```cpp
// 2^4=2*2*2*2
// or,b^p=b*b*b*b.....p time;

// or,making a function name 'power' taking two inputs base and power;

// or,power(b,p)=power(b,p-1)*b; {{follow the 2nd step explaination}}
// or,if p==0 then it is 1 ;



// for making it more faster we have

// consider a example 2^4

// or,(2*2)^2 ;
// or, (b*b)^p/2;
// or, pow(b*b,p/2); {{for recursive calling}} [[Base condition remain same]]
// this will decrease the number of function calls;

// consider example 2^5

// or, 2*(2*2)^2;
// or, b*{(b*b)^(p-1)/2};
// or, b*power(b*b,(p-1)/2);{{for recursive calling}}
// this is for odd power only;

#include<iostream>
using namespace std;
int powerslower(int b,int p){
    if (p==0)
    {
        return 1;
    }
    else
    {
        return powerslower(b,p-1)*b;
    }

}
int powerfaster(int b,int p){
    if(p==0){
        return 1;
    }
    if (p%2==0)
    {
        return powerfaster(b*b,p/2);
    }
    else{
        return b*(powerfaster(b*b,(p-1)/2));
    }

}
int main(){
    cout<<powerslower(2,4)<<endl;
    cout<<powerfaster(2,4)<<endl;

    return 0;
}
```