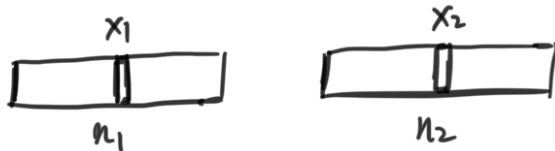


Leetcode Hard #4

计算给定两个数组 $nums1$ 和 $nums2$ 合并后的中位数。

A1: 双指针. $O(n)$

A2: 二分 $O(\log n)$ 相当于求第 k 小数, 因为每次更新后是可更新该需求的。



令 $x_1 = a[k/2]$, $x_2 = b[k/2]$.

若 $x_1 < x_2$, $a[1], a[2], \dots, a[k/2]$ 均不是。

若 $x_1 > x_2$, $b[1], b[2], \dots, b[k/2]$ 均不是。

即 k 以指数速度下降. $O(\log n)$

LeetCode #10

给定字符串 s 和目标正则 p , 正则 p 只包括 '.' 和 '*'。

$$\begin{cases} 0 \leq s.length \leq 20 \\ 0 \leq p.length \leq 30 \end{cases}$$

mississippi

$mis*is*ip*$

$$a^*ap \begin{cases} aap & \checkmark \\ ap & \checkmark \\ p & \times \end{cases}$$

$dp[0][0] \checkmark$

$dp[i][j]$

$\rightarrow dp[i+1][j+2]$
 $(s[i+1] == p[j+1],$
 $p[j+2] == '*')$

$dp[i+1][j+1]$

$(s[i+1] == p[j+1],$
 $p[j+2] == '*')$

$dp[i+1][j]$

$(p[j] == '*'$
 $s[i+1] == p[j-1])$

$dp[i][j]:$

s 串 $0 \sim i-1$ 与
 p 串 $0 \sim j-1$ 匹配。

\leftarrow lens 合适
 \leftarrow lenp 合适

$p[j+1] == p[j+1]$
 $j+2 \geq \text{lenp}$

$dp[i][j+2]$
 $(s[i+1] != p[j+1],$
 $p[j+2] == '*')$

#Exp1:
 SSS. S*

$dp[i][j]$	0	1	2	3
0	✓			
1				
2				
3				

#Exp2:

aab. c*a*b

$dp[i][j]$	0	1	2	3	4	5
0	✓		✓			
1					✓	
2					✓	
3						✓

#Exp3

a. ab*

$dp[i][j]$	0	1	2	3
0	✓			
1		✓		

