Day 3

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 题目名称 | 数字三角形 | 合作 | 彩虹 | Flowfree |
| 可执行文件名 | triangle | eliminate | rainbow | flowfree |
| 输入文件名 | triangle.in | eliminate.in | rainbow.in | flowfree.in |
| 输出文件名 | triangle.out | eliminate.out | rainbow.out | flowfree.out |
| 每个测试点时限 | 1秒 | 1秒 | 1秒 | 1秒 |
| 内存限制 | 256MB | 256MB | 256MB | 256MB |
| 测试点数目 | 10 | 20 | 20 | 20 |
| 每个测试点分值 | 10 | 5 | 5 | 5 |
| 是否有部分分 | 否 | 否 | 否 | 否 |
| 题目类型 | 传统型 | 传统型 | 传统型 | 传统型 |
| 是否有样例文件 | 否 | 否 | 否 | 否 |
| 是否有附加文件 | 否 | 否 | 否 | 否 |

提交源程序须加后缀

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 对于C++ 语言 | triangle.cpp | eliminate.cpp | rainbow.cpp | flowfree.cpp |

编译开关

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 对于C++ 语言 | -O2 -lm | -O2 -lm | -O2 -lm | -O2 -lm |

nodesnodesnodes1.in~nodes10.innodes1.out~nodes10.outnodes1.ans~nodes10.ans

数字三角形(Triangle)

【问题描述】

现在，你有一个𝑁行的数字三角形，你从三角形的最上方出发，每次可以朝左下或者右下走，你需要想办法最大化你所经过的格子的数字之和。

1

2 3

3 3 1

如图所示为一个数字三角形，你的最大收益是从最上方出发，采取右下然后左下的行动，这样你会获得7的收益。

然而，这个问题过于简单了，狡猾的出题人想要你输出非严格第K大的收益是多少（非严格解释见样例1）。

【输入格式】

输入文件为***triangle.in***。

第一行两个整数 𝑁,𝐾，代表三角形的大小和所需要求的值。

接下来𝑁行，第𝑖行有𝑖个整数，代表数字三角形第𝑖行的数字。

【输出格式与部分分】

输出文件为***triangle .out***。

输出仅一行一个整数，代表不严格第𝐾大的答案。

【样例1输入】

3 2

1

1 1

1 2 1

【样例1输出】

4

【样例1解释】

注意，这个三角形中有两种方案能够取得4的价值，所以这算两种方案，所

以答案为 4。

【样例2输入】

3 1

1

2 3

3 3 1

【样例2输出】

7

【子任务】

所有测试数据的范围和特点如下:

对于的数据有

对于的数据有

对于的数据有 。

对于的数据有 ，数字三角形中数字大小绝对值不超过,保证有解。

合作(Eliminate)

【问题描述】

国际上有个国家合作登陆火星，其中每个国家有位宇航员，每位宇航员有自己的性格。现在，各国需要各选出一名宇航员登陆火星。长时间宇宙航行中和谐很重要，这就要求所有被选出的名宇航员性格**极差**尽可能小，问最小的极差是多少。

极差即序列最大值减去最小值。

【输入格式】

输入文件为***eliminate.in***。

输入文件的第一行包含一个正整数 。

接下来行，每行首先是一个整数表示第个国家的候选宇航员数量，之后个数表示这个国家各个宇航员的性格。

【输出格式与部分分】

输出文件为 ***eliminate .out***。

输出一个整数，即最小极差。

【样例1输入】

2

1 6

3 -7 7 10

【样例1输出】

1

【样例2输入】

4

9 -5 -9 2 8 5 4 3 3 8

2 10 8

1 -7

3 1 6 10

【样例2输出】

15

【样例3输入】

1

1 9

【样例3输出】

0

【子任务】

所有测试数据的范围和特点如下:

对于的数据有

对于的数据有 。

对于的数据有

彩虹(Rainbow)

【问题背景】

彩虹是有着许多平行彩色条纹组成的漂亮图案。如下矩阵是彩虹矩阵的一种：

形式化地，对于的**方矩阵**A，若其满足：

* 对于任意且，有
* 对于任意且，有

则称矩阵为彩虹矩阵。

现在已知一个的矩阵，请问一共有多少个彩虹子矩阵。

【输入格式】

输入文件为rainbow.in。

输入文件的第一行包含两个正整数 ，用空格隔开，表示行数和列数。

接下来 行，每行 个数，表示。

【输出格式】

输出文件为rainbow***.out***。

输出文件只有一行，包含一个正整数，即的彩虹子矩阵个数。

【样例1输入】

2 2

1 2

3 1

【样例1输出】

5

【样例2输入】

2 2

1 2

2 1

【样例2输出】

4

【样例3输入】

3 3

1 2 3

4 1 2

5 4 1

【样例3输出】

14

【子任务】

所有测试数据的范围和特点如下:

对于的数据有

对于的数据有

对于的数据有

对于的数据有

Flowfree(Flowfree)

【问题背景】



FlowFree是这样一款游戏：你需要将图中同颜色的点对用上图的方式连接起来，要求不同点对之间路径不相交，且最后你的路径需要覆盖整个棋盘，求方案。

【输入格式】

输入文件为flowfree.in。

第一行两个正整数，代表地图的大小是并且有个点对。

接下来行，每行四个整数，代表点对的第一个点在行列，第二个点在行列。

【输出格式】

输出文件为flowfree***.out***。

输出要求输出任意一种方案按照颜色的顺序输出。

对于某种颜色首先输出一行代表这种颜色的路径长度。接下来行,每行两个整数代表路径上的一个点

请按照路径顺序输出。

【样例1输入】

2 2

1 1 2 1

1 2 2 2

【样例1输出】

2

1 1

2 1

2

2 2

1 2

【子任务】

所有测试数据的范围和特点如下:

对于的数据有

对于的数据有

对于的数据有