Contest 4 题解

2023.3.29

钱霂泽

A 袋中小球

题目描述

Mas 有 N 个袋子,其中每个袋子内有 T_i 个相同的小球

定义一次操作规则如下:

1. 每次操作可任意选择一个不为空的袋子 x
2. 将其中的 T_x 小球任意选取出不超过 T_x 的 P 个放入新的袋子中
3. 再将剩余的 $T_x - P$ 个小球放入另一个新的袋子中

至多允许 Mas 进行 Q 次操作

在操作结束后袋子内最多的小球最少有多少个?

输入格式

第一行输入两个整数 N, Q

第二行输入 N 个整数 T_i

输出格式

输出一个整数表示答案

输入样例1

```
3 1
1 9 123
```

输出样例1

```
62
```

样例解释1

第 1 次操作将 123 拆分成 $62, 61$
盒子内小球数量为 $1, 9, 61, 62$

输入样例2

```
1 2
10
```

输出样例2

```
4
```

样例解释1

第 1 次操作将 10 拆分成 $4, 6$
第 2 次操作将 6 拆分成 $3, 3$
袋子内小球数量为 $3, 3, 4$

数据规模

对于 5% 的数据, $1 \le N \le 10, Q = 1$

对于另外 10% 的数据, $1 \le N \le 10, 1 \le Q \le 2$

对于前 50% 的数据, $1 \le N, T_i, Q \le 1000$

对于 100% 的数据, $1 \le N \le 2 \times 10^5, 1 \le Q, T_i \le 10^9$

A 袋中小球

本题主要考察二分答案中的“最大值最小”问题。

根据题意，操作次数不允许超过Q次，可以枚举每个袋子中小球数的最大值X，而一个袋子需要进行操作的次数由X决定（分成若干个袋中球数不超过X的袋子）。袋中球数与操作次数的关系如下表所示：

袋中球数	操作次数
$0 < T_i \leq X$	0
$X < T_i \leq 2X$	1
$2X < T_i \leq 3X$	2
······	

由此可见，对于一个袋子，需要进行的操作次数为$\lfloor (T_i-1)/x \rfloor$，则所有袋子需要操作的次数为$\sum_{i=1}^{n} \lfloor (T_i-1)/x \rfloor$，将其计算出结果并检验是否小于等于Q即可。

A 袋中小球

核心代码:

```
bool check(int n,int q,int x) {
    int i, cnt = 0;
    for (i=1;i<=n;i++) {
        cnt = cnt+(a[i]-1)/x;
    }
    return cnt<=q;
}
```

B 泳池

题目描述

Mas 要在 $n \times n$ 的方格区域修建泳池,方格区域的第 i 第 j 列的高度为 h_{ij} .

这个泳池的大小为 $k \times k$

修建泳池的费用由选定的区域内第 $(\lfloor \frac{K^2}{2} \rfloor + 1)$ 高的的区域决定

现在 Mas 想要知道修建泳池的最小花费,请你帮帮他

输入格式

第一行输入两个整数 n, k

接下输出 n 行 n 列,其中第 i 第 j 列的整数表示 h_{ij} .

输出格式

输出一个整数,表示最小花费

输入样例1

```
3 3
1 2 3
4 5 6
7 8 9
```

输出样例1

```
5
```

输入样例2

```
3 2
1 7 0
5 8 11
10 4 2
```

输出样例2

```
4
```

样例解释2

一共存在 4 个 2×2 的区域
其中 $(\lfloor \frac{K^2}{2} \rfloor + 1) = 3$
对于左上角为 $(1, 1)$ 右下角为 $(2, 2)$ 第 3 大的高度为 5
对于左上角为 $(1, 2)$ 右下角为 $(2, 3)$ 第 3 大的高度为 7
对于左上角为 $(2, 1)$ 右下角为 $(3, 2)$ 第 3 大的高度为 5
对于左上角为 $(2, 2)$ 右下角为 $(3, 3)$ 第 3 大的高度为 4

数据规模

对于 20% 的数据 $1 \le n \le 10$
对于 100% 的数据 $1 \le n \le 800, 1 \le k \le n, 0 \le h_{ij} \le 10^9$

B 泳池

这道题的关键点在于二维前缀和+二分答案，这个思路不易被想到，但比较易懂。

首先，二分一个高度X，将X在某一K*K区域内的高度排名小于等于小于($\lfloor K^2/2 \rfloor$+1)作为条件，再将X和原数组中每一个高度H_{ij}进行比较，如果X大于H_{ij}，那么记为1，否则记为0。对这个01数组求出前缀和数组sum[]，再遍历求出sum[]中每一个K*K区域的和，如果在遍历过程中发现区域的和小于给出的($\lfloor K^2/2 \rfloor$+1)，说明X在该区域中的高度排名小于等于($\lfloor K^2/2 \rfloor$+1)，返回true；如果在遍历结束后仍没有发现满足条件的区域，返回false。

B 泳池

核心代码:

```
bool check(long long n,long long k,long long x) {
    long long i, j, num1, num2;
    for (i=1;i<=n;i++) {
        for (j=1;j<=n;j++) {
            sum[i][j] = sum[i-1][j]+sum[i][j-1]-sum[i-1][j-1];
            if (a[i][j]>x) sum[i][j]++;
        }
    }
    for (i=1;i+k-1<=n;i++) {
        for (j=1;j+k-1<=n;j++) {
            num1 = i+k-1;
            num2 = j+k-1;
            if (sum[num1][num2]-sum[num1][j-1]-sum[i-1][num2]+sum[i-1][j-1]<k*k/2+1) return true;
        }
    }
    return false;
}
```

C 绩点计算

题目描述

Mas 选择了大学的 n 门课程

第 i 门课程的学分是 s_i ,第 i 门课程的分数是 c_i

他的最终成绩为

$$\frac{\sum(s_i \times c_i)}{\sum s_i}$$

现在 Mas 最多可以删除 k 个课程,他想知道可以得到的最高分是多少

输入格式

第一行输入两个正整数 n, k

第二行有 n 个正整数 s_i

第三行有 n 个正整数 c_i

输出格式

输出最高的最终得分,保留 5 位小数

输入样例1

```
3 1
1 2 3
3 2 1
```

输出样例1

```
2.33333
```

样例解释1

删除第三门课程,最终得分为 $\frac{2 \times 2 + 3 \times 1}{2+1} = \frac{7}{3}$

输入样例2

```
3 1
1 3 6
1 3 5
```

输出样例2

```
4.42857
```

样例解释2

删除第二门课程,最终得分为 $\frac{1 \times 1 + 5 \times 6}{1+6} = \frac{31}{7}$

数据规模

对于全部的数据 $1 \le n \le 10^5, 0 \le k \le n, 1 \le s_i, c_i \le 10^3$

C 绩点计算

这道题在简化了以后难度和A+B Problem差不多，思路仍然是二分答案。

题目中要求计算在删除了K门课程后$\sum_{i=1}^{n-k} S_iC_i / \sum_{i=1}^{n-k} S_i$的最大值，首先二分答案X，将该式子的结果大于等于X作为条件，则:

$$\sum_{i=1}^{n-k} S_iC_i / \sum_{i=1}^{n-k} S_i \geq X$$
$$\sum_{i=1}^{n-k} S_iC_i \geq \sum_{i=1}^{n-k} S_iX$$
$$S_i\sum_{i=1}^{n-k} C_i \geq S_i\sum_{i=1}^{n-k} X$$
$$S_i\sum_{i=1}^{n-k} (C_i-X) \geq 0$$

条件变为了$S_i\sum_{i=1}^{n-k} (C_i-X) \geq 0$。
需要先将每一个式子的结果都算出来(1~n中每一个$(C_i-X)S_i$)，然后排序找出最大的(n-k)个数并累加，与0作比较即可。

C 绩点计算

核心代码：

```
bool check(int n,int k,double x) {
    int i;
    double cnt = 0;
    for (i=1;i<=n;i++) {
        sum[i] = s[i]*(c[i]-x);
    }
    sort(sum+1,sum+n+1);
    for (i=k+1;i<=n;i++) {
        cnt = cnt+sum[i];
    }
    return cnt>=0;
}
```

D 未出现的数

题目描述

给出一个长度为 N 的单调递增序列 A

保证 $A_1 < A_2 < \cdots < A_{n-1} < A_n$

同时给出 Q 个询问

每组询问给出一个整数 K ,请你找出不在 A 中的第 K 小的整数值

输入格式

第一行输入两个整数 N, Q

第二行输入 N 个整数 A_i

接下来 Q 行,每行一个整数 K

输出格式

每组询问输出一行,输出第 K 小不在 A 中的整数值

输入样例

```
4 3
3 6 8 100
2
5
3
```

输出样例

```
2
7
4
```

数据规模

对于 10% 的数据 $1 \leq N, Q, A_i \leq 50$

对于 30% 的数据 $1 \leq N \times Q \leq 10^8$

对于全部的数据 $1 \leq N, Q \leq 10^5, 1 \leq A_i, K \leq 10^{18}$

D　未出现的数

这道题又双叒叕是一道二分答案题，也是整个Contest 4最水的题。

很明显，想要判断一个数是在正整数范围内中除给定数组元素外第几个数，只需将自己的值减去在数组中比自己小的元素的个数就行。而实现方法就是二分一个X，在check()函数中利用upper_bound()函数计算一下，并检查X在排除比自己小的数后是否大于等于给定的K即可。

D 未出现的数

这道题不会给出任何代码。

THANK'S