

# 选拔考试(2024.6.16)

仔细审题。 有一道英文题，看不懂的同学直接跳过。

编译选项：

C++	-lm -Wl,--stack=2147483647 -std=c++14 -O2
-----	---

注意事项：

- 1. 文件名（程序名和输入输出文件名）必须使用英文小写。
- 2. C++ 中函数 main() 的返回值类型必须是 int，程序正常结束时的返回值必须是 0。
- 3. 若无特殊说明，结果比较方式为忽略行末空格、文末回车后的全文比较。
- 4. 评测时采用的机器配置为：Intel(R) Core(TM) i7-5600U CPU @ 2.60GHz，内存 8GB。上述时限以此配置为准。
- 5. 选手应将各题的源程序放在选手文件夹内，**不要建立子文件夹**。
- 6. 评测使用 Windows 系统，系统为 64 位。
- 7. 开启子任务捆绑

题号	名称	输入文件名	输出文件名	源代码名
A	运算	a.in	a.out	a.cpp
B	方程	b.in	b.out	b.cpp
C	任务	c.in	c.out	c.cpp
D	集合	d.in	d.out	d.cpp
E	括号	e.in	e.out	e.cpp
F	拆分	f.in	f.out	f.cpp
G	平行	g.in	g.out	g.cpp
H	序列	h.in	h.out	h.cpp
I	船运	i.in	i.out	i.cpp

提示：题目基本上是按难度排序的。

## A 【运算】 (a.cpp)

### 题目描述

给出一个形如  $A \text{ op } B$  的字符串，其中  $A, B$  均为一个不含前导零的，在  $[1, 10^9]$  中的正整数。 $\text{op}$  为  $-$  或  $+$ ，代表运算。

请对这个表达式求值，并输出。

### 输入格式

输入的一行包括一个形如  $A \text{ op } B$  的字符串。

### 输出格式

输出的一行包括唯一的一个整数，表示答案。

### 数据范围

- $1 \leq A, B \leq 10^9$

### 样例

#### 样例输入 1

```
1 + 2
```

#### 样例输出 1

```
3
```

#### 样例输入 2

```
5 - 7
```

#### 样例输出 2

```
-2
```

## B 【方程】 (b.cpp)

### 题目描述

给出  $K, S$ , 求关于  $X, Y, Z$  的方程  $X + Y + Z = S, 0 \leq X, Y, Z \leq K$  的解的组数。

### 输入格式

输入的唯一的一行, 包括由空格分隔的两个整数, 分别代表  $K, S$ 。

### 输出格式

输出一行, 表示答案。

### 数据范围

- $2 \leq K \leq 2500$
- $0 \leq S \leq 3K$

### 样例

#### 样例输入 1

```
2 2
```

#### 样例输出 1

```
6
```

#### 样例解释 1

由一个三元组分别表示  $X, Y, Z$ , 则可行的组为

- $(0, 0, 2)$
- $(0, 2, 0)$
- $(2, 0, 0)$
- $(0, 1, 1)$
- $(1, 0, 1)$
- $(1, 1, 0)$

#### 样例输入 2

```
5 15
```

#### 样例输出 2

```
1
```

## C 【任务】 (c.cpp)

有  $n$  个任务，编号为  $1 \sim n$ ，需要被完成。

假设：任务  $i$  只能在时刻  $r_i$  后开始处理。任务  $i$  需要  $p_i$  个时间单位才能完成。同一个时间单位只能处理一项任务；一项任务可分多次处理（可暂停，之后继续处理） 设计方案使得  $\sum t_i$  最小， $t_i$  为任务  $i$  被完成的时刻。

输入格式：

$n$

$r_1 \ r_2 \ \dots \ r_n$

$p_1 \ p_2 \ \dots \ p_n$

输出格式：

一个数，表示最小的  $\sum t_i$ 。

数据范围：

$1 \leq n \leq 100$

$0 \leq r_i \leq 200, \quad 1 \leq p_i \leq 200$

示例

输入样例 1

3

1 2 3

4 2 2

输出样例 1

19

输入样例 2

5

5 8 8 6 0

8 6 6 1 3

输出样例 2

70

## D 【集合】 (d.cpp)

有一个 $n \times m$ 的迷宫，每个格子不是平地就是障碍物，迷宫的四周（上、下、左、右边界）都是障碍物。有 $p$ 个机器人，全都站在平地上。

你可以向所有机器人发布同样的指令序列，每一条指令都是 $NSWE$ 之一，告诉机器人向某个方向前进。N表示向上，S表示向下，W表示向左，E表示向右。

如果某个机器人能够往该方向前进（即不碰到障碍物）则向该方向移动一格，否则原地不动。要求用**较少**的指令数集结机器人——即让他们站到一块儿去。

### 输入：

第一行， $n, m \leq 50, p \leq 20$

第二行 $2p$ 个整数， $x_1, y_1, x_2, y_2, \dots, x_p, y_p$ ，分别描述 $p$ 个机器人初始位置。

( $1 < x < n, 1 < y < m$ )

接下来一个 $n \times m$ 的01矩阵，描述迷宫(有空格)：0代表为平地，1代表障碍物。

第1行、第 $n$ 行、第1列、第 $m$ 列保证都是1。

### 输出：

一个长为 $T$ 的 $ESWN$ 序列。序列长度**不能超过32768**。

要求所有机器人按着这个序列执行后到达同一格。

### 示例：

Robots.in

```
6 6 4
2 2 2 5 5 5 5 2
1 1 1 1 1 1
1 0 0 1 0 1
1 1 0 0 0 1
1 0 0 0 1 1
1 0 1 0 0 1
1 1 1 1 1 1
```

Robots.out

```
9
ESWSNWSWN
```

## E 【括号】 (e.cpp)

减法是不满足结合律的, 例如  $(5 - 2) - 1 = 2$ , 但是  $5 - (2 - 1) = 4$ , 因此  $(5 - 2) - 1 \neq 5 - (2 - 1)$ 。这意味着形式如  $5 - 2 - 1$  的式子的值依赖于减法的顺序。一般情况下, 没有括号是我们从左到右运算, 即  $5 - 2 - 1$  等价于  $(5 - 2) - 1$ 。

给定一个表达式的形式:

$$x_1 + / - x_2 + / - \dots + / - x_n$$

其中每一个  $+/-$  表示  $+$  或  $-$ ,  $x_1, x_2, \dots, x_n$  表示不同的变量。在表达式  $x_1 - x_2 - \dots - x_n$  中, 我们要插入  $n - 1$  对括号来清晰的表示运算的顺序, 同时, 必须保证它等价于给定的表达式。

例如, 如果我们要使它等价于表达式:

$$x_1 - x_2 - x_3 + x_4 + x_5 - x_6 + x_7$$

我们可以在  $x_1 - x_2 - x_3 - x_4 - x_5 - x_6 - x_7$  中这样插入括号:

$$(((x_1 - x_2) - ((x_3 - x_4) - x_5)) - (x_6 - x_7))$$

注意: 我们只注意有正确且完全括起的表达式。正确且完全是指: 单个变量或者是表达式  $(w_1 - w_2)$ , 其中  $w_1$  和  $w_2$  是正确且完全括起的表达式。通俗一点, 我们对含有  $()$ ,  $(x_i)$ ,  $((\dots))$  的式子不感兴趣。再例如表达式  $x_1 - (x_2 - x_3)$  没有完全括起因为它缺少最外层的括号。

要求, 对所有上述所描述的不同的括号插入方式计数, 对  $10^9$  取模。

### 输入格式

第一行输入一个正整数  $n$ , 接下来  $n - 1$  行, 输入一个字符  $c_i$ , 满足  $c_i \in +, -$ , 意义为表达式中位于  $x_{i-1}$  与  $x_i$  之间的符号。

### 输出格式

第一行输出一个整数, 表示答案。

### 样例

#### 输入样例

```
1 | 7
2 | -
3 | -
4 | +
5 | +
6 | -
7 | +
```

#### 输出样例

```
1 | 3
```

## 数据规模与约束

对于  $5pts$  的数据, 满足  $n \leq 5$ 。

对于另外  $15pts$  的数据, 满足  $n \leq 500$ 。

对于  $100pts$  的数据, 满足  $n \leq 5000$

## F 【拆分】 (f.cpp)

正整数  $N$  可以拆分成若干个 2 的非负整数幂的和, 比如  $10=2^0+2^0+2^1+2^1+2^2$ , 请问一共有多少种拆分的方案? 如果两个表示法的差别仅在于他们的各个数相加的次序不同, 则这两个表示法就被视为相同的。

输入:  $N$

输出: 方案数。

示例:

输入样例

4

输出样例

4

说明: 方案为 4;  $2+2$ ;  $2+1+1$ ;  $1+1+1+1$ 。

$N \leq 10000$

有 60% 的数据  $N \leq 1000$

## G 【平行】 (g.cpp)

You are given two lines  $L, L'$  intersecting at origin  $O$ , and a set of points  $S=\{s_1, s_2, \dots, s_n\}$  on the plane. You want to select four points  $A, B, A', B'$  with the following properties:

1.  $A$  lies on  $L$  and  $A'$  lies on  $L'$ .
2.  $B, B'$  both lie in  $S$ ; i.e.  $B \in S$  and  $B' \in S$ .
3. The midpoint of  $A, A'$  coincides with the midpoint of  $B, B'$ .

This implies that  $ABA'B'$  is a parallelogram (or a degenerated parallelogram).

4. The area of parallelogram  $ABA'B'$  is maximized.

### Input:

The first line contains four integers  $a, b, a', b'$ , which respectively describe the equations of  $L$  and  $L'$ . Specifically, Line  $L$  has the equation  $ax+by=0$ ; line  $L'$  has the equation  $a'x+b'y=0$ .

The second line contains one integer  $n$ .

The next  $n$  lines contain  $n$  pair of coordinates, which represent the  $n$  points in  $S$ .

### Constraints:

$$n \leq 10^6$$

Other inputs are integers with absolute value less or equal to  $10^4$ .

$$ab' \neq ba', \quad a^2+b^2 > 0, \quad a'^2+b'^2 > 0$$

### Output:

Output a real number, which is the maximal area you can get.

### Example input:

1 0 0 1

5

1 1

-1 1

-1 -1

1 -1

0 0

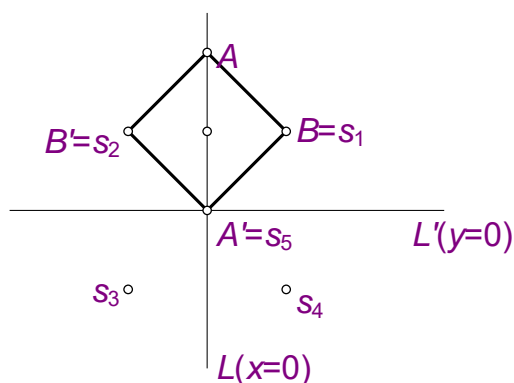


Figure 1 Illustration of the example

### Example output:

2

本题要所有数据全部通过才能拿分。



## H 【序列】 (h.cpp)

小P找到了小T玩，他们在玩数列。

小T给了小P一个正整数序列，小P将它划分成了若干个连续子段，使得每段的元素都严格递增，并且小P很懒，所以他找到了划分段数最少的划分方案。

但是现在他一不小心把序列打乱了，小T很生气。为了安慰小T，他决定找回原来的序列。但他只记得每个元素出现了多少次，以及最后算出的最少划分段数。

现在他想知道，有多少种序列符合他的记忆呢？请求出这个值对  $10^9 + 7$  取模的结果。由于小P记忆力不太好，所以他可能记忆有误，此时应当输出 0。

形式化地说，给定序列  $\{a_i\}$ ，表示在原序列中  $i$  出现了  $a_i$  次，再给定  $k$ ，表示将原序列划分为最少的连续严格递增子段的个数，求可能的原序列个数，对  $10^9 + 7$  取模。

### 输入格式

第一行两个整数  $n, k$  分别表示原序列值域与最少划分段数。

第二行  $n$  个整数，第  $i$  个整数  $a_i$  表示原序列中  $i$  的出现次数。

### 输出格式

一行一个整数，表示方案数对  $10^9 + 7$  取模的结果。

#### 样例 #1

输入

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

输出

```
1 4
```

样例解释

有 4 种情况：

1.  $\{1, 3, 2\}$ ，划分为  $\{1, 3\}, \{2\}$
2.  $\{2, 1, 3\}$ ，划分为  $\{2\}, \{1, 3\}$
3.  $\{2, 3, 1\}$ ，划分为  $\{2, 3\}, \{1\}$
4.  $\{3, 1, 2\}$ ，划分为  $\{3\}, \{1, 2\}$

样例 #2

输入

1	5 20
2	3 2 11 5 7

输出

1	474640725
---	-----------

样例 #3

输入

1	2 30
2	28 14

输出

1	768379691
---	-----------

数据范围与约定

对于所有数据，满足  $1 \leq n \leq 50, 1 \leq k \leq 2500, 1 \leq a_i \leq 50$

子任务1（5分）：满足  $n \leq 2$

子任务2（20分）：满足  $n \leq 5, \sum a_i \leq 10$

子任务3（25分）：满足  $\sum a_i \leq 20$

子任务4（50分）：无附加限制

## I 【船运】 (i.cpp)

上海船运公司每天都有千万计的从海外进口的集装箱需要从上海的码头中转到长江沿岸各城市，比如南京，武汉，重庆，等等。你需要编写程序来帮助码头进行调度工作，将所有集装箱分给若干货船并由它们运到目的地。

假定某天总共有  $n$  个集装箱  $C_1, C_2, \dots, C_n$  等待发货。集装箱  $C_i$  有一个目的地，它位于长江上游距离上海  $s_i$  米的地方。集装箱  $C_i$  有一个重量  $w_i$  公斤。

编号越小的集装箱需要越早发货，即，集装箱必须根据编号从小到大的顺序被安排进货船。若安排  $C_i$  到  $a$  号货船而  $C_{i+1}$  到  $b$  号货船，需要保证  $b \geq a$ 。

货运公司每艘货船的荷载为  $w_0$  公斤；货船上的集装箱总重不得超过  $w_0$  公斤。

如果某条货船负责运输  $C_i \sim C_j$  这些货物，它需要跑的里程为  $2 \max(s_i, \dots, s_j)$  米，因为它需要保证将最远的那个货物（以及其他所有货物）送到然后折返。

公司拥有足够多货船。请问如何装载货物进船使得所有船的总里程最短？

### 输入格式

$n \ w_0$

$s_1 \ \dots \ s_n$

$w_1 \ \dots \ w_n$

### 输出格式

总里程

### 样例输入

8 8

3 5 1 3 2 1 2 2

2 3 2 1 2 2 3 2

### 样例输出

18

### 样例解释

$C_1 \sim C_4$  一条船;  $C_5 \sim C_7$  一条船;  $C_8$  一条船。总里程:  $2*(5+2+2)=18$ 。

### 数据范围

$1 \leq n \leq 6 * 10^6$ 。  $w_0 \leq 10^{15}$ 。  $s_1, \dots, s_n \leq 10^9$ 。  $w_1, \dots, w_n \leq 10^9$ 。全都非负。

保证  $w_1, \dots, w_n$  都不超过  $w_0$ 。

共 40%, 20% 的数据 满足  $n \leq 10^5$ ,  $n \leq 10^4$ 。

此外, 还有 10% 的数据 满足  $n \leq 10^6$  且  $w$  和  $s$  是随机的。