

# 提高组 模拟

唐泽

（请选手务必仔细阅读本页内容）

## 1 题目概况

中文题目名称	妹子	旅程	老大
文件名	girls.pas/c/cpp	journey.pas/c/cpp	ob.pas/c/cpp
可执行文件名	girls	journey	ob
输入文件名	girls.in	journey.in	ob.in
输出文件名	girls.out	journey.out	ob.out
每个测试点时限	1s	1s	1s
测试点数目	10	10	10
每个测试点分值	10	10	10
附加样例文件	有	有	有
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	512M	512M	512M

## 2 注意事项

- 2.1 文件名（程序名和输入输出文件名）必须使用英文小写。
- 2.2 C/C++ 中函数 `main()` 的返回值类型必须是 `int`，程序正常结束时的返回值必须是 `0`。
- 2.3 造数据时采用的机器配置为：处理器 Intel® Core™ i5-6200U CPU @ 2.30GHz，内存 8G，评测时限不以此配置为准。
- 2.4 特别提醒：评测在 Linux 下进行，各语言的编译器版本以评测机为准，评测时不打开除 `O2` 外任何优化开关。C/C++ 应使用 `%lld` 进行 64 位整数的输入输出。
- 2.5 请妥善保存下发的样例，以免误删。
- 2.6 第二题不是我出的，现第一题是原第二题，现第二题是从普及组换过来的，出题人为裔家祺。

## 1. 妹子

### 1 问题描述

万人迷皮皮轩收到了很多妹子的礼物，由于皮皮轩觉得每个妹子都不错，所以将她们礼物的包装盒都好好保存，但长此以往皮皮轩的房间里都堆不下了，所以只能考虑将一些包装盒放进其他包装盒里节省空间。

方便起见，我们不考虑包装盒的高度和厚度，只考虑包装盒的长宽。

一句话题意：给出两个矩形，问是否可以将一个矩形放在另一个矩形的内部（含边界），多测。

### 2 输入格式

输入文件名为`girls.in`。

第一行，一个整数  $n$ ，表示数据组数。

对于下面的每一组数据：

第一行，四个整数  $a_1, b_1, a_2, b_2$  表示两个盒子的长宽。

### 3 输出格式

输出文件名为`girls.out`。

$n$  行，每行一个 `"Yes"` 或 `"No"`（不含引号），分别表示其中一个盒子可以放到另一个盒子中或两个盒子都不能放到另一个盒子中。

### 4 样例

见下发 `/girls/girls.in(out)`。

### 5 数据规模与约定

对于 100% 的数据， $n \leq 10$ ， $a_1, b_1, a_2, b_2 \leq 1000$ 。

## 2. 旅程

### 1 背景

您曾经带领着我，穿过我的白天的拥挤不堪的旅程，而到达了黄昏的孤寂之境。在通宵的寂静里，我等待着它的意义。

### 2 问题描述

神即将带领一些人去他们的孤寂之境，由于这个世界的的不稳定，地点之间的有向道路会不定期地毁坏，出于工作准备，神想知道在某些道路毁坏之后某两点之间的最短路。

就是给定一个有向图，现有两个操作，操作 1 是删除一条边（一条边可重复删除），操作 2 是询问两个点之间的最短路。

### 3 输入格式

输入文件名为 `journey.in`。

第 1 行两个正整数  $n, m$ ，分别表示图的点数和操作数。

第 2 行至第  $n + 1$  行每行  $n$  个正整数，为图的邻接矩阵，第  $i$  行第  $j$  列的数表示点  $i$  和点  $j$  间距离，保证对角线为 0。

接下来  $m$  行每行三个正整数  $c, x, y$ ， $c$  表示操作种类，为 1 或 2，当  $c = 1$  时表示删除  $x$  与  $y$  相连的边，当  $c = 2$  时表示询问  $x$  到  $y$  的最短路，若不可达则输出  $-1$ 。

### 4 输出格式

输出文件名为 `journey.out`。

输出若干行，每个 2 操作对应一行，答案为询问中  $x$  到  $y$  的最短路或  $-1$

### 5 样例

见下发 `/journey/journey.in(out)`。

### 6 数据规模与约定

对于 30% 的数据： $n, m \leq 10$

对于 50% 的数据： $n, m \leq 50$

对于 100% 的数据： $n \leq 200, m \leq 100000$ ，操作 1 不超过 200 次，边权不超过 10000

### 3. 老大

#### 1 问题描述

因为 OB 今年拿下 4 块金牌，学校赞助扩建劳模办公室为劳模办公室群，为了体现 OI 的特色，办公室群被设计成了树形 ( $n$  个点  $n - 1$  条边的无向连通图)，由于新建的办公室太大以至于要将奖杯要分放在两个不同的地方以便同学们丢硬币进去开光，OB 想请你帮帮他看看奖杯放在哪两个办公室使得在任意一个在劳模办公室做题的小朋友能最快地找到奖杯来开光。

一句话题意：给出一个  $n$  个点的树，在两个合适且不同的点放上奖杯，使得每个点到最近的奖杯距离最大值最小。

#### 2 输入格式

输入文件名为 `ob.in`。

第一行，一个整数  $n$ 。

接下来的  $n - 1$  行，每行两个数  $x\ y$ 。

#### 3 输出格式

输出文件名为 `ob.out`。

三个数，分别表示最小的最大距离，两个奖杯的位置。

#### 4 样例

见下发 `/ob/ob.in(out)`。

#### 5 数据规模与约定

对于前 60% 的数据， $n \leq 100$ 。

对于前 80% 的数据， $n \leq 2000$ 。

对于 80% 的数据，保证树的形态随机。

对于 100% 的数据，保证  $3 \leq n \leq 200000$ 。

由于读入文件过大，C/C++ 选手应使用读入优化以缩短读入时间，以下是一份模板：

---

```
template <typename T> void read(T &x) {
    x = 0; char c = getchar();
    for (; !isdigit(c); c = getchar());
    for (; isdigit(c); c = getchar()) x = x * 10 + c - '0';
}
```

---

其中 `t` 的类型可以是 `int/long long` 等数字类型，值得注意的是，这份模板仅支持非负整数的读入，你可以这样使用这份模板：

---

```
#include <bits/stdc++.h>
using namespace std;
template <typename T> void read(T &x) {
    x = 0; char c = getchar();
    for (; !isdigit(c); c = getchar());
    for (; isdigit(c); c = getchar()) x = x * 10 + c - '0';
}
int main() {
    int n;
    read(n);
    long long m;
    read(m);
    return 0;
}
```

---

一鼓作气，三道暴力。  
最后祝 AK 愉快！