Math

// Binary Exp & Mul

int mul(int x, int y)

{

    int ans=0;

    x%=mod; y%=mod;

    while (y)

    {

        if (y&1) ans=(ans+x)%mod;

        x=(2\*x)%mod;

        y>>=1;

    }

    return ans;

}

int pow(int x, int y)

{

    int ans=1;

    x%=mod;

    while (y)

    {

        if (y&1) ans=(ans\*x)%mod;

        x=(x\*x)%mod;

        y>>=1;

    }

    return ans;

}

// Prime Sieve

void init()

{

    for (int i=2; i<=N; i++)

    {

        if (lp[i]==0)

        {

            lp[i]=i;

            p.push\_back(i);

        }

        for (int j=0; i\*p[j]<=N; j++)

        {

            lp[i\*p[j]]=p[j];

            if (p[j]==lp[i]) break;

        }

    }

}