

信奥测试

(考试时长: 3 小时)

题目名称	捡球	俄罗斯方块	色香味	计数
目录	ranwen	tetris	dish	count
可执行文件名	ranwen	tetris	dish	count
输入文件名	ranwen.in	tetris.in	dish.in	count.in
输出文件名	ranwen.out	tetris.out	dish.out	count.out
每个测试点时限	1S	1S	1S	2S
空间限制	512MB	512MB	512MB	512MB
题目类型	传统型	传统型	传统型	传统型

注意: 最终测试时, 开启 O2 优化, 评测结果以 Noi Linux 为准。

比赛文件夹结构要求:

选手比赛时, 需在 D 盘根目录下建立一个文件夹, 该文件夹以本人编号命名。如: 选手编号为 “JH-001”, 则该文件夹命名为 “JH-001”。在此文件夹里, 为每题建立对应的题目文件夹, 文件夹名称与题目英文名称相同, 严格区分大小写。选手根据题目要求, 将自己编写的源程序, 保存在该题的文件夹里。

比赛结束前, 选手应检查自己的源程序和文件夹是否已按要求保存在 D 盘中或者主文件夹下供提交。未按规定建立文件夹导致提交失败的, 将无法进行测试。

例如: 假设题目有 cashier、dune、manhattan 三题。

选手 JH-001 分别使用 C++ 答题, 其最终提交的文件为 cashier.cpp、dune.cpp、manhattan.cpp, 则该选手提交的目录结构如下所示:

```
D:\
|--- JH-001
|   |---cashier\
|   |   |---cashier.cpp
|   |--- dune\
|   |   |--- dune.cpp
|   |---manhattan\
|       |---manhattan.cpp
```

1. 捡球

【问题描述】

在一条直线上, 有 n 个位置 (坐标为 $1, 2, \dots, n$), 每个位置上有 a_i 个球。小 A 派了 m 个学生, 去捡这些球。所有学生初始位置在 0, 每秒钟, 他们有两种选择:

- ① 向右走一步。
- ② 捡起一个位置上的一个球。

要捡起所有球，最少需要多少秒。

【输入格式】

第一行输入两个正整数 n, m ，如题目所述。

第二行输入 n 个非负整数 a_i ，表示每个位置上球的个数。

【输出格式】

一个数字，表示最少时间。

【输入样例】

```
3 2
1 0 2
```

【输出样例】

```
5
```

【样例解释】

第一个学生走到位置 3，把这里的两个球都捡起来，一共花费 5 秒。第二个学生走到 1，把这里的球捡起来，花费 2 秒。两个学生可以同时操作，所以最终时间是 5 秒。

【数据范围】

对于 30% 的数据， $m=1$ 。

对于另 30% 的数据， $m=2$ ，球的总个数 ≤ 21 。

对于 100% 的数据， $1 \leq n, m \leq 100,000$ ， $0 \leq a_i \leq 10^9$ ，保证至少一共有 1 个球。

2. 俄罗斯方块

【问题描述】

俄罗斯方块是一款非常经典的游戏。游戏中方块的大小是 1×1 。游戏的主页面是一个底的宽度为 n 、高无限大的矩阵。在游戏过程中，一个个骨牌会从上面降落下来。骨牌是由若干方块组成的。小 A 玩了一阵后，主界面上从左到右的高度分别是 $h_1, h_2, \dots, h_n$ 。接下来假设主界面上降下来的骨牌只会是长方形，而且在降落过程中无法旋转和移动。接下来会降下 m 个骨牌，第 i 个骨牌的大小和位置可以用 l_i, r_i 和 d_i 三个参数表示。表示这个骨牌降落下来后，它的底位于 $[l_i, r_i]$ 这个区间，高是 d_i 。对于这个 m 个骨牌，求每个骨牌降落后，它的底部所处的高度。

【输入格式】

第一行一个整数 n ，表示游戏主界面的底的宽度。

第二行 n 个正整数 $h_1, h_2, \dots, h_n$ 。

第三行一个整数 m ，表示有 m 个骨牌要降落。

接下来 m 行，每行三个整数 l_i, r_i 和 d_i 。

【输出格式】

输出共 m 行，每行一个整数。第 i 行的整数表示第 i 个骨牌降落后底部所处的高度。

【输入样例 1】

```
5
1 2 3 6 6
4
1 1 1
1 3 1
```

1 1 1

1 4 3

【输出样例 1】

1

3

4

6

【输入样例 2】

7

1 3 1 4 5 1 5

5

3 5 4

2 5 4

4 7 1

4 7 4

1 3 2

【输出样例 2】

5

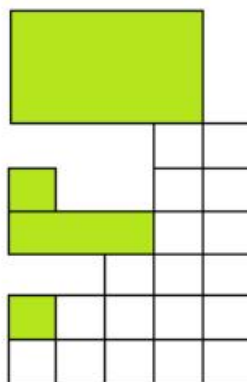
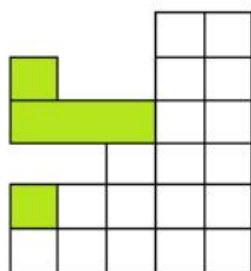
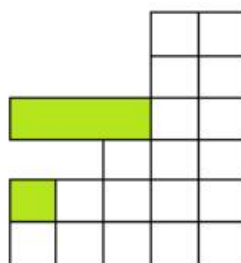
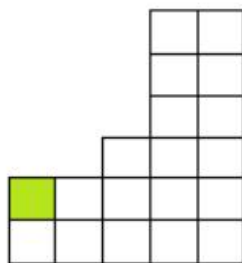
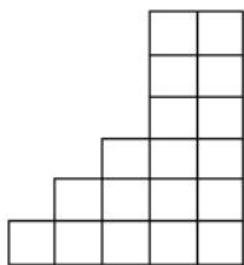
9

13

14

13

【样例 1 解释】



【数据范围】

测试点编号	总分值	n	m	特殊性质
1 ~ 2	10	$n = 1$	$1 \leq m \leq 10^3$	有
3 ~ 6	20	$1 \leq n \leq 10^3$	$1 \leq m \leq 10^3$	有
7 ~ 14	40	$1 \leq n \leq 2 \times 10^5$	$1 \leq m \leq 2 \times 10^5$	有
15 ~ 20	30	$1 \leq n \leq 2 \times 10^5$	$1 \leq m \leq 2 \times 10^5$	无

特殊性质: $h_1 \leq h_2 \leq \dots \leq h_n$; 且对于 $1 \leq i \leq m$, 都有 $l_i = 1$ 。

对于所有数据, 都有:

对于 $1 \leq i \leq n$, 都有 $1 \leq h_i \leq 10^3$;

对于 $1 \leq i \leq m$, 都有 $1 \leq l_i \leq r_i \leq n$ 且 $1 \leq d_i \leq 10^3$ 。

3. 色香味

【问题描述】

暑假期间, 你来到了 A 市最受欢迎的菜馆。这家菜馆一共有 n 道菜肴, 从 1 到 n 编号。为了方便顾客点菜, 这家菜馆在菜单上给出了每道菜肴的三个属性: 色泽、香气和味道。这三个属性的值都是正整数: 第 i ($1 \leq i \leq n$) 道菜的三个属性分别用 r_i, x_i 和 d_i 表示。

你是一个特别会点菜的人。假设你点了 k ($k \geq 1$) 道菜, 它们的编号分别是 $t_1, t_2, \dots, t_k$ 。你点菜遵从以下规则:

- 你所点的每道菜的色泽 r_i 互不相同, 色泽不同的菜肴凑在一起会让你心旷神怡。
- 你想尝试菜单上所有的香气。也就是说, 对于任意一个 x_i ($1 \leq i \leq n$), 都存在至少一个 j ($j \leq k$), 满足 $x_{t_j} = x_i$ 。
- 你所点的菜肴的味道满足 $d_{t_1} \oplus d_{t_2} \oplus \dots \oplus d_{t_k} = 0$, 这会充分打开你的味蕾。 $\oplus$ 指的是异或操作。

现在你想知道, 你有多少种不同的点菜方案, 满足上述三个条件。

【输入格式】

第一行有一个整数 n , 如题意。

第二行到第 $n+1$ 行, 每行都有三个整数, 表示 r_i, x_i 和 d_i 。

【输出格式】

一行一个整数, 表示答案。保证答案在 c++ 的 int 范围内。

【输入样例】

```
5
1 2 1
1 1 2
1 2 2
2 1 1
3 2 2
```

【输出样例】

```
2
```

【样例解释】

方案 1: 点菜肴 1 和菜肴 4 ;

方案 2: 点菜肴 2 和菜肴 5。

【数据范围】

测试点编号	总分值	n	r_i	x_i	d_i
1	4	$n = 2$	$1 \leq r_i \leq 2$	$1 \leq x_i \leq 2$	$d_i < 2^6$
2 ~ 4	12	$1 \leq n \leq 16$	$1 \leq r_i \leq 16$	$1 \leq x_i \leq 5$	$d_i < 2^6$
5 ~ 7	12	$1 \leq n \leq 32$	$1 \leq r_i \leq 5$	$1 \leq x_i \leq 5$	$d_i < 2^7$
8 ~ 11	16	$1 \leq n \leq 32$	$1 \leq r_i \leq 16$	$1 \leq x_i \leq 6$	$d_i < 2^7$
12 ~ 15	16	$1 \leq n \leq 32$	$1 \leq r_i \leq 12$	$1 \leq x_i \leq n$	$d_i < 2^{17}$, 所有 d_i 都相等
16 ~ 20	20	$1 \leq n \leq 32$	$1 \leq r_i \leq 16$	$1 \leq x_i \leq n$	$d_i < 2^{17}$
21 ~ 25	20	$1 \leq n \leq 32$	$1 \leq r_i \leq n$	$1 \leq x_i \leq n$	$d_i < 2^{17}$

4.计数

【问题描述】

给你一个长度为 n 的序列 a ，求出所有满足区间最小值与最大值的二进制数中 1 的数量相同的区间数量。

【输入格式】

一行一个整数 n 代表序列的长度。

接下来一行 n 个整数表示序列 a 。

【输出格式】

一行一个数表示满足条件的区间数量。

【输入样例】

5
1 2 3 4 5

【输出样例】

9

【样例解释】

1
2
3
4
5
1-2
1-4
2-4
3-5

【数据范围】

对于 30% 的数据， $n \leq 1000$

另有 20% 的数据，序列 a 单调递增。

对于 100% 的数据， $1 \leq n \leq 10^6, 0 \leq a_i \leq 10^{18}$ 。