

三校联考测试 2

比赛时间：2019 年 8 月 31 日 08:00–11:30

中文题目名称	精灵球	球星卡	整除	树的问题
英文题目名称	Ball	card	div	tree
文件名	ball.cpp/ball.c	card.cpp/card.c	div.cpp/div.c	tree.cpp/tree.c
输入文件名	ball.in	card.in	div.in	tree.in
输出文件名	ball.out	card.out	div.out	tree.out
每个测试点时限	1 秒	1 秒	1 秒	1 秒
测试点数目	10	20	10	10
每个测试点分值	10	5	10	10
题目类型	传统	传统	传统	传统
内存限制	128MB	128MB	128MB	128MB

注意事项：

1. 每名选手建立一个文件夹，文件夹名称用“英文缩写+中文姓名”命名，如“ssy 段苏航”。
2. 不需要建立子目录。
3. 使用文件输入输出。

1. 精灵球

(ball.cpp/.c)

【题目描述】

你有 N 个精灵球，将它们排成一行，精灵球有红色的和蓝色的。这些精灵球具有魔法力，魔法力的计算方法如下：把精灵球分成最小的部分，每部分是连续的，只包含一种颜色。魔法力是每部分长度的平方和。

具有魔法的你有一次改变一个精灵球颜色机会，当然，你可以选择使用魔法和不使用魔法，这样，精灵球的魔法力可能就改变了。请问：经过你的工作后，可能产生最大的魔法力是多少。

【输入格式】

共 1 行 1 个字符串，只包含字符 ‘R’（表示红色精灵球） 和 ‘B’（蓝色精灵球）。

【输出格式】

共 1 行 1 个整数，表示经过你的操作后，这一行精灵球能产生的最大魔法力。

【输入输出样例】

【输入样例 1】

RRRRBB

【输出样例 1】

26

【样例 1 解释】

将第一个 B 改为 R，那么产生的魔法力为 $5^2 + 1^2 = 26$ 。

【输入样例 2】

RBRB

【输出样例 2】

10

【数据范围】

对于 40% 的数据，字符串长度 N 满足， $1 \leq N \leq 1000$ 。

对于 100% 的数据，字符串长度 N 满足， $1 \leq N \leq 100,000$ 。

2. 球星卡

(card.cpp/.c)

【题目描述】

校园超市出售 NBA 球星卡，对于购买数量的不同收费标准也不一样。比如，购买少于 100 张球星卡，每张 20 金币；如果购买数量大于等于 100，每张 10 金币。显然，想购买 99 张的时候购买 100 张会更省钱。

给出超市球星卡的收费策略，多次询问，求每次询问所需要的最少钱数。

【输入格式】

输入数据共 3 行。

第 1 行包含两个数 n 和 m ，表示价格分类共有 n 种，询问有 m 次。

第 2 行包含 $2n$ 个整数，即 $N_1, P_1, N_2, P_2 \dots N_n, P_n$ ，表示当打印数量不少于 N_i 时，每张收取 P_i 金币。

第 3 行包含 m 个整数 $Q_1, Q_2, Q_3 \dots Q_m$ ，表示想购买 Q_i 张球星卡。

【输出格式】

对于每次询问，输出一行一个整数，表示购买 Q_i 张球星卡最少需要多少金币。

【输入输出样例】

【输入样例】

```
2 3
0 20 100 10
0 99 100
```

【输出样例】

```
0
1000
1000
```

【数据范围】

对于 30% 的数据： $0 < n, m \leq 10^3$ ， $0 = N_1 < N_2 < \dots < N_n \leq 10^5$ ， $10^5 \geq P_1 \geq P_2 \geq \dots \geq P_n \geq 0$ ， $0 \leq Q_i \leq 10^5$ 。

对于 100% 的数据： $0 < n, m \leq 10^5$ ， $0 = N_1 < N_2 < \dots < N_n \leq 10^9$ ， $10^9 \geq P_1 \geq P_2 \geq \dots \geq P_n \geq 0$ ， $0 \leq Q_i \leq 10^9$ 。

3. 整除

(div.cpp/.c)

【题目描述】

给定 n 个正整数 A_1 、 A_2 、.....、 A_n 和一个正整数 m ，求在 $[1, m]$ 之间的整数中，能被 n 个整数至少一个数整除的数的个数。

【输入格式】

输入数据共 2 行。

第 1 行包含 2 个正整数 n 和 m 。

第 2 行包含 n 个正整数 A_1 、 A_2 、.....、 A_n 。

【输出格式】

一行一个整数，表示所求的答案。

【输入输出样例】

【输入样例 1】

```
3 2
2 3 7
```

【输出样例 1】

```
1
```

【样例 1 解释】

$[1, 2]$ 之间只有整数 2 能被 2、3、7 中的 2 整除，所以答案为 1。

【输入样例 2】

```
3 6
2 3 7
```

【输出样例 2】

```
4
```

【样例 2 解释】

$[1, 6]$ 之间 6 能被 2 或者 3 整除，4 能被 2 整除，3 能被 3 整除，2 能被 2 整除，所以答案为 4。

【数据范围】

对于 30% 的数据: $1 \leq N \leq 10, 1 \leq M \leq 10,000$ 。

对于 100% 的数据: $1 \leq N \leq 10, 1 \leq M \leq 1,000,000,000$ 。

4. 树的问题

(tree.cpp/.c)

【题目描述】

一棵树，有 n 个结点，1 号结点为根结点，树中每条边有权值。求树中每个结点能够达到的最远距离（不允许走重复的边），数据保证最远的距离不超过 10^9 。

【输入格式】

输入数据共 2 行。

第 1 行为一个整数 n ，表示结点的个数。

第 2 行到第 n 行，第 i 行两个整数 x 和 w ，表示在树中， x 结点与 i 结点相连， x 和 i 的边权为 w 。

【输出格式】

共 n 行，第 i 行表示结点 i 能够到达的最远距离。

【输入输出样例】

【输入样例 1】

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

【输出样例 1】

```
3
2
3
4
4
```

【数据范围】

对于 10% 的数据是一条链。

对于另 30% 的数据 $n \leq 1,000$ 。

对于 100% 的数据， $5 \leq n \leq 20,000$ 。