

乘法逆元(exgcd)

```
11 exgcd(11 a,11 b,11 &x,11 &y){  
    if(b==0){x=1,y=0;return a;}  
    11 d=exgcd(b,a%b,y,x);  
    y-=a/b*x;return d;  
}  
11 inv(11 a,11 p){  
    11 x,y;  
    exgcd(a,p,x,y);  
    return x;  
}
```

乘法逆元(费马小定理)(p为质数且a不是p的倍数)

```
11 qpow(11 x,int y,int p){  
    11 ans=1;  
    for(;y;y>>=1){  
        if(y&1) ans=ans*x%p;  
        x=x*x%p;  
    }  
    return ans;  
}  
int inv(int a,int p){return qpow(a,p-2,p);}
```

乘法逆元(递推)

```
#include <bits/stdc++.h>  
using namespace std;  
typedef long long ll;  
const int maxn=3000005;  
int n,p;  
ll inv[maxn];  
int main(){  
    scanf("%d%d",&n,&p);  
    inv[1]=1;  
    for(int i=2;i<=n;i++){  
        inv[i]=(p-inv[p%i]*(p/i)%p);  
    }  
    for(int i=1;i<=n;i++){  
        printf("%lld\n",inv[i]);  
    }  
    return 0;  
}
```

乘法逆元(倒推)

```
#include <bits/stdc++.h>  
using namespace std;  
typedef long long ll;  
const int maxn=305;
```

```
int n,p;
ll inv[maxn],fac[maxn];
ll qpow(ll x,int y,int p){
    ll ans=1;
    for(;y;y>>=1){
        if(y&1) ans=ans*x%p;
        x=x*x%p;
    }
    return ans;
}
int main(){
    scanf("%d%d",&n,&p);
    fac[0]=1;
    for(int i=1;i<=n;i++){
        fac[i]=fac[i-1]*i%p;
    }
    inv[n]=qpow(fac[n],p-2,p);
    for(int i=n-1;i>=0;i--){
        inv[i]=inv[i+1]*(i+1)%p;
    }
    for(int i=1;i<=n;i++){
        printf("%lld\n",fac[i-1]*inv[i]%p);
    }
    return 0;
}
```