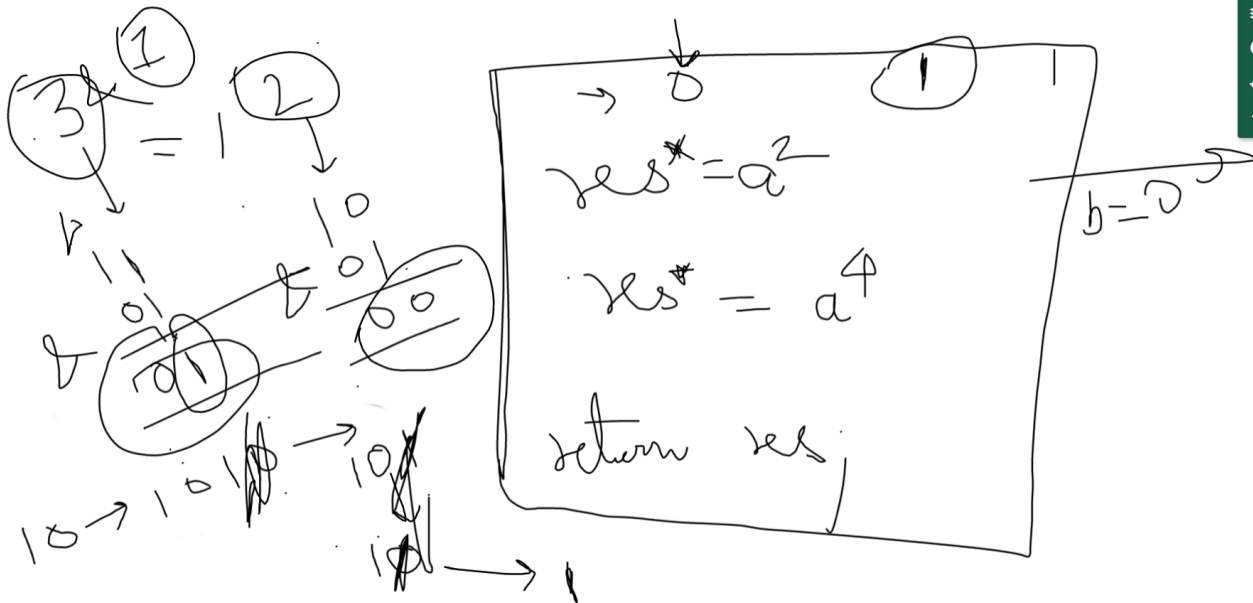


Binary Exponentiation



```

int binpow(int a,int n){

    //a^n%mod

    if(n==0)
        return 1;

    int subans=binpow(a,n/2);

    //n is even
    if((n%2)==0){
        return subans*subans;
    }

    //n is odd
    return subans*subans*a;

}

```

```

int binpow2(int a,int n){

    //a^n
    int res=1;

```

```

while(n>0){

    if(n&1)
        res*=a;//a^8

    a=a*a;//a=a^2

    n>>=1;//(n/=2)

}

return res;

}

```

Modular arithmetic

Read the following two articles.

[Modular arithmetic](#)

[Fermat little theorem](#)

->Modular Binary Exponentiation

```

int modpower(int a, int b, int m)
{
    if(b == 0) return 1;

    a = a % m;

    int temp = power(a, b/2, m);

    if(b%2 !=0){
        return (((temp * temp) % m) * a) % m;
    }
}

```

```
    return (temp * temp) % m;  
}
```

-> $a \% m$ will not have a range from $0 \leq \text{range} \leq m-1$ but from $-m < \text{range} < m$ so be careful in questions where the result of modulo might be negative and don't forget to make it positive by adding an extra m .

Q1) Calculating nCr under modulo p . ([Link to problem](#))

Q2) calculate A^B^C under modulo m where m is a prime number and $0 < A, B, C < 10^9$