50. Pow(x, n)

题目描述：https://leetcode.com/problems/powx-n/

> 实现pow(x, n)函数。

解题思路：

二分法
if n % 2 == 0 return pow(x * x, n / 2)
else return x * pow(x * x, n / 2);

代码1：

```
class Solution {
public:
    double caculatePow(double x, int n) {
        if(n == 0)
            return 1;
        if(n % 2 == 0)
            return caculatePow(x*x, n/2);
        return x*caculatePow(x*x, n/2);
    }
    double myPow(double x, int n) {
        if(x == 0)
            return 0;
        if(n == 0)
            return 1;
        if(n < 0) {
            return caculatePow(1/x, -n);
        }
        return caculatePow(x, n);
    }
};
```

代码2：

```
double myPow(double x, int n) {
    double res = 1;
    unsigned long long p;
    if(n == 0) return 1;
    if(n < 0) {
        p = -n;
        x = 1 / x;
    }else {
        p = n;
    }
    while(p) {
        if(p & 1) res *= x;
        x *= x;
        p >>= 1;
    }
    return res;
}
```