

水题大赛B卷：搜索

2022. 6. 18 上午 8:20-11:50



预祝各位虐场成功！

注意：

1. 评测在 windows 下，行未必加回车表示输出完成。
2. **仔细读题，不要跳过任何一个字**，坠吼先通读一遍 pdf。
3. 不准喊“这套题好难啊”，影响别人 AK。
4. 可以喊“这套题好水啊”“我要 ak 辣”，不过 ak 之后不要和别人谈笑风生，影响他人爆零。
5. 个人感觉本套题难度大概小学毕业也是可以做的，毕竟能用到的数学思维是小学就会的，而且还有原题可以水过（除非你读了个假的小学... 或者集训你没好好听课）。

宝岛 (island.cpp/.c)

限制：1S 256MB

【问题描述】

小哼通过秘密方法得到一张不完整的钓鱼岛航拍地图。钓鱼岛由一个主岛和一些附属岛屿组成，小哼决定去钓鱼岛勘探。下面这个 10×10 的二维矩阵就是钓鱼岛的航拍地图。图中数字 0 表示海洋，非 0 数字表示陆地。小哼的飞机将会降落在岛屿某处，你的任务是帮助小哼统计共有多少个独立岛屿，并且每个岛屿的面积是多大？

岛屿的面积是指该岛屿所有格子的数值之和。

1	2	1	0	0	0	0	0	2	3
3	0	2	0	1	2	1	0	1	2
4	0	1	0	1	2	3	2	0	1
3	2	0	0	0	1	2	4	0	0
0	0	0	0	0	0	1	5	3	0
0	1	2	1	0	1	5	4	3	0
0	1	2	3	1	3	6	2	1	0
0	0	3	4	8	9	7	5	0	0
0	0	0	3	7	8	6	0	1	2
0	0	0	0	0	0	0	0	1	0

【输入格式】

输入文件名为 **island.in**。

输入数据的第一行，包含两个正整数 n 和 m ，它们之间用一个空格隔开，表示地图有 n 行 m 列。

接下来的 n 行 m 列，是由大于等于 0 的整数组成的二维阵列，表示钓鱼岛的航拍地图。每一行的相邻整数之间用一个空格隔开。

【输出格式】

输出文件名为 **island.out**。

输出文件的第一行包含一个整数，表示岛屿的数目。

输出文件的第二行包含若干用一个空格隔开的整数，这些整数从小到大排列，表示每个岛屿的面积。

【输入输出样例】

island.in	island.out
10 10 1 2 1 0 0 0 0 2 3 3 0 2 0 1 2 1 0 1 2 4 0 1 0 1 2 3 2 0 1 3 2 0 0 0 1 2 4 0 0 0 0 0 0 0 0 1 5 3 0 0 1 2 1 0 1 5 4 3 0 0 1 2 3 1 3 6 2 1 0 0 0 3 4 8 9 7 5 0 0 0 0 0 3 7 8 6 0 1 2 0 0 0 0 0 0 0 0 1 0	4 4 9 19 124

【 数据规模与约定 】

$0 \leq \text{每个格子的整数值} \leq 1000$;

$1 \leq n, m \leq 100$ 。

积木覆盖

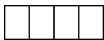
(puzzle.cpp/c)

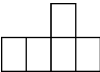
限制：1S 256MB


【问题描述】

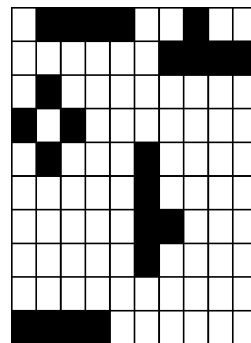
D最喜欢玩积木了。每一次玩时，他都会把积木在地面上摊开，清点积木的个数。假设地面是一个 $n \times m$ 的网格，每个积木都由若干个正方形拼成。每一块地要么是空，要么被某个积木完全覆盖。可以保证，在地面上没有两块积木有公共边的情况。如下图所示，就是一种积木覆盖情况。D希望我们帮助编程求出：对于给定的地面覆盖情况，共有多少块积木。

在右图中，地面共有 8 块积木。

其中，有 2 块 

2 块 

4 块 



【输入格式】 (puzzle.in)

第一行输入两个整数 n 和 m 。

接下来是一个 $n \times m$ 的 0 1 矩阵，1 表示该单位正方形被某个积木覆盖，0 表示没有被覆盖。

【输出格式】 (puzzle.out)

一个整数，表示地面上共有的积木块数。

【样例输入】

```
10 10
0111100100
0000001111
0100000000
1010000000
0100010000
0000010000
0000011000
0000010000
0000000000
1111000000
```

【样例输出】

8

【数据规模】

$1 < n, m \leq 100$

踩方格 ([check.cpp/.c](#))

限制：1S 256MB

有一个方格矩阵，矩阵边界在无穷远处。我们做如下假设：

- a. 每走一步时，只能从当前方格移动一格，走到某个相邻的方格上；
- b. 走过的格子立即塌陷无法再走第二次；
- c. 只能向北、东、西三个方向走；

请问：如果允许在方格矩阵上走 n 步，共有多少种不同的方案。2种走法只要有一步不一样，即被认为是不同的方案。

Input ([check.in](#))

允许在方格上行走的步数 n ($n \leq 20$)

Output ([check.out](#))

计算出的方案数量

Sample

Input	Output
2	7

计数

(count.cpp/.c)

限制: 1s

64MB

【问题描述】

我们要求找出具有下列性质数的个数(包含输入的自然数 n):

先输入一个自然数 $n(n \leq 1000)$, 然后对此自然数按照如下方法进行处理:

1. 不作任何处理;
2. 在它的左边加上一个自然数, 但该自然数不能超过原数的一半;
3. 加上数后, 继续按此规则进行处理, 直到不能再加自然数为止.

【输入格式】

输入一个自然数 n 。

【输出格式】

满足条件的数的个数。

【输入样例】(count.in)

6

【输出样例】(count.out)

6

【输出样例解释】

6 个数分别是:

6

16

26

126

36

136

营救

(save.cpp/ .c)

限制: 1s 128MB

【问题描述】

铁塔尼号遇险了！它发出了求救信号。距离最近的哥伦比亚号收到了讯息，时间就是生命，必须尽快赶到那里。

通过侦测，哥伦比亚号获取了一张海洋图。这张图将海洋部分分化成 $n*n$ 个比较小的单位，其中用 1 标明的是陆地，用 0 标明是海洋。船只能从一个格子，移到相邻的四个格子。

为了尽快赶到出事地点，哥伦比亚号最少需要走多远的距离？

【输入格式】

第一行为 n , 下面是一个 $n*n$ 的 0、1 矩阵，表示海洋地图

最后一行为四个小于 n 的整数，分别表示哥伦比亚号和铁塔尼号的位置。

【输出格式】

哥伦比亚号到铁塔尼号的最短距离，答案精确到整数。

【输入样例】(save.in)

```
3
001
101
100
1 1 3 3
```

【输出样例】(save.out)

```
4
```

【数据范围】

$N \leq 1000$

炸弹人 (bomberman.cpp/.c)

限制：1S 256MB

【问题描述】

“炸弹人”游戏——用放置炸弹的方法来消灭敌人。

现在有一个特殊的关卡如下（图 1）。炸弹人只有一枚炸弹，但是这枚炸弹威力超强（杀伤距离超长，可以消灭杀伤范围内所有的敌人）。杀伤范围是指炸弹爆炸时可以从上、下、左、右四个方向延伸直到碰到墙体为止）。炸弹人从某一指定的空地出发，可以上、下、左、右行走并安置炸弹，但不能穿过墙体和敌人。

你的任务是帮助炸弹人计算出在哪里放置炸弹才可以消灭最多的敌人？

我们先将地图模型化（图 2）。墙用 # 表示，墙是不能被炸掉的，且炸弹的杀伤力是不能穿过墙的。敌人用 G 表示，空地用 . 表示，当然炸弹只能放在空地上。

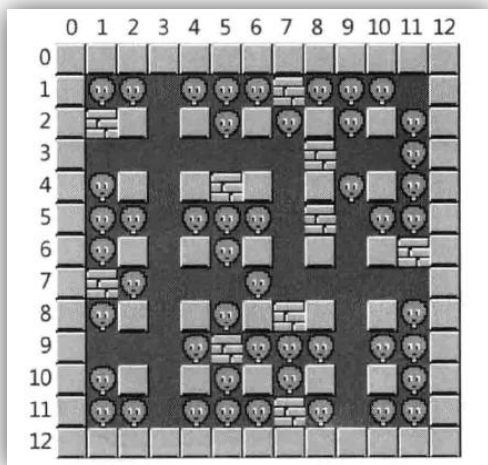


图 1

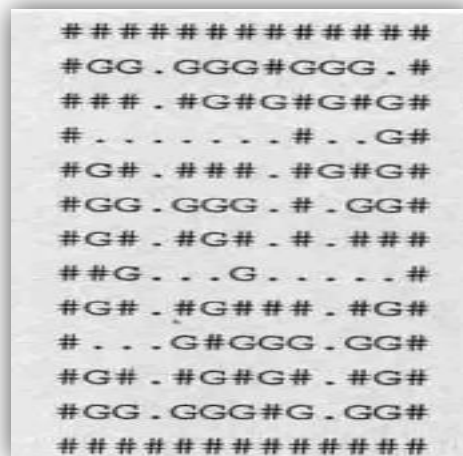


图 2

【输入格式】

输入文件名为 `bomberman.in`。

输入数据的第一行，包含两个正整数 n 和 m ，它们之间用一个空格隔开，表示地图有 n 行 m 列。

输入数据的第二行，包含两个正整数 p 和 q ，它们之间用一个空格隔开，表示炸弹人出发点所在的行列位置，即 p 行 q 列。

接下来的 n 行 m 列，是由若干“#”、“G”、“.”三种字符组成的二维阵列，表示地图。且二维阵列最外面的一圈都是由“#”组成，表示地图四个边都是一圈墙体。

【输出格式】

输出文件名为 **bombberman.out** 。

输出文件有若干行，除最后一行外，每行都包含两个整数，它们之间用一个空格隔开，表示消灭最多敌人时能安放炸弹的所有可能的行、列位置。输出时按照行号从小到大的顺序输出，若行号相同，则按照列号从小到大的顺序输出。

输出文件的最后一行，包含一个整数，表示消灭最多的敌人数目。

【输入输出样例】

bomberman.in	bomberman.out
13 13 2 4 ##### #GG. GGG#GGG. # ###. #G#G#G#G# #. #. . G# #G#. ###. #G#G# #GG. GGG. #. GG# #G#. #G#. #. ### ##G. . . G. # #G#. #G###. #G# #. . . G#GGG. GG# #G#. #G#G#. #G# #GG. GGG#G. GG# #####	10 10 8

【 数据规模与约定 】

对于 30%的数据： $1 \leq n, m \leq 30$ 。

对于 60%的数据： $1 \leq n, m \leq 60$ 。

对于 100%的数据： $1 \leq n, m \leq 100$ 。

城堡问题-A

(castle_A.cpp/.c)
限制 : 1S 256MB

```

  1   2   3   4   5   6   7
#####
1 #   |   #   |   #   |   |   #
   #####---#####---#---#####---#
2 #   #   |   #   #   #   #   #
   #---#####---#####---#####---#
3 #   |   |   #   #   #   #   #
   #---#####---#####---#---#
4 #   #   |   |   |   |   #   #
   #####

```

= Wall
| = No wall
- = No wall

(图 1)

图1是一个城堡的地形图。请你编写一个程序，计算城堡一共有多少房间，最大的房间有多大。城堡被分割成 $m \times n$ ($m \leq 50, n \leq 50$) 个方块，每个方块可以有 0~4 面墙。

Input (castle_A.in)

程序从标准输入设备读入数据。第1、2行每行1个整数，分别是南北向、东西向的方块数。在接下来的输入行里，每个方块用一个数字 ($0 \leq p \leq 50$) 描述。用一个数字表示方块周围的墙，1表示西墙，2表示北墙，4表示东墙，8表示南墙。每个方块用代表其周围墙的数字之和表示。城堡的内墙被计算两次，方块(1,1)的南墙同时也是方块(2,1)的北墙。输入的数据保证城堡至少有两个房间。

Output (castle_A.out)

输出2行，每行一个数，表示城堡的房间数、城堡中最大房间所包括的方块数。结果显示在标准输出设备上。

Sample

Input	Output
4 7 11 6 11 6 3 10 6 7 9 6 13 5 15 5 1 10 12 7 13 7 5 13 11 10 8 10 12 13	5 9

道路

(road.cpp/.c)

限制：1S 256MB

Bob 和 Alice 是一对恋人，曾经住在城市 1。Bob 在发现 Alice 在他们喜欢玩的纸牌游戏中作弊后，他决定与她分手，并计划搬到城市 N。

N 个以数字 1... N 命名的城市之间有单向道路相连。每条道路都有两个与之相关的参数：道路长度和道路的通行费。

我们想帮助 Bob 找到他支付得起通行费用且从城市 1 到城市 N 的最短路径。

输入 (road.in)

输入的第一行包含整数 K， $0 \leq K \leq 10000$ ，是 Bob 在路上可以花费的最大金额。

第二行包含整数 N， $2 \leq N \leq 100$ ，城市总数。

第三行包含整数 R， $1 \leq R \leq 10000$ ，道路总数。

以下 R 行中的每一行，用整数 S、D、L 和 T 来描述一条道路：

- S 为源城市， $1 \leq S \leq N$
- D 是目的地城市， $1 \leq D \leq N$
- L 是道路长度， $1 \leq L \leq 100$
- T 是通行费， $0 \leq T \leq 100$

请注意，不同的道路可能具有相同的源城市和目的地城市。

输出 (road.out)

输出从城市 1 到城市 N 的最短路径的总长度，其总通行费小于或等于 K。

如果这样的路径不存在，则输出 -1。

样例

输入	输出
5 6 7 1 2 2 3 2 4 3 3 3 4 2 4 1 3 4 1 4 6 2 1 3 5 2 0 5 4 3 2	11