Modular Exponentiation :-

372. Super Pow

Medium · Topics · Companies

Your task is to calculate a^b mod 1337 where a is a positive integer and b is an extremely large positive integer given in the form of an array.

Example 1:

Input: a = 2, b = [3]
Output: 8

Example 2:

Input: a = 2, b = [1,0]
Output: 1024

Example 3:

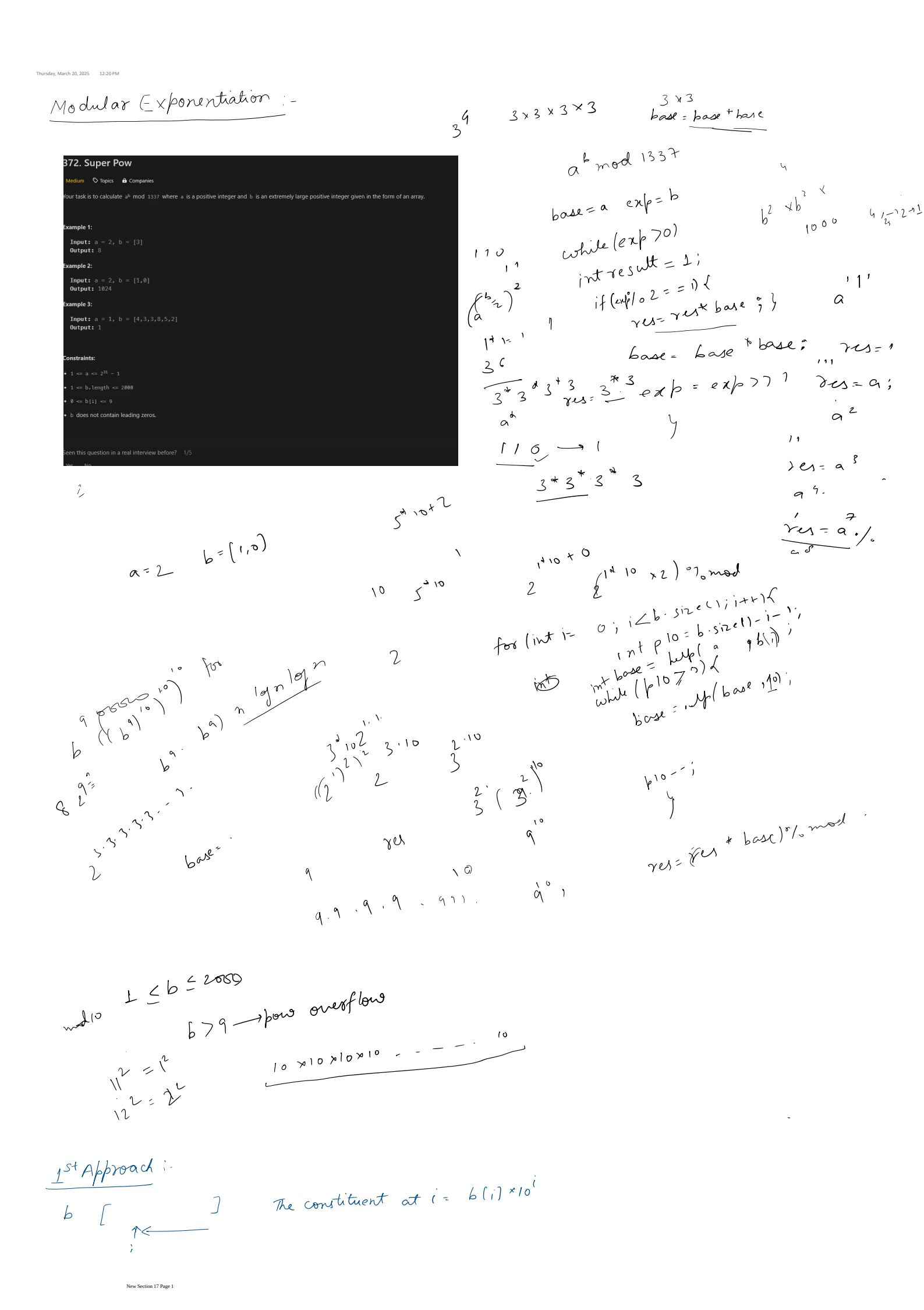
Input: a = 1, b = [4,3,3,8,5,2]
Output: 1

Constraints:

- 1 <= a <= 2^{31} - 1
- 1 <= b.length <= 2000
- 0 <= b[i] <= 9
- b does not contain leading zeros.

Seen this question in a real interview before? 1/5

Yes No



(

```cpp
class Solution {
public:
    int mod=1337;
    int superPow(int a, vector<int>& b) {
        a=a%mod;
        int res=1;
        for(int i=0;i<b.size();i++){
            int j=b.size()-i-1;
            int base=help(a,b[j]);
            res=(res*base)%mod;
            a=help(a, 10);
        }
        return res;
    }
    int help(int base, int pow){
        int res=1;
        while(pow>0){
            if((pow%2)==1){
                res=(res*base)%mod;
            }
            base=(base*base)%mod;
            pow=pow>>1;
        }
        return res;
    }
};
```

Remember:- (Exponentiation by squaring)

```
int pow(int exp, int base){
    base = base % mod;
    int res = 1;
    while (res > 0){
        if( exp&1 == 1){
            res = base x res;
        }
        base = base x base;
        res = res >> 1;
    }
```

Approach 2 :-

```cpp
class Solution {
public:
    int mod=1337;
    int superPow(int a, vector<int>& b) {
        int res=1;
        a=a%mod;
        for(int i=0;i<b.size();i++){
            int p10=b.size()-i-1;
            int base=help(a, b[i]);
            while(p10>0){
                base=help(base, 10);
                p10--;
            }
            res=(res*base)%mod;
        }
        return res;
    }
    int help(int base, int pow){
        int res=1;
        while(pow>0){
            if((pow%2)==1){
                res=(res*base)%mod;
            }
            base=(base*base)%mod;
            pow=pow>>1;
        }
        return res;
    }
};
```