

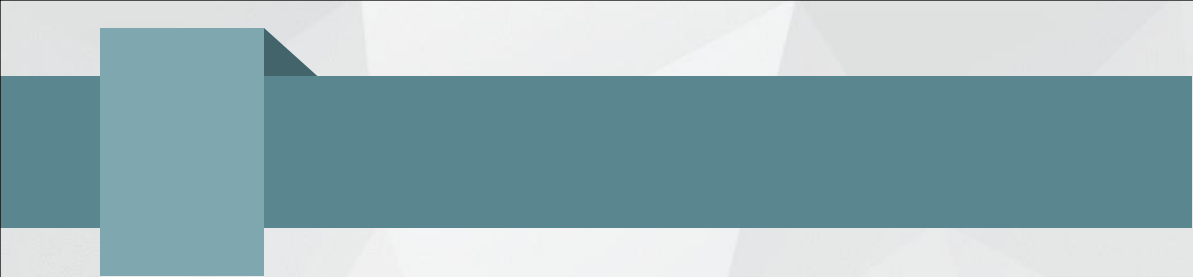
二分

基础

by hjl

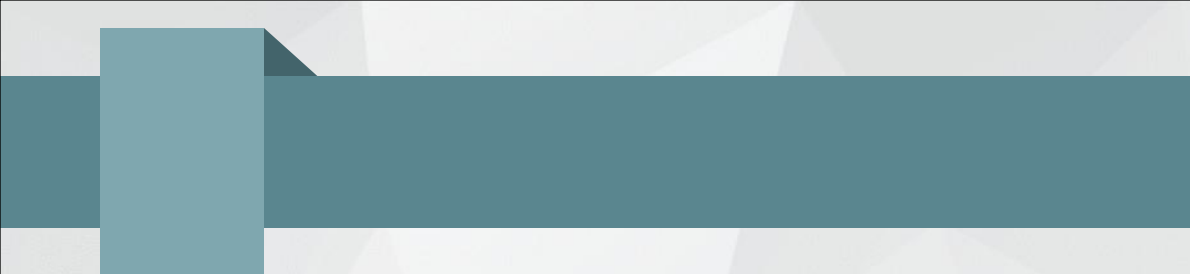
二分的应用

- 猜数字
- 单调函数找零点
- 快速排序
- 找第K大数
- 二分查找
-

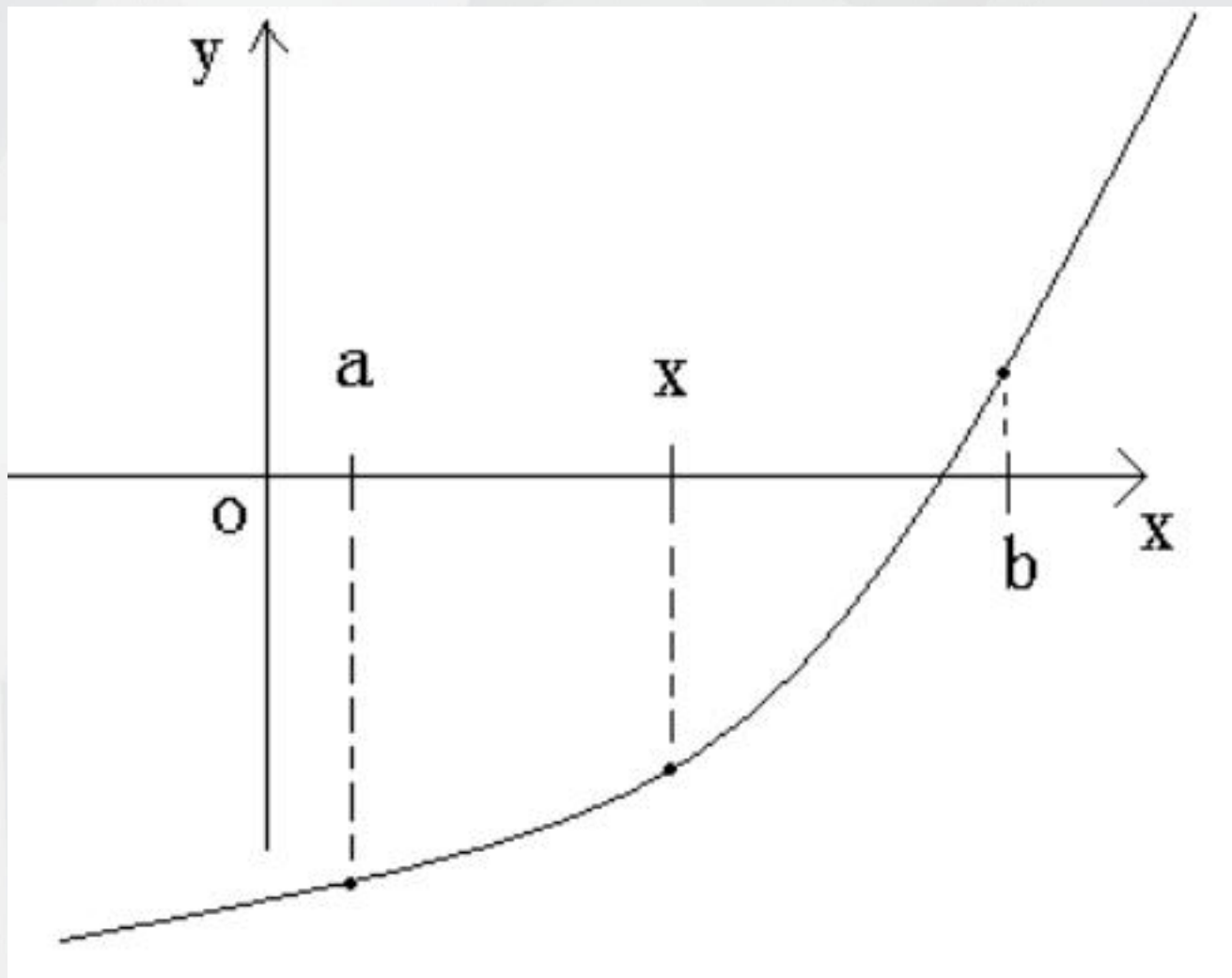
- 
- 有序数列中寻找某一个数
 - 每一次查找的时间复杂度为 $\log N$

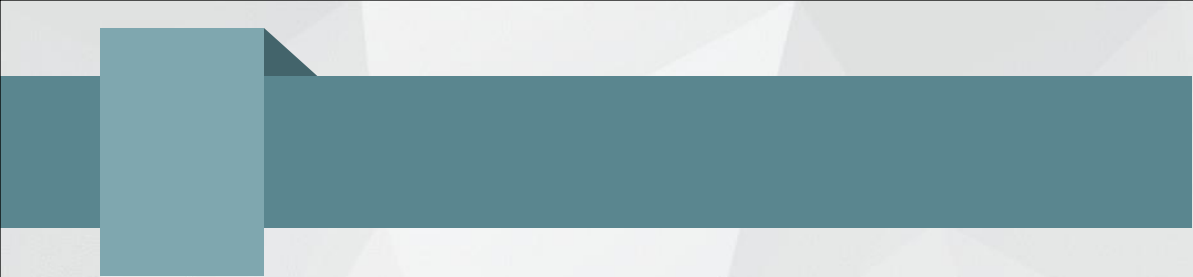
二分答案

- 顾名思义，即利用二分枚举，判断这个值是否满足要求，不断逼近答案，最终得到答案。
- 适合使用二分答案的题目必须满足以下特征：
 - 1、候选答案必须在一个明确的区间内；
 - 2、候选答案必须是离散的（如果答案是连续的，那么在一定的精度范围内可以转化为离散问题）；
 - 3、候选答案在区间内某种属性依次排列，各个类别不能混杂（即满足单调性）。

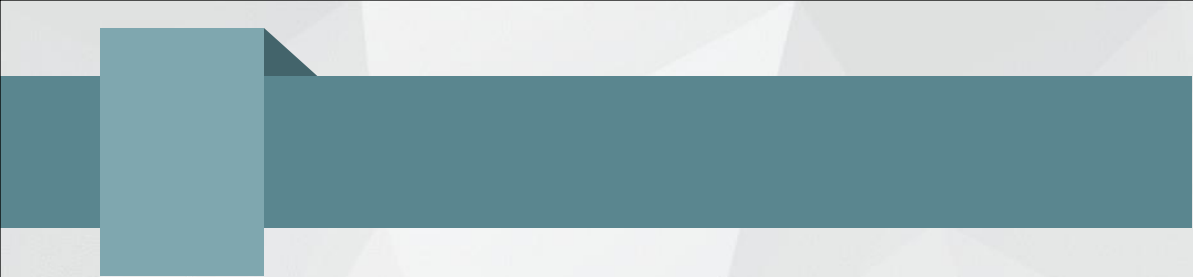
- 
- 在解决二分答案的题目时，通常需要分析出答案的单调性和有界性，并且需要设计出正确高效的验证算法。
 - 由于题目的千变万化，贪心、动态规划、模拟、图论等都有可能成为验证算法。
 - 因此，在近年的各类比赛中，二分答案的题目常常出现。

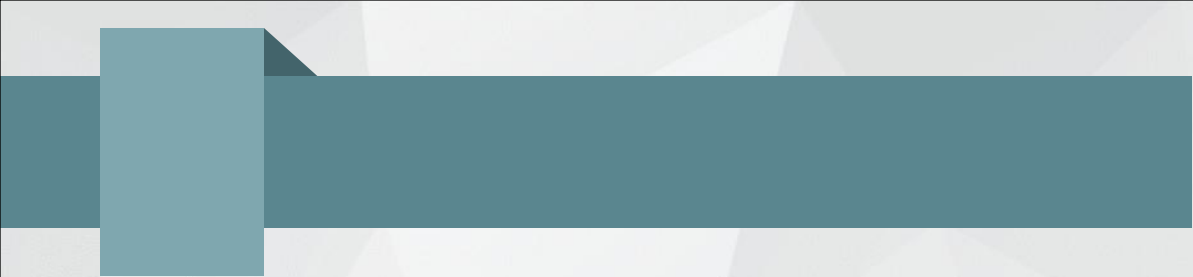
假装你们学了二分法求函数零点

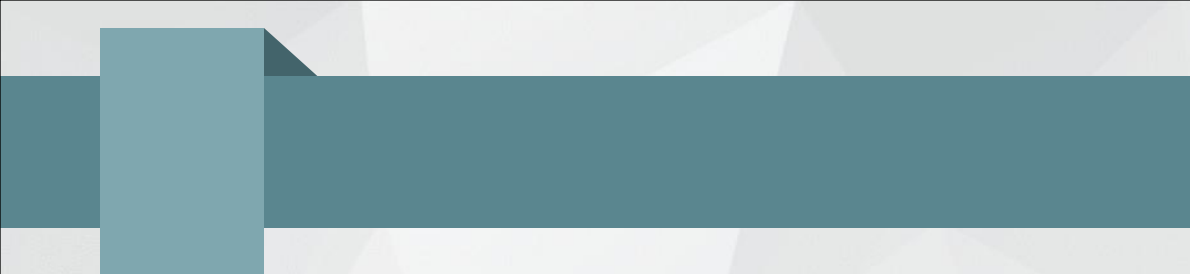


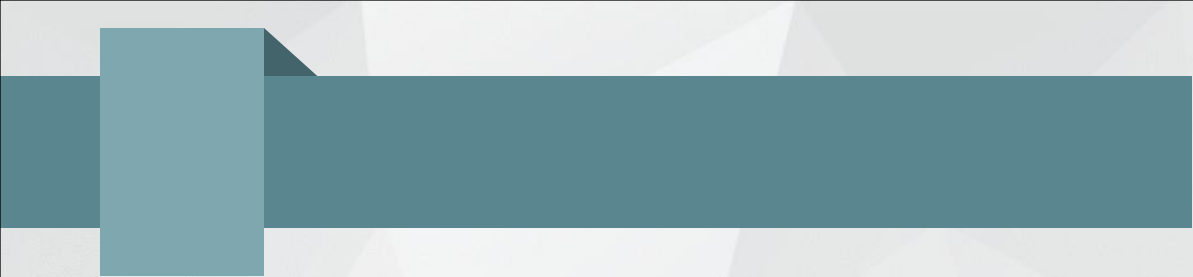


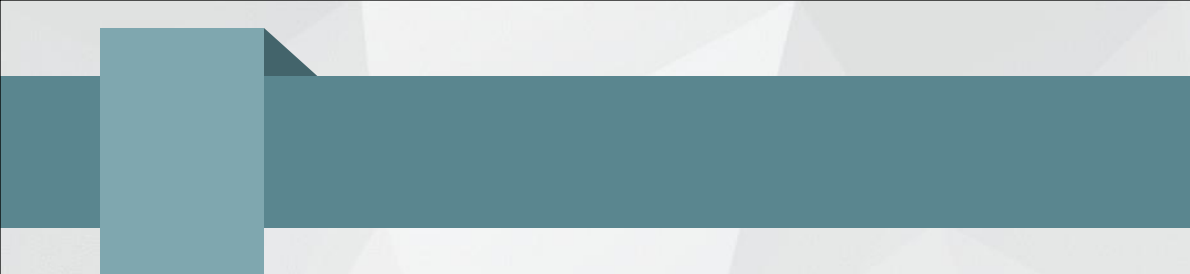
考虑一些具有单调性问题

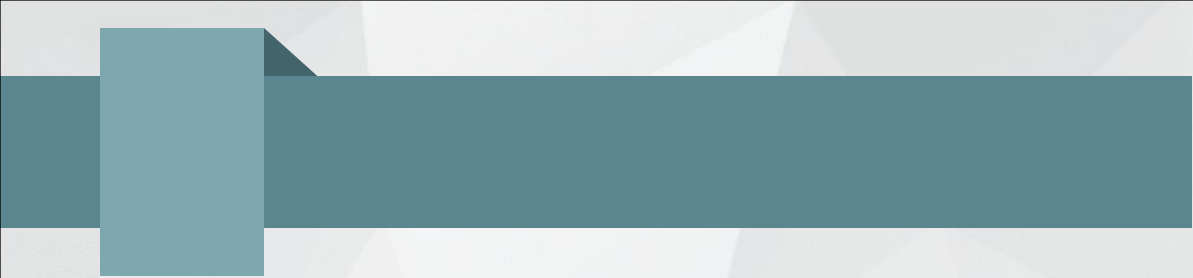
- 
- 题目大意: x 正半轴上 n 个点, 从中选 m 个点使得这些点之间的距离的最小值最大。
 - $n, m \leq 1e5, x_i \leq 1e9$

- 
- 求“最大值最小”,“最小值最大”基本上就可以确定是二分答案了
 - 考虑分析问题答案的单调性

- 
- 题解:假设答案为 X , 可以发现 X 越大能选出来的点就会小于 m ,
 - 同理 X 越小能选出来的点就会越多
 - 满足单调性, 二分答案为 X , 变成可行性问题
 - 如果一个点和上一个被选的点的距离 $\geq X$, 那么这个点就要选
 - 最后判断选了的点是否 $\leq m$

- 
- 题意:给你n个木头, 需要切出m个长度相同的木块, 问木块的最长长度
 - $n \leq 1e5, m \leq 1e8, len \leq 1e9$

- 
- 显然木块长度越长能切出来的数量越少，反之越多
 - 题解：二分答案为 x
 - 一根木头能切出来的木块个数是 len/x (向下取整)
 - 判断总数与 m 的关系就ok了



考虑如何代码实现？

这是答案单调递增的模板

- `int L=(下界),R=(上界),mid;`
- `while(L<R){`
- `mid=(L+R)/2;`
- `if(check(mid)==true)R=mid;`
- `else L=mid+1`
- `}`
- `printf("%d",R);`

单调递减同理