

全国信息学奥林匹克联赛复赛模拟赛

提高组

时间：3 小时

满分：400 分

提交：自己姓名的文件夹（只包含 4 个源程序）

评测：JSOI 官方评测系统/Lazarus 下

限制：每题每个测试点的默认时限均为 1 秒、空限 64 兆

注意：输入输出格式请严格按照题目要求，否则不得分，输出的最后一行末尾请换行

1. 锻炼计划(exercise.pas)

身体是革命的本钱，OIers 不要因为紧张的学习和整天在电脑前而忽视了健康问题。小 x 设计了自己的锻炼计划，但他不知道这个计划是否可行，换句话说如果计划不当可能会让他的体力超支，所以小 x 请你帮助他。

一天有 1440 分钟，所以小 x 列出的是这一整天第 1 至第 1440 分钟的计划。小 x 的体力用一个整数来表示，他会按照计划表进行锻炼，同时，每分钟小 x 的体力会自动增加 1。如果某一分钟末小 x 的体力小于等于零，那么可怜的小 x 就累死了……

输入(exercise.in)

第一行是用空格分开的两个整数 n, m ，分别表示小 x 的初始体力值和计划的项目数量。

从第二行开始的 m 行，每行描述一个锻炼项目：名称、开始时间 a 、结束时间 b 、每分钟耗费的体力(用空格分隔)，表示此项目从第 a 分钟初开始，第 b 分钟末结束。锻炼项目按照开始时间递增顺序给出，不会出现两个项目时间冲突的情况。

输出(exercise.out)

输出包括两行，如果计划可行，第一行输出 "Accepted"，第二行输出这一天过后最后剩余的体力；否则在第一行输出 "Runtime Error"，第二行输出在第几分钟累死。

样例

Input	Output
10 1 Basketball 1 10 1	Accepted 1440
1 1 Nunchakus 1 1 2	Runtime Error 1

约定

$$0 < n \leq 2^{31} - 1$$

$$0 \leq m \leq 500$$

所有中间值的绝对值不会超过 $2^{31}-1$

每一个锻炼项目的名称不超过 20 个字符，其中不含空格

2. 魔兽争霸(war.pas)

小 x 正在销魂地玩魔兽

他正控制着死亡骑士和 n 个食尸鬼(编号 $1 \sim n$)去打猎

死亡骑士有个魔法，叫做“死亡缠绕”，可以给食尸鬼补充 HP

战斗过程中敌人会对食尸鬼实施攻击，食尸鬼的 HP 会减少

小 x 希望随时知道自己部队的情况，即 HP 值第 k 多的食尸鬼有多少 HP，以便决定如何施放魔法

请同学们帮助他:)

小 x 向你发出 3 种信号：（下划线在输入数据中表现为空格）

A_i_a 表示敌军向第 i 个食尸鬼发出了攻击，并使第 i 个食尸鬼损失了 a 点 HP，如果它的 $HP \leq 0$ ，那么这个食尸鬼就死了(Undead 也是要死的……)。

敌军不会攻击一个已死的食尸鬼。

C_i_a 表示死亡骑士向第 i 个食尸鬼放出了死亡缠绕，并使其增加了 a 点 HP。HP 值没有上限。

死亡骑士不会向一个已死的食尸鬼发出死亡缠绕

Q_k 表示小 x 向你发出询问

输入(war.in)

第一行，一个正整数 n

以后 n 个整数 表示 n 个食尸鬼的初始 HP 值

接着一个正整数 m

以下 m 行 每行一个小 x 发出的信号

输出(war.out)

对于小 x 的每个询问，输出 HP 第 k 多的食尸鬼有多少 HP，如果食尸鬼总数不足 k 个，输出-1。每个一行数。

最后一行输出一个数：战斗结束后剩余的食尸鬼数

样例

Input	Output
5	4
1 2 3	5
4 5	-1
10	11
Q 2	3

A 4 6	
C 1 4	
Q 2	
A 2 1	
A 3 3	
A 1 3	
Q 4	
C 2 10	
Q 1	

约定

40%的数据 $n \leq 3000$ $m \leq 5000$

70%的数据 $n \leq 8000$ $m \leq 10000$

100%的数据 $n \leq 30000$ $m \leq 50000$

90%的数据随机生成

食尸鬼 HP 没有上限

数据保证任意时刻食尸鬼的 HP 值在 `longint` 范围内

数据保证 A 和 C 命令中的食尸鬼是活着的

输入数据中没有多余空格、换行

3. 盟军敢死队(cmdo.pas)

2005 年是第二次世界大战胜利 60 周年，让我们的视线回到战火纷飞的年代……

苏德战争爆发后，苏、美、英 3 国曾多次商讨在西欧开辟第二战场，共同打击法西斯德国的问题。几经周折，1943 年 11 月，苏、美、英 3 国首脑在德黑兰会议上最终达成协议。1943 年圣诞节前夕，美国正式宣布任命艾森豪威尔将军为盟军最高司令，统一指挥盟军在西欧的登陆作战。1944 年初，盟军开始进行登陆作战的准备工作。

盟军开辟第二战场的意图是，在法国西北部登陆，夺取登陆场和港口，保障主力上陆和后勤供应，然后发动攻势占领整个法国西北部地区，并与在法国南部登陆的部队配合，向德国内地进攻，协同苏军最后战胜法西斯德国。盟军在比较了 3 处较为合适的登陆地点后认为，在诺曼底地区登陆条件最优越。

身经百战的敢死队员们这次接受的是一个看似简单但意义重大的任务：占领诺曼底最大的军火仓库。为了配合即将展开的登陆战，防止德军撤退时摧毁仓库设施，队员们要悄悄的取得仓库的控制权——消灭里面的所有敌人。

显然，敌众我寡，但是对于我们训练有素的队员们来说，这并不是一件困难的事情。

输入(cmdo.in)

仓库是一个 $m \times n$ 的矩形区域，每一格用一个字符来描述：“.”代表空地；“#”代表墙或障碍物；“^”，“v”(小写)，“<”，“>”四个字符分别表示正向 NSWE 四个方向看的敌人。敌人总是保持固定不动并朝着一个方向看，从这个方向一直

延伸直到边界或障碍物的区域是他的视线范围，如果一个敌人没有在任何人的视线范围之内，敢死队员就可以消灭他。你不能消灭一个正在另一个活着的敌人视线范围内的敌人，否则你就会被发现，后果不堪设想。一个敌人不会成为遮挡视线的障碍物。

输入数据的第一行是用空格分开的两个整数 n, m ，分别表示仓库的长和宽。接下来有 n 行，每行 m 个字符，是仓库的描述。

输出(cmdo.out)

如果能够成功消灭所有敌人，输出消灭所有敌人的不同顺序的数量，否则输出 “Impossible”（不含引号）。

样例

Input	Output
2 2 >^ #^	2
1 3 >.<	Impossible

约定

100%的数据中， $1 \leq m, n \leq 60$

90%的数据中，敌人数不超过 10

100%的数据中，敌人数不超过 15

4.暗黑破坏神(diablo.pas)

无聊中的小 x 玩起了 Diablo I...

游戏的主人公有 n 个魔法

每个魔法分为若干个等级，第 i 个魔法有 $p[i]$ 个等级(不包括 0)

每个魔法的每个等级都有一个效果值，一个 j 级的 i 种魔法的效果值为 $w[i][j]$

魔法升一级需要一本相应的魔法书

购买魔法书需要金币，第 i 个魔法的魔法书价格为 $c[i]$

而小 x 只有 m 个金币(好孩子不用修改器)

你的任务就是帮助小 x 决定如何购买魔法书才能使所有魔法的效果值之和最大

开始时所有魔法为 0 级 效果值为 0

输入(diablo.in)

第一行 用空格隔开的两个整数 $n m$

以下 n 行 描述 n 个魔法

第 $i+1$ 行描述 第 i 个魔法 格式如下

$c[i] \ p[i] \ w[i][1] \ w[i][2] \ \dots \ w[i][p[i]]$

输出(diablo.out)

第一行输出一个整数，即最大效果值。

以后 n 行输出你的方案：

第 $i+1$ 行有一个整数 $v[i]$ 表示你决定把第 i 个魔法学到 $v[i]$ 级

如果有多解 输出花费金币最少的一组

如果还多解 输出任意一组

样例

Input	Output
3 10	11
1 3 1 2 2	1
2 3 2 4 6	0
3 3 2 1 10	3

约定

$0 < n \leq 100$

$0 < m \leq 500$

$0 < p[i] \leq 50$

$0 < c[i] \leq 10$

保证输入数据和最终结果在 `longint` 范围内