

北京市集训队 NOI 模拟试题

Day 4——图论专题

试题	派发请柬	κ 取方格数	瞬间转移
文件名称	invite	knumber	transfer
测试时限	1s	1s	1s
最大可用内存	128M	64M	64M
测试点的个数	5	10	5
本题满分	100	100	100

派发请柬(invite.c/cpp/pas)

题目描述

清华百年校庆，举国欢腾，光是临时舞台就直接干掉三千万。当然清华的领导们不希望这么美好的晚会没有人看，于是找到了学生志愿者们，希望让他们负责派发请柬来让更多的人参加百年校庆盛典。

每个学生志愿者将被分配到一个指定的服务站，并那里待一整天，给经过大爷大妈们发请柬。学生们由于接受了党和国家的洗礼，懂得如何成功说服各个年龄段的男男女女参加晚会（尽管有可能到时候连站的地方都没有）。

为志愿者们提供的校车很特殊：所有路线都是单向连接两个服务站。如果有志愿者需要乘车，校车立刻前往站点搭载志愿者飞速前往终点站。到达终点站后它们便空车光速返回到始发站，等待下一次出发。两站之间的费用在一张表中给出。所有的清华志愿者们会在每天早晨（其实往往是中午了）离开温暖的紫荆园到一个领导已经指定好的服务站工作。由于清华学子的能力高超，每个服务站只需要一名志愿者就足以应付各路人马。一天结束时，所有志愿者会乘车返回。他们觉得每天开销实在巨大，不符合科学发展观，于是他们找到了你，想请你编写一个程序，计算志愿者们每天一共最少花掉多少路费。

输入描述

第一行输入两个数 N 和 M ，其中 N 表示车站数（当然包括紫荆园）， M 表示校车线路的条数。接下来 M 行每行 3 个数，分别代表 M 条路线的起点、终点和费用。

输出描述

输出一个数，表示志愿者们每天最少的总花费。

样例输入 `invite.in`

```
4 6
1 2 10
2 1 60
1 3 20
3 4 10
2 4 5
4 1 50
```

样例输出 `invite.out`

```
210
```

数据规模

对于 40% 的数据， $1 \leq N, M \leq 20$

对于 100% 的数据， $1 \leq N, M \leq 1000000$

K 取方格数(knumber.c /cpp/pas)

题目描述

Bella 在 NOIp08 遇到了“传纸条”这道题，当时没有听说过二取方格数这个经典题，而这道题恰巧是一模一样的题，于是果断悲剧。什么，你不知道二取方格数？这个问题是这样的：有一个 $N \times N$ 大小的矩阵，每个位置上都有一个非负整数。Bella 从 $SUM=0$ 开始，每次都是从矩阵的最左上角走到右下角位置，每次移动只能向右移动或者向下移动。每次移动到某个方格，Bella 将方格中的数字加到 SUM ，并将该位置上的数字置零。二取方格数就是求在这样的一个要求下从左上角出发走两次的 SUM 的最大值。Bella 在后来瞬秒了这道题，但还是觉得意犹未尽，如果是走 K 次而不只是两次呢？她找到了聪明的你，帮她求出 K 取方格数的答案。

输入描述

测试数据的第一行为两个整数 N 和 K 。接下来有 N 行，每行 N 个数，矩阵中的元素都不超过 1000。

输出描述

输出 K 取方格数的最大 SUM 值。

样例输入 knumber.in

```
3 2
1 2 3
0 2 1
1 4 2
```

样例输出 knumber.out

```
15
```

数据规模

Case 1: $1 \leq N \leq 10$, $K=1$
Case 2: $1 \leq N \leq 10$, $K=2$
Case 3,4: $1 \leq N \leq 50$, $K=2$
Case 5,6: $1 \leq N \leq 10$, $K \leq 10$
Case 7~10: $1 \leq N \leq 50$, $K \leq 10$

瞬间转移(transfer.c/cpp/pas)

题目描述

在很久以前玩的红警游戏中，可以对游戏中的物体执行一种魔法功能，称为瞬间转移，当一种物体使用这种功能时，它可以瞬间移动到指定的位置，不管有多远。

现在有一个矿区，你驾驶一辆采矿的矿车。你的任务是采集到最大数量的矿。矿区是一个长方形的区域，包含 $n \times m$ 个小方格，有些放各种藏有矿石，其他方格中没有。矿区采完后不能再生。采矿车的起始位置为区域的西北角，它只能移动到东面或者南面相邻的方格，而不能移动到北面或西面的相邻方格。其中有些方格有魔法功能，它能将矿车瞬间移动到指定方格。然而作为矿车的驾驶员，你可以决定是否使用这种魔法功能。如果某个方格有魔法功能，则这个功能永远不会消失，你可以在到达任意一个此类方格时使用魔法功能。

输入描述

第一行为两个整数 N 和 M ，接下来有 N 行描述了矿区的地图，每行为包含 M 个字符的字符串，每个字符可能为数字字符 x ($'0' \leq x \leq '9'$)， $'*'$ 或 $'\#'$ 字符。整数字符 x 表示该方格中有 x 单位的矿石，你的采矿车可以全部采集， $'*'$ 字符表示该方格有魔法功能， $'\#'$ 字符表示该方格布满了岩石，采矿车不能通过。假定起始方格不会是 $'\#'$ 。假设地图有 K 个 $'*'$ 字符，则接下来有 K 行，描述了每个 $'*'$ 将采矿车移动到指定的方格， $'*'$ 的顺序为从北到南、从西到东。(起点在西北角，坐标方向为南-北，西-东，方格的坐标从 0 开始计数)

输出描述

输出可以采集到的最多矿石。

样例输入 transfer.in

```
2 2
11
1*
0 0
```

样例输入 transfer.out

```
3
```

数据规模

对于 40% 的数据： $1 \leq n, m \leq 30$

对于 100% 的数据： $1 \leq n, m \leq 100$