair numbered 4.

Test case 2**:** Chef can make a minimum of 4 moves. He can climb 3 steps in 2 of those moves and 1 step each in remaining 2 moves to reach stair numbered 8.

Test case 3**:** Chef can make 3 moves and climb 1 step in each move to reach stair numbered 3.

```python
# Update the code below to solve the problem

t = int(input())
for i in range(t):
    x, y = map(int,input().split())
    if x%y == 0:
        print(x//y)
    elif x%y != 0:
        print((x//y) + (x%y))
```

Test against Custom Input

```
4
4 2
8 3
3 4
```