U - Floor Number

Vasya goes to visit his classmate Petya. Vasya knows that Petya's apartment number is n.

There is only one entrance in Petya's house and the distribution of apartments is the following: the first floor contains 2 apartments, every other floor contains x apartments each. Apartments are numbered starting from one, from the first floor. I.e. apartments on the first floor have numbers 1 and 2, apartments on the second floor have numbers from 3 to (x + 2), apartments on the third floor have numbers from (x + 3) to (2⋅*x*+2), and so on.

Your task is to find the number of floor on which Petya lives. Assume that the house is always high enough to fit at least n apartments.

You have to answer t independent test cases.

**Input**

The first line of the input contains one integer t(1≤*t*≤1000) — the number of test cases. Then t test cases follow.

The only line of the test case contains two integers n and x (1≤*n*,*x*≤1000) — the number of Petya's apartment and the number of apartments on each floor of the house except the first one (there are two apartments on the first floor).

**Output**

For each test case, print the answer: the number of floor on which Petya lives.

**Example**

**Input**

4

7 3

1 5

22 5

987 13

**Output**

3

1

5

77

**Note**

Consider the first test case of the example: the first floor contains apartments with numbers 1 and 2, the second one contains apartments with numbers 3, 4 and 5, the third one contains apartments with numbers 6, 7 and 8. Therefore, Petya lives on the third floor.

In the second test case of the example, Petya lives in the apartment 1 which is on the first floor.

#include <iostream>

#include <algorithm>

using namespace *std*;

int main(void)

{

int t; *cin* >> t;

while (t--)

{

int n, x; *cin* >> n >> x;

int i = 2;

if (n == 1 || n == 2) *cout* << 1 << *endl*;

else

{

while(true)

{

if ((((i - 2)\*x) + 3 <= n) && (((i - 1)\*x + 2) >= n))

{

*cout* << i << *endl*;

break;

}

i++;

}

}

}

return 0;

}