

NOIP2018 提高组模拟赛 Day1

一、题目概况

| | | | |
|----------|----------|----------|----------|
| 中文题目名称 | 铃仙的红色之瞳 | 永琳的竹林迷径 | 辉夜的夜空明珠 |
| 目录名 | eyes | path | moon |
| 可执行文件名 | eyes.exe | path.exe | moon.exe |
| 输入文件名 | eyes.in | path.in | moon.in |
| 输出文件名 | eyes.out | path.out | moon.out |
| 每个测试点时限 | 1.0 s | 2.0 s | 1.5 s |
| 测试点数目 | 20 | 25 | 10 |
| 每个测试点分值 | 5 | 4 | 10 |
| 样例附加文件 | 无 | 无 | 无 |
| 结果比较方式 | 全文比较 | 全文比较 | 全文比较 |
| 题目类型 | 传统题 | 传统题 | 传统题 |
| 运行内存上限 | 512MB | 512MB | 512MB |
| 是否打开优化开关 | 无 | -O2 | -O2 |

二、提交源程序文件名

| | | | |
|--------------|----------|----------|----------|
| 对于 C++ 语言 | eyes.cpp | path.cpp | moon.cpp |
| 对于 C 语言 | eyes.c | path.c | moon.c |
| 对于 pascal 语言 | eyes.pas | path.pas | moon.pas |

三、注意事项

1. 文件名必须采用英文小写。
2. C/C++中 main 函数的返回值类型必须是 int，程序正常结束时必须返回 0。
3. 评测时可用栈空间设置为内存上限大小。
4. 题目很水，请放心食用。

1. 铃仙的红色之瞳(eyes.cpp/c/pas)

【题目背景】

赤红的月光映在永远亭上，妖异的双瞳凝结了真实与虚幻。

是幻象吗？在虚无与真实之间来回闪跃的是什么？

“狂符——幻视调律！”

“你看到的既是真的，也是假的。”

【题目描述】

为了方便你的预测，铃仙对该符卡进行了改造。

敌方非常强大，可以看作有无限的体力。通过该符卡，铃仙可以释放出子弹，敌方触碰到子弹就会损失一格体力。注意，每次敌方损失体力之后，其位置不会改变。

当敌方和铃仙 x 坐标相同上时，每秒铃仙损失一点体力(敌方的攻击比较特殊，可以对前后都进行攻击)。注意，这里的秒指的是时间间隔，即第 A 秒时候敌方开始与铃仙处于一条直线上，第 B 秒时候敌方离开这条直线，那么铃仙受到的伤害是 $(B-A)$ 点。特殊地，如果刚开始铃仙就和敌方在一条直线上，则第 0 秒~第 1 秒也受到一点伤害。若最后铃仙和敌方在 x 坐标相同的位置或者敌方处于有子弹的位置上时，它们只会在第 $Q-1$ 秒到第 Q 秒时受到伤害，并不会在第 Q 秒到第 $Q+1$ 秒受到伤害。

地图是一个 $n*m$ 的矩形，分别对应的 x 坐标为 $[1,n]$ ， y 坐标为 $[1,m]$ ，给定敌方的初始位置和铃仙的位置（铃仙不会移动），当铃仙发出的子弹超过边界时子弹消失，当敌方越过边界时敌方自动回到初始位置。

现在有 Q 秒，每秒一次操作，可能是铃仙的操作也可能是敌方的操作，可能存在的操作如下：

Ins $x\ y$ 铃仙在 (x,y) 处召唤了一颗子弹(如果原先就有子弹则该操作无效)

Del $x\ y$ 铃仙收回法力，让 (x,y) 处的子弹消失(如果原先没有子弹则该操作无效)

MA U 铃仙将所有子弹上移一格

MA D 铃仙将所有子弹下移一格

MA L 铃仙将所有子弹左移一格

MA R 铃仙将所有子弹右移一格

MB U 敌方上移一格

MB D 敌方下移一格

MB L 敌方左移一格

MB R 敌方右移一格

现在要你计算 Q 秒后敌方损失的体力和铃仙损失的体力。

(注：左移指的是横坐标-1，下移指的是纵坐标-1)

【输入格式】

输入文件名为 eyes.in。

第一行一个数表示 Case，表示测试点编号。（样例的编号表示其满足第 Case 个测试点的性质）

第二行两个数表示 n, m ，表示地图大小为 $n*m$ 。

第三行两个数 ax, ay ，表示铃仙的坐标。

第四行两个数 bx, by ，表示敌方的初始坐标。

第五行一个 Q，表示持续时间为 Q 秒，即有 Q 个操作，每秒一个。

下面 Q 行，每行一个操作：

Ins x y 铃仙在(x,y)处召唤了一颗子弹(如果原先就有子弹则该操作无效)

Del x y 铃仙收回法力，让(x,y)处的子弹消失(如果原先没有子弹则该操作无效)

MA U 铃仙将所有子弹上移一格

MA D 铃仙将所有子弹下移一格

MA L 铃仙将所有子弹左移一格

MA R 铃仙将所有子弹右移一格

MB U 敌方上移一格

MB D 敌方下移一格

MB L 敌方左移一格

MB R 敌方右移一格

【输出格式】

输出文件名为 eyes.out。

输出文件有两行：

第一行一个数表示敌方受到伤害损失的体力，第二行一个数表示铃仙受到伤害损失的体力。

【样例 1 输入】

```
1
5 5
2 2
2 5
3
Ins 2 5
MA D
Del 2 4
```

【样例 1 输出】

```
1
3
```

【数据范围与约定】

| 测试点 | n | m | Q |
|-----|-------|-------|--------|
| 1 | <=10 | <=10 | <=10 |
| 2 | | | |
| 3 | | | |
| 4 | | | |
| 5 | | | |
| 6 | | | |
| 7 | <=30 | <=30 | <=300 |
| 8 | | | |
| 9 | | | |
| 10 | | | |
| 11 | <=100 | <=100 | <=10 |
| 12 | | | <=1000 |
| 13 | | | |
| 14 | | | |
| 15 | | | |
| 16 | | | |
| 17 | | | |
| 18 | | | |
| 19 | | | |
| 20 | | | |

对于 100%的数据，

$n \leq 100, m \leq 100, Q \leq 1000$ ，保证操作中 $1 \leq x \leq n, 1 \leq y \leq m$ 。

2. 永琳的竹林迷径(path.cpp/c/pas)

【题目背景】

月下的永远亭里。

“我学会了新的魔法哦！你要试试吗？”

“那就来试试吧！”

永琳瞬间被因幡帝扔到了迷途竹林的外围。

【题目描述】

竹林可以看作是一个 n 个点的树，每个边有一个边长 w_i ，其中有 k 个关键点，永琳需要破坏这些关键点才能走出竹林迷径。

然而永琳打算将这 k 个点编号记录下来，然后随机排列，按这个随机的顺序走过 k 个点，但是两点之间她只走最短路线。初始时永琳会施展一次魔法，将自己传送到选定的 k 个点中随机后的第一个点。

现在永琳想知道，她走过路程的期望是多少，答案对 998244353 取模。

注意，如果对期望不理解，题目最后有详细解释，请自行阅读。

【输入格式】

第一行一个数 Case，表示测试点编号。（样例的编号表示其满足第 Case 个测试点的性质）

下一行一个 n ，表示树的点数。

下面 $n-1$ 行，每行三个数 u_i ， v_i ， w_i ，表示一条边连接 u_i 和 v_i ，长度为 w_i 。

下面一行一个数 k ，表示关键点数。

下面一行 k 个数，表示 k 个关键点的编号。

【输出格式】

一行一个数，表示答案(对 998244353 取模)。

【样例输入】

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

【样例输出】

```
4
```

【数据范围与约定】

| 测试点编号 | n | k | 特殊性质 |
|-------|-------------------|-------------------|------|
| 1 | <=10 | =1 | 无 |
| 2 | | | |
| 3 | | <=5 | |
| 4 | | | |
| 5 | <=1000 | <=7 | |
| 6 | | | |
| 7 | <=10 ⁵ | <=8 | |
| 8 | | | |
| 9 | | | |
| 10 | | <=16 | |
| 11 | | | |
| 12 | | | |
| 13 | | | |
| 14 | | | |
| 15 | | <=10 ⁵ | |
| 16 | | | |
| 17 | | | |
| 18 | <=10 ⁶ | <=10 ⁶ | 是一条链 |
| 19 | | | 无 |
| 20 | | | |
| 21 | | | |
| 22 | | | |
| 23 | | | |
| 24 | | | |
| 25 | | | |

对于 100%的数据，保证 $1 \leq w_i \leq 10^4$ 。

【可能会用到的知识】

关于期望：

期望的定义：离散随机变量的一切可能值与其对应的概率 P 的乘积之和称为数学期望。

$$\text{即: } E(x) = \sum P(x = k) * val(k)$$

其中 $E(x)$ 是期望， $P(x = k)$ 是 $x=k$ 发生的概率。

提示：答案必定可以表示成 $\frac{p}{q}$ 的形式，在模意义下， $\frac{p}{q} = p * q^{-1}$ ，其中 q^{-1} 是 q 的逆元。

【提示】

读入数据较大，请使用快速的读入方式。

3. 辉夜的夜空明珠 (moon.cpp/c/pas)

【题目背景】

夜空珠里，藏着的是月球的公主。

连接幻想乡与月亮的回廊，在红月中隐灭。

“放逐了多少年，你还是月球的罪人。”

“是就是吧，永恒也是须臾，须臾也是永恒。”

【题目描述】

整个回廊可以看作一个 n 个点 m 条边的无向图，每条边走动花费的时间为 1。辉夜、永琳、铃仙、因幡帝等 k 个人或兔子可以通过传送阵分别进入这个图上的 k 个特殊的点，然后去寻找闯入者。但是在寻找闯入者之前，他们要聚集到一个点，以增强战斗力。注意，可以先到的人停下不走等后来的人。

闯入者不知道回廊的规则，因此被困住，对辉夜等 k 个人的行动没有影响。而辉夜等 k 个人必须按照回廊的规则走动。

回廊的规则如下：每个点有一个颜色，一共 4 种颜色，红、蓝、黄、绿，分别以 R、B、Y、G 表示。走动时必须在第 $4p+1$ 步到 $4p+4$ 步的时候走四种不同的颜色，当然最后一个不完整的周期内也不能走动相同颜色。注意，起点算第 1 步。

现在给定 k 个起点，辉夜想知道他们最短多长时间能够汇合，若不能汇合输出 -1。

【输入格式】

第一行一个 Case，表示测试点编号。

第二行两个数 n 和 m ，表示图有 n 个点 m 条边。

第三行一个 k ，表示有 k 个人。

第四行 k 个数，表示 k 个入口的编号。

第五行一个长度为 n 的字符串，仅包含 RBYG，表示 n 个点的颜色。

下面 m 行每行两个数， u_i, v_i ，表示第 i 条边连接了 u_i 和 v_i 。

【输出格式】

一个数表示最短汇合时间，不能汇合输出 -1。

【样例输入】

```
1
5 4
4
1 2 3 4
RRRRG
1 5
2 5
3 5
4 5
```

【样例输出】

```
1
```

【数据范围与约定】

对于 100%的数据，保证 $k \geq 2$ 。

| 测试点编号 | n | m | k |
|-------|--------------|---------------|-----------|
| 1 | ≤ 8 | ≤ 16 | ≤ 5 |
| 2 | ≤ 30000 | $= n-1$ | ≤ 5 |
| 3 | | | |
| 4 | ≤ 30000 | ≤ 150000 | ≤ 5 |
| 5 | | | |
| 6 | | | |
| 7 | ≤ 50000 | ≤ 200000 | ≤ 10 |
| 8 | | | |
| 9 | | | |
| 10 | | | |