75 Days of Code
Day 56
Problem no : Leetcode 875
Problem Title :Koko Eating Bananas
Problem type : Binary Search

Koko loves to eat bananas. There are n piles of bananas, the ith pile has piles[i] bananas. The guards have gone and will come back in h hours.

Koko can decide her bananas-per-hour eating speed of k. Each hour, she chooses some pile of bananas and eats k bananas from that pile. If the pile has less than k bananas, she eats all of them instead and will not eat any more bananas during this hour.

Koko likes to eat slowly but still wants to finish eating all the bananas before the guards return.

Return the minimum integer k such that she can eat all the bananas within h hours.

Example 1:

Input: piles = [3,6,7,11], h = 8
Output: 4
Example 2:

Input: piles = [30,11,23,4,20], h = 5
Output: 30
Example 3:

Input: piles = [30,11,23,4,20], h = 6

Output: 23

```
function minEatingSpeed(piles: number[], h: number): number {
    let maxBananaToEat = Math.max(...piles);
    let lowestBananaToEat =1;

    while(lowestBananaToEat <maxBananaToEat){
        const  numberOfBanana = Math.floor(lowestBananaToEat + (lowestBananaToEat-lowestBananaToEat)/2);

        const timeToEatBanana = piles.reduce((time ,pile)=>time+ Math.ceil (pile/numberOfBanana) ,0);

        if(timeToEatBanana>h){
            lowestBananaToEat = numberOfBanana+1;

        }else{
            maxBananaToEat = numberOfBanana-1;
        }

    }
    return lowestBananaToEat ;

};
```

Runtime Details

10776 ms

Beats 5.17% of users with TypeScript

Memory

45.49 MB

Beats 91.46% of users with TypeScript