

Problem F. Is This FFT?

Input file: *standard input*
 Output file: *standard output*
 Time limit: 15 seconds
 Memory limit: 998 244 353 bytes

Consider the following method of generating a random tree of size n :

Take all $n(n - 1)/2$ possible edges, and choose a uniformly random permutation on them. Start with an empty graph, iterate over the edges in the chosen order, and if the edge connects two different connected components, add it to the graph. In the end, you will get a tree.

For all n from 2 to given N , calculate the probability that this algorithm generates a bamboo modulo given P . It is guaranteed that P is prime and $P \bmod 2^{16} = 1$.

Recall that bamboo is a tree in which degrees of all vertices are at most 2.

Since we give you the modulus, you might want to use Barrett reduction (https://en.wikipedia.org/wiki/Barrett_reduction).

It allows you to compute $x \bmod P$ several times faster than usual, where P is constant but is not known at compile time. Link to KACTL implementation with additional info:

<https://github.com/kth-competitive-programming/kactl/blob/main/content/various/FastMod.h>.

Code:

```
using u32 = unsigned int;
using u64 = unsigned long long;
using u128 = __uint128_t;
struct Barrett {
    u64 b, m;

    Barrett() : b(), m() {}
    Barrett(u64 _b) : b(_b), m(-1ULL / _b) {}

    u32 reduce(u64 x) {
        u64 q = (u64)((u128(m) * x) >> 64), r = x - q * b;
        return r - b * (r >= b);
    }
} BA;
```

Usage example:

```
u32 mult(u32 x, u32 y) {
    return BA.reduce((u64)x * y);
}

int main() {
    int n, P;
    cin >> n >> P;
    BA = Barrett(P);
    cout << mult(123456, 7890123) << endl;
    return 0;
}
```

Don't forget to construct an instance of Barrett before using it.

We do not guarantee that this problem is solvable without Barrett reduction or other fast multiplication methods.

Input

The only line contains two integers N and P ($2 \leq N \leq 250$, $2 < P < 10^9$). It is guaranteed that P is prime and $P \bmod 2^{16} = 1$.

Output

Print $N - 1$ integers — answers to the problem for all n from 2 to N .

Recall that, if probability can be expressed as an irreducible fraction u/v , and v is coprime with P , the probability modulo P is $(u \cdot v^{-1}) \bmod P$, where v^{-1} is the multiplicative inverse of v modulo P .

Example

| <i>standard input</i> | <i>standard output</i> |
|-----------------------|--|
| 10 998244353 | 1 1 532396989 328786831 443364983 567813846 34567523 466373946 474334062 |