#include <cassert>

#include <iostream>

class Fibonacci\_1 {

public:

static int get(int n) {

assert(n >= 0);

int x=0,y=1,z=1;

for (int i=2; i<n; i++)

{x=y;

y=z;

z=x+y;

}

//printf("%d", z);

return z;}

};

class array {

public:

static int get\_remainder(long long n,int m) {

long long z;

int array[0], i;

for (i=1; i<n+1; i++)

{

z=Fibonacci\_1::get(i)%m;

array[i-1]=z;

if(z==0&&Fibonacci\_1::get(i+1)%m==1&&Fibonacci\_1::get(i+2)%m==1)

{printf("%d \n", array[i-1]); break;}

printf("%d \n", array[i-1]);

}

return 12%6;}

};

class Fibonacci {

public:

static int get\_remainder(long long n, int m) {

assert(n >= 1);

assert(m >= 2);

long long x=0,y=1,z=1;

for (int i=2; i<n; i++)

{x=y;

y=z;

z=(x+y);

//if (n>1){z=(x%100+y%100)%100;}

}

//return z;

return z%m;

}

};

int main(void) {

long long n;

int m;

std::cin >> n >> m;

//std::cout << Fibonacci::get\_remainder(n, m) << std::endl;

std::cout <<array::get\_remainder(n,m) << std::endl;

return 0;

}

Privet

poka