

三五堂

C++竞赛课程

高阶挑战练习



为适应考试需要，以下所有题目都要求用文件输入输出，每道题目都给定了输入输出的文件名，请严格按照这个规则输入输出。所有的题目都要求 1s 内执行完毕。

第1题 流水 (water.cpp)

给你一个数字矩阵代表一个大陆大陆，每一个格子的位置的数字代表这个位置的高度。
Pacific ocean 位于这个矩阵的左上，Atlantic ocean 位于这个矩阵的右下。如下图：

Pacific ocean					
	1	2	2	3	5
Pacific	3	2	3	4	4
ocean	2	4	5	3	1
	6	7	1	4	5
	5	1	1	2	4
Atlantic ocean					

水可以四个方向流动，上下左右，水可以从一个格子流入另外一个高度小于等于他的格子。找出哪些位置如果有水，可以同时流动到 Pacific ocean 和 Atlantic ocean。

输入格式(输入文件 water.in):

第一行两个整数 m 和 n ，表示矩阵大陆的行数和列数。 $1 \leq m, n \leq 200$

第二行开始 m 行，每行 n 个大于等于 1 并且小于 10^5 的整数，表示当前行的 n 个整数。

输出格式(输出文件 water.out):

输出水可以同时流动到 Pacific ocean 和 Atlantic ocean 的位置的行和列。每一个位置两个整数分别表示行和列，空格分开。每一个位置一行。注意：行和列从 0 行 0 列开始。

示例输入：

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

示例输出：

```
0 4
1 3
1 4
```

2 2

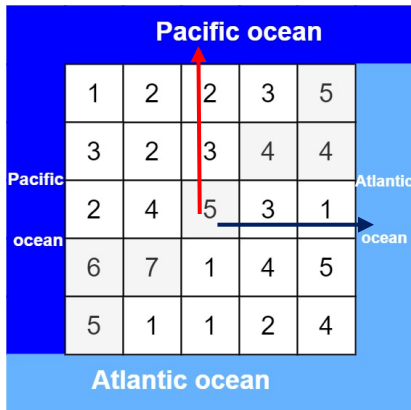
3 0

3 1

4 0

示例说明：

输入矩阵如下图：



位置(0,4)和(4,0)在两个 ocean 的边上，显然满足条件。

位置(2,2)的高度是 5，可以通过位置 (1, 2) ,(0,2) 流入上面 Pacific ocean，也可以通过位置(2,3),(2,4)流入 Atlantic Ocean,如上图箭头所示，经过的路线上的位置的高度不会大于前一个位置。

注意：这里 m 和 n 都是最大 200，如果你的算法时间复杂度是 $O(m^2 \cdot n^2)$ ，肯定是会超时的。你有没有更快的，比如 $O(m \cdot n)$ 的算法？

第2题 求零序列 subzero.cpp

给定一个从 1 到 n 的数字序列 ($n \leq 9$) 1,2,3... n ，在每一对两个数字之间可以插入 +-, 或者空格 (两个数字合成一个新的数)，得到一个新的表达式序列。 请输出所有结果为 0 的表达式序列。

输入格式 (文件名: subzero.in)

一个整数 n , $3 \leq n \leq 9$

输出格式 (文件名: subzero.out) :

按照 ASCII 顺序, 输出在 1 到 n 的数字序列中每一对数字中间插入一个 +-, 或者空格得到的新的表达式计算结果为 0 的序列。

输入样例:

7

输出样例:

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

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

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

1-2 3-4 5+6 7

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

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

样例说明:

1+2-3+4-5-6+7=0

1+23-4+5+6-7=0

1-23+4+5+6+7=0

1-23-45+67=0

1-2+3+4-5+6-7=0

1-2-3-4-5+6+7=0



第3题 买东西 buy.cpp

Tom 妈妈给了他 n 元钱买东西, Tome 想买的物品很多, 他给每一个想买的物品规定了一个 1 到 5 之间的重要度, 还查到了每一个物品的价格。现在他想不超过这 n 元购买最合适的物品, 所谓最合适就是买到的所有物品的价格*重要度, 加起来的和最大。每一个物品只能购买 1 件。

输入格式: (输入文件 buy.in)

第 1 行为两个正整数 n 和 m , 用一个空格隔开, 其中 n (<30000) 表示总钱数, m (<25) 为希望购买物品的个数。

从第 2 行到第 $m+1$ 行, 第 i 行给出了编号为 $i-1$ 的物品的基本数据, 每行有 2 个非负整数

v 和 p , 其中 v 表示该物品的价格($v \leq 10000$), p 表示该物品的重要度(1~5))

输出格式: (输出文件 buy.out)

一个正整数, 为不超过总钱数 n 可以购买到的物品的价格与重要度乘积的总和的最大值 (<100000000)。

输入样例 1:

1000 5

800 2

400 5

300 5

400 3

200 2

输出样例 1:

3900

样例说明:

Tom 可以购买第 3, 4, 5 件物品, $300*5+400*5+200*2=3900$ 这时候总共花费了 900 不超过 1000 元。这个是最大的。

第4题 特殊二叉树 bintree.cpp

给出 n ($1 \leq n \leq 1000$) 个不重复的整数，每一个整数大于 1 并且小于 10 的 9 次方。用给出的这些整数构造一个特殊的二叉树，每一个整数可以被使用任意次。这个二叉树的要求是：所有非叶子节点的值等于他的孩子节点的值的乘积。

问一共多少种构造方式，结果对 10^9+7 取模。

输入格式：（输入文件 bintree.in）

第 1 行为 1 个正整数 n ，表示给出的整数的个数。

第 2 行给出 n 个正整数，空格分开。

输出格式：（输出文件 bintree.out）

一个正整数，表示一共多少构造方式。

输入样例 1:

4

2 4 8 10

输出样例 1:

9

样例说明:

2 4 8 10 这 4 个整数可以单独构成 1 个节点数为 1 的树。

节点 4 为根，可以左右都是 2，新的 1 个。所以根为 4 一共 2 个树。

节点 8 为根，左节点是 2，右节点是 4，一共 2 个树，左右节点交换，还有 2 个，所以根为 8 的一共 5 个树。

节点 2，10 只有各自自己单独构成的 1 个树。

总共 $2+5+1+1=9$ 个树。