



哈尔滨工业大学

C/C++ JAVA

XPCHF





```
scanf("%d", &n);  
scanf("%s", s);  
scanf("%[0-9]", s);  
scanf("%[^\\r\\n]", s);
```



sort(a, a + n, cmp)
unique(a, a + n)



Why JAVA?

Java在ACM比赛中的应用：高精度，进制转换

时限：2倍 / 3倍



```
import java.io.*;  
import java.util.*;  
public class Main  
{  
    public static void main(String[] args)  
    {  
        Scanner cin = new Scanner (System.in);  
        while (cin.hasNext())  
        {  
            int a = cin.nextInt();  
            System.out.println(a);  
        }  
    }  
}
```



**hasNext()
nextInt()
next()
nextLine()
nextBigInteger()**

**System.out.println()
System.out.printf()**



```
DecimalFormat fd = new DecimalFormat("#.00#");  
DecimalFormat gd = new DecimalFormat("0.000");  
System.out.println("x =" + fd.format(x));  
System.out.println("x =" + gd.format(x));
```



```
String st = "abcdefg";  
System.out.println(st.charAt(0));  
char [] ch;  
ch = st.toCharArray();  
for (i = 0; i < ch.length; i++) ch[i] += 1;  
System.out.println(ch);  
if (st.startsWith("a")) { 20 st = st.substring(1);}
```




```
int n = cin.nextInt();  
int a[] = new int [n];  
for (int i = 0; i < n; i++) a[i] = cin.nextInt();  
Arrays.sort(a);
```



```
String st = Integer.toString(num, base);
```

```
int num = Integer.parseInt(st, base);
```

```
BigInteger m = new BigInteger(st, base);
```



哈尔滨工业大学

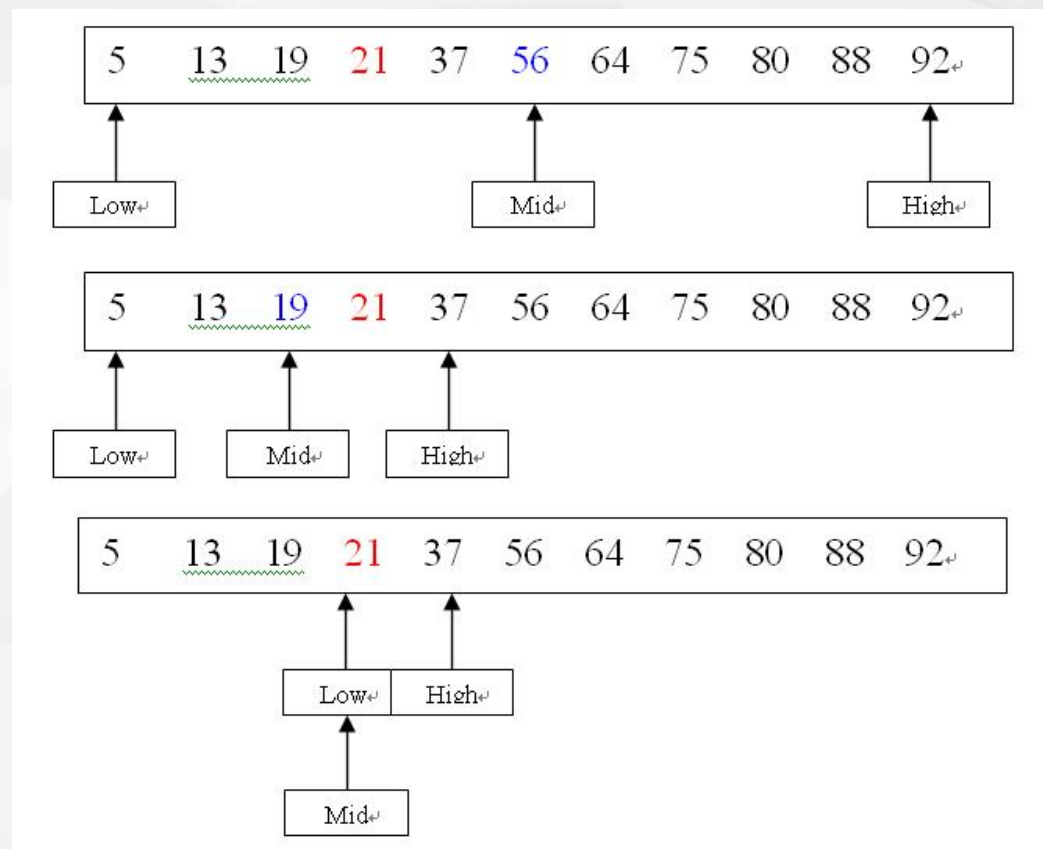
二分

XPCHF





在一个单调有序的集合中查找元素，每次将集合分为左右两部分，判断解在哪个部分中并调整集合上下界，重复直到找到目标元素。





```
int binary_search (int l, int r)
{
    while (r - l > 1)
    {
        int mid = (l + r) / 2;
        if (check(mid)) l = mid;
        else r = mid;
    }
    return l;
}
```



给定浮点数 x ，求出 $\text{sqrt}(x)$ 。

$$x \leq 10^9$$



给定一个有序数组**A**，查找**x**在**A**中第一次出现的下标和最后一次出现的下标？



lower_bound(a, a + n, x)
upper_bound(a, a + n, x)



给定四个长度为 n 的数组，从每个数组中选取一个值。问有多少种选择方案使四个值的和为0。 $n < 4000$ 。



给定一个二维数组**A**，相邻的两个元素保证右侧的大于左侧的，下侧的大于上侧的。查找元素**x**是否出现在此数组中。



给定一个数组 $A[1:n]$ ，找出一个位置使得与它相邻两个位置的值都比当前位置的值小。假定 $A[0] = A[n + 1] = -\infty$ 。



设将一个数组整体右移，后侧数据移到前侧为数组的旋转操作。比如[1,2,3,4,5] -> [4,5,1,2,3]。

给定一个可能被旋转过的单调递增数组，求出该数组的最小值。



对于很多问题，答案是满足单调性的，这时我们可以二分答案，每次对二分出的答案进行判断，从而不断逼近我们需要的正确答案。



给了 n 个石头，给了它们到起点的距离。问要去掉这 n 个石头中的 m 个，使得其间距的最小值最大。



有一些衣服，每件衣服有一定水量，有一个烘干机，每次可以烘一件衣服，每分钟可以烘掉 k 件衣服。每件衣服每分钟可以自动蒸发掉一滴水，用烘干机烘衣服时不蒸发。问最少需要多少时间能烘干所有的衣服。



有 n 个物品，每个物品有两个属性 **A** 与 **B**。请你选取 k 个物品，使得这些物品的 **A** 值之和除以这些物品的 **B** 值之和最大。 $n < 10^5$ 。



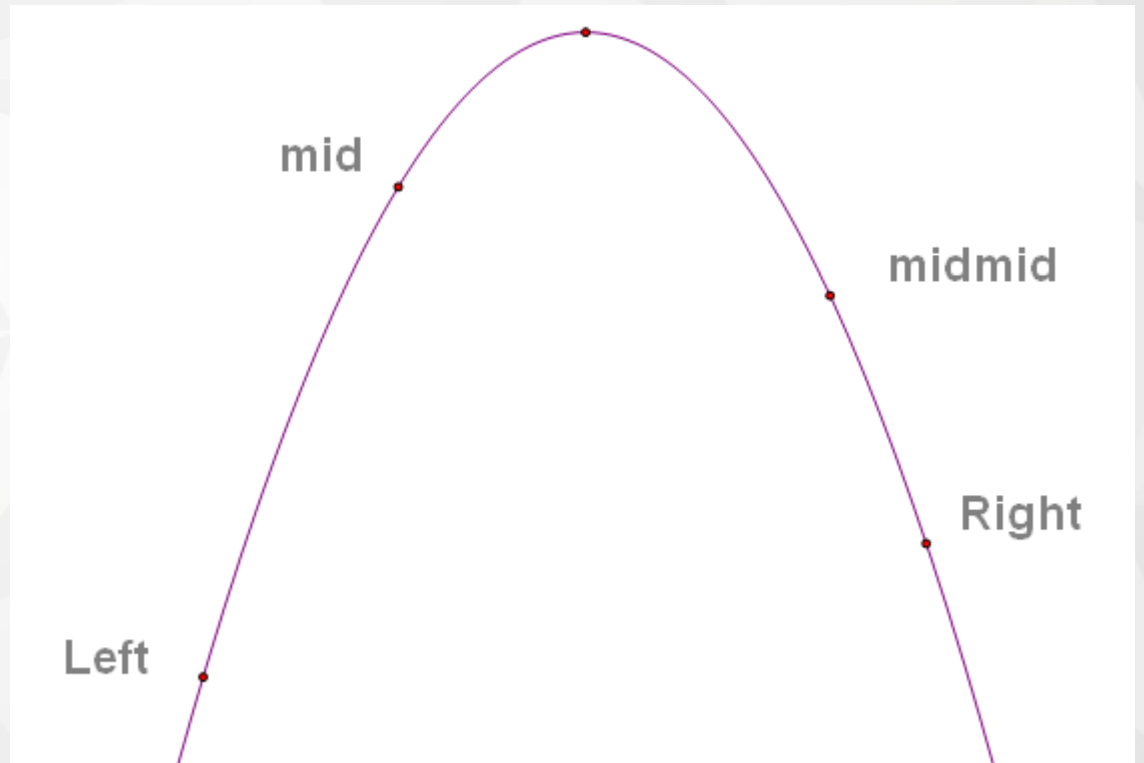
有 n 个物品，每个物品有两个属性 **A** 与 **B**。请你选取 k 个物品，使得这些物品的 **A** 值之和除以这些物品的 **B** 值之和最大。 $n < 10^6$ 。



`nth_element(a, a + n)`



当需要求某凸性或凹形函数的极值，通过函数本身表达式并不容易求解时，就可以用三分法不断逼近求解。





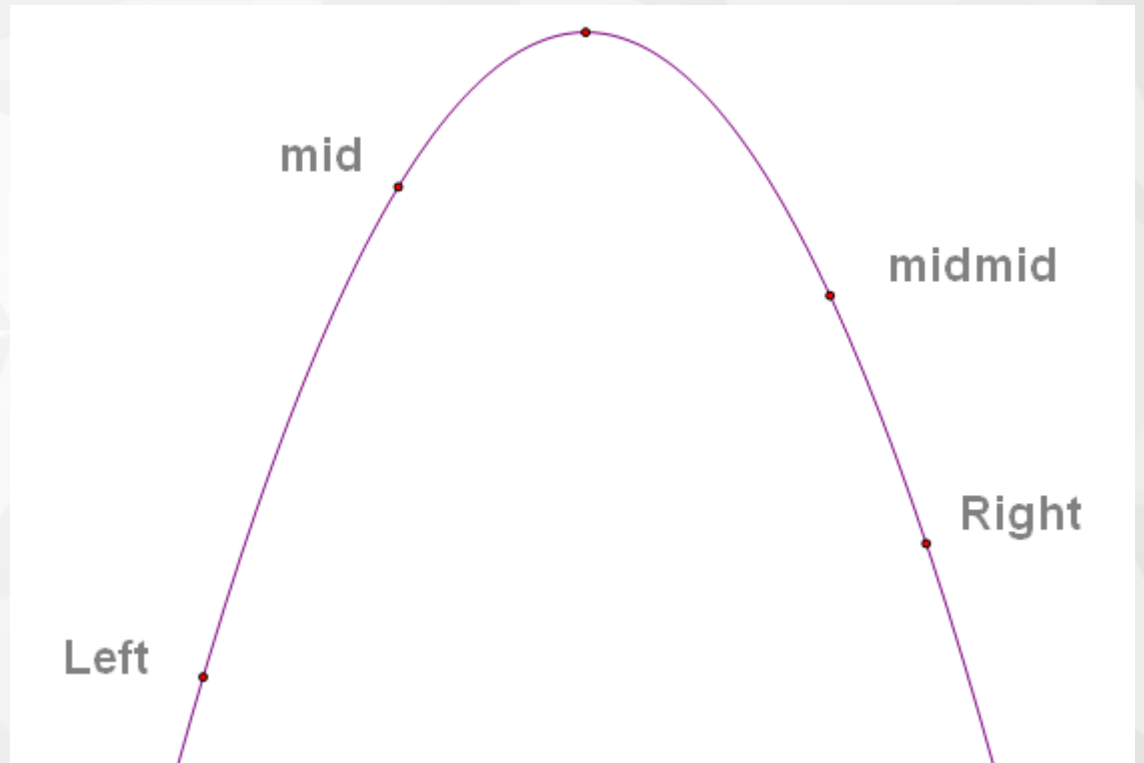
类似二分的定义Left和Right

$\text{mid} = (\text{Left} + \text{Right}) / 2$

$\text{midmid} = (\text{mid} + \text{Right}) / 2$;

如果mid靠近极值点，则Right = midmid;

否则(即midmid靠近极值点)，则Left = mid;





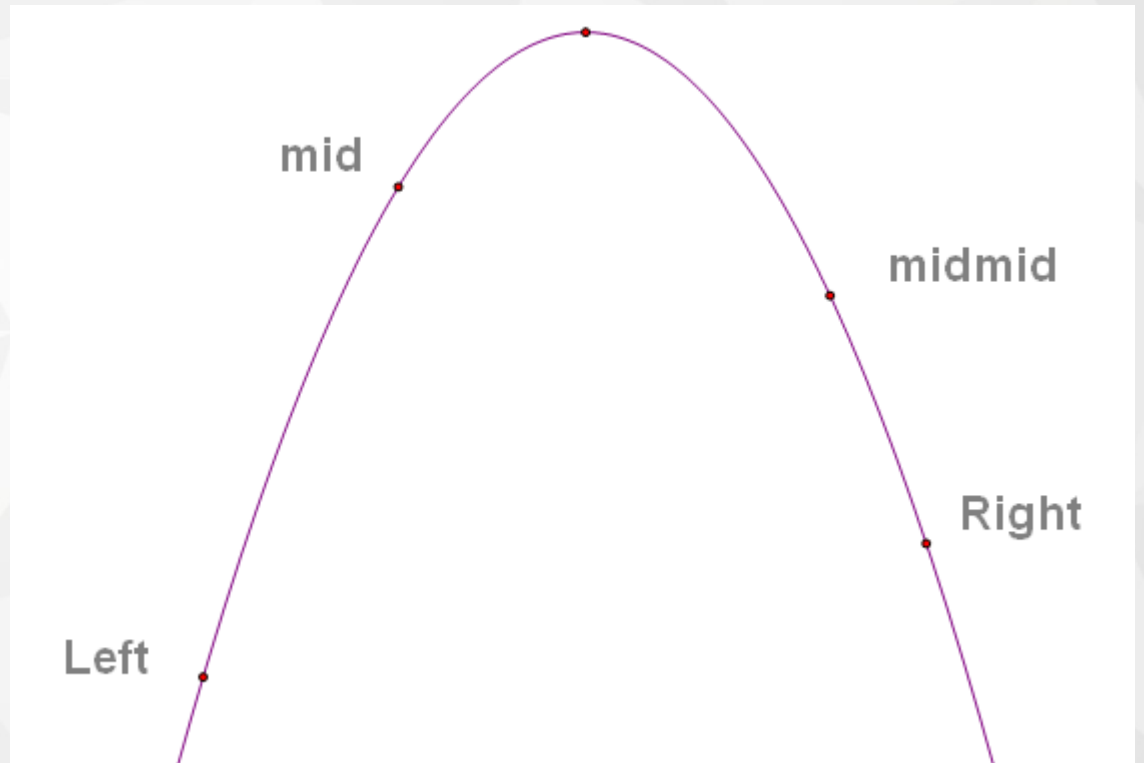
分界点的选取:

$$(L + L + R) / 3$$

$$(L + R + R) / 3$$

$$L + (R - L) * 0.382$$

$$L + (R - L) * 0.618$$





给定 n 个开口向上的二次函数，设 $f(x)$ 为这 n 个二次函数的最大值，求 $f(x)$ 的极小值点。



给定一个序列**A**

一个区间的**poorness**定义为这个区间内和的绝对值

weakness等于所有区间最大的**poorness**

求一个**x**使得，序列**A**全部减**x**后**weakness**最小

$1 \leq n \leq 2 * 1e5$



给定整数 x ，求 $(x + x^2 + \dots + x^y) \% p$ 。
 $x, y < 10^9$ 。



给定两个 n 位大整数，求他们的乘积，要求复杂度小于 $O(n^2)$ 。



给定两个 n 位大整数，求他们的乘积，要求复杂度小于 $O(n^2)$ 。

$$X = \overset{n/2\text{位}}{A} \overset{n/2\text{位}}{B} \quad Y = \overset{n/2\text{位}}{C} \overset{n/2\text{位}}{D}$$

$$\begin{aligned} XY &= (A2^{n/2} + B)(C2^{n/2} + D) \\ &= AC2^n + (AD+BC)2^{n/2} + BD \end{aligned}$$



$$X = \overset{n/2\text{位}}{A} \overset{n/2\text{位}}{B} \quad Y = \overset{n/2\text{位}}{C} \overset{n/2\text{位}}{D}$$

$$\begin{aligned} XY &= (A2^{n/2} + B)(C2^{n/2} + D) \\ &= AC2^n + (AD+BC)2^{n/2} + BD \end{aligned}$$

$$T(n) = 4T(n/2) + O(n)$$

$$AD + BC = (A-B)(D-C) + AC + BD$$

$$T(n) = 3T(n/2) + O(n)$$

$$T(n) = O(n^{\log_2 3})$$



给定 n 个物品，每个物品具有体积与价值。给定背包容量 V ，求在背包容量允许的情况下，选取不超过 k 个物品，可以收获的最大价值。

$n < 500$, $V < 10000$ 。



一条直线上有 N 个村庄，给出每个村庄的坐标（整数）。要在这条直线上选 K 个地方建雕像，使得每个村庄到离其最近的雕像的距离的和最小。输出最小的和。 $N < 100000$ 。