

ĐẾM SỐ LƯỢNG SỐ FIBONACCI.

Số fibonacci được định nghĩa như sau:

$$\begin{cases} f_0 = 0 \\ f_1 = 1 \\ f_n = f_{n-1} + f_{n-2} \text{ với mọi } n \geq 2 \end{cases}$$

Khi đó, đoạn đầu tiên của dãy fibonacci như sau: 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89.....

Để đếm tất cả các số fibonacci nhỏ hơn n , người ta thường code như sau:

```
#include <iostream>
#include <algorithm>
#include <cmath>
using namespace std;
long long a, b, c, n, dem;
int main()
{
    cin >> n;
    if(n == 0)
    {
        cout << 0; return 0;
    }
    else if(n == 1)
    {
        cout << 2; return 0;
    }
}
```

```
else
{
    dem = 2;
    a = 0, b = 1;
    while(a + b <= n)
    {
        c = a + b;
        a = b, b = c;
        dem++;
    }
    cout << dem;
}
```

Yêu cầu: Cho 2 số l, r . Hãy đếm số lượng số fibonacci trong đoạn $[l, r]$.

Dữ liệu: Một dòng gồm 2 số nguyên dương l, r ($l \leq r \leq 10^{18}$)

Kết quả: In ra số số fibonacci trong đoạn từ l đến r .

Ví dụ:

Sample Input	Sample Output
1 10	6