

黄冈中学信息奥赛测试题

测试时间 8:00—12:00

(请仔细阅读本页面内容)

一. 题目概况

中文题目名称	DIY 手动	魔塔打怪兽	趣味运动会
英文题目与子目录名	diy	tower	sport
可执行文件名	diy	tower	sport
输入文件名	diy.in	tower.in	sport.in
输出文件名	diy.out	tower.out	sport.out
每个测试点时限	1s	2s	2s
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（过滤行末空格及文末换行）		
题目类型	传统	传统	传统
运行内存上限	256M	256M	256M

二. 提交源程序文件名

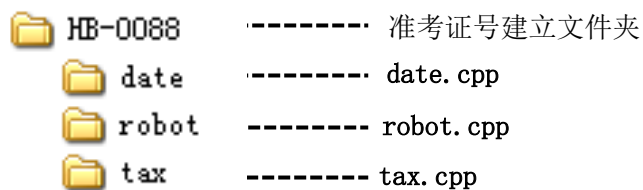
对于 C++ 文件	diy.cpp	tower.cpp	sport.cpp
对于 C 文件	diy.c	tower.c	sport .c

三. 编译命令 (-std=c++14 -O2)

对于 C++ 文件	g++.exe %s.cpp -o %s.exe -lm
对于 C 文件	gcc.exe %s.c -o %s.exe -lm

注意事项:

1. 文件名（程序名和输入输出文件名）必须使用英文小写。
2. C/C++ 中的函数 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。
3. 评测时允许使用万能头文件 #include <bits/stdc++.h>。
4. 提交的文件目录如下图所示，HB-0088 为考生准考证号，date、robot、tax 为题目规定的英文名称，目录中只包含源程序，不能包含其他任何文件。



DIY 手工(diy)

【题目描述】

H 小姐的生日很快就要到了,小 Z 还在思索应该送 H 小姐什么生日礼物才好。送化妆品太贵,送巧克力又太俗了……突然,小 Z 灵机一动,打算做一个 diy 手工以显诚意。

小 Z 找来了好多长度为 N 的冰棒棍,他打算把这些冰棒棍切割成若干段,然后拼接成一个个长方体,然后再用这些长方体垒成城堡。然而,小 Z 却开始为长方体长宽高的选取而烦恼:设长方体的长宽高分别为 L, W, H,为了更好地利用这些长度为 N 的冰棒棍,小 Z 希望把这些冰棒棍分为四组,其中第一组都能恰好切割成若干段长 (L 整除 N),第二组切割成若干段宽 (W 整除 N),第三组切割成若干段高 (H 整除 N),最后一组恰好切割成长宽高各一段 ($N=L+W+H$)。

现在,小 Z 好奇,拼成的长方体,体积最大是多少?

【输入格式】

从文件 diy.in 中读入数据。

第一行为测试数据组数 T,表示接下来有 T 组数据。

每组测试数据第一行为一个正整数 N,表示冰棒棍的长度。

【输出格式】

输出到文件 diy.out 中。

每组测试数据输出一行,包含一个整数,表示长方体的最大体积。若不存在合法方案,输出-1。

【样例输入】

```
3
1
2
3
```

【样例输出】

```
-1
-1
1
```

【数据范围与约定】

对于全部测试数据,满足 $T \leq 10^6$ 。

对于 30%数据,满足 $N \leq 10^2$ 。

对于 60%数据,满足 $N \leq 10^4$ 。

对于 100%数据,满足 $N \leq 10^6$ 。

魔塔打怪兽 (tower)

【题目描述】

小 Z 暑假无聊, 开始回顾起他小时候玩过的游戏。其中魔塔曾经是小 Z 的最爱, 但现在, 由于小 Z 太久没玩了, 技艺已经生疏, 他忘了最佳打怪顺序, 导致经常跑到高层打不过大怪物, 又得回到低层打小怪刷属性值, 玩得好不痛快。

于是乎, 小 Z 重新思考最佳打怪顺序的问题。已知勇士有武力值、防御值、魔法攻击、魔法防御……每个怪物也有同样多的属性, 为了保证不浪费生命值, 小 Z 总是在确保勇士的武力值和防御值分别大于等于怪物的防御值和武力值, 魔法攻击和魔法防御也分别大于等于怪物的魔法防御和魔法攻击时, 才会选择这只怪物刷属性值。

把问题再抽象一下就是, 勇士有 K 个属性, 大小分别为 $v[1], v[2], \dots, v[K]$, 一共有 N 只怪物, 每只怪物也有相应的 K 个属性, 第 i 只怪物的第 j 项属性标记为 $a[i][j]$ 。若对于任意的 j ($1 \leq j \leq K$) 都有 $a[i][j] \leq v[j]$, 则勇士可以干掉第 i 只怪物, 而且干掉第 i 只怪物后, 勇士的各项属性都会得到提升, 其中第 j 项属性的提升了 $b[i][j]$ 。

现在小 Z 好奇, 按照最优策略来打怪, 最多能干掉多少只怪物。

【输入格式】

从文件 tower.in 中读入数据。

第一行为测试数据组数 T ($1 \leq T \leq 10$)。

每组测试数据的第一行为怪物数 N 及属性数 K 。

第二行包含 K 个非负整数分别表示 $v[1], v[2], \dots, v[K]$ 。

接下来 N 行, 第 i 行包含 $2 \times K$ 个非负整数, 表示 $a[i][1], a[i][2], \dots, a[i][K]$ 和 $b[i][1], b[i][2], \dots, b[i][K]$ 。

【输出格式】

输出到文件 tower.out 中。

对于每组测试数据, 输出第一行包含一个整数 M , 表示小 Z 最多能干掉多少只怪物。第二行包含 K 个整数, 表示干掉 M 只怪物后, 勇士的 K 项属性分别是多少。

【样例输入】

```
1
4 3
7 1 1
5 5 2 6 3 1
24 1 1 1 2 1
0 4 1 5 1 1
6 0 1 5 3 1
```

【样例输出】

```
3
23 8 4
```

【数据范围与约定】

对于全部测试数据，满足 $T \leq 10$ ，且所有属性值不超过10000。

测试点	N	K
1	≤ 100	≤ 5
2		
3		
4	$\leq 10^3$	≤ 1
5		
6	$\leq 10^5$	≤ 1
7		
8	$\leq 10^5$	≤ 5
9		
10		

趣味运动会(sport)

【题目描述】

一年一度的趣味运动会马上就要开始啦!作为班长的小 Z 最近正忙于制定策略,决定派哪些同学参加哪些趣味项目。其中最棘手的莫过于二人三足,顾名思义,这个运动需要每组两位同学默契配合,才能走得尽可能的快。

已知全班共有 N 位同学报名参加趣味运动会,小 Z 需要在这 N 位同学中选出若干对同学,组队参加二人三足。可惜的是,这 N 位同学之间总是小摩擦不断,说不准昨天 A 和 B 吵架了,不再适合组队,而没过多久,前天吵架的 C 和 D 就又突然和好了。小 Z 得知这 N 位同学的吵架及和好事件的发生,他好奇每发生一次吵架或和好后,派出 k 组同学参加二人三足的方案数,分别是多少。

其中 $k=1, 2, \dots, N/2$ 。

【输入格式】

从文件 sport.in 中读入数据。

第一行为测试数据组数 T ($1 \leq T \leq 10$)。

每组测试数据的第一行为学生数量 N 及事件数 M , 其中 N 为偶数。

接下来 M 行, 第 i 行通过一个三元组 cuv 描述一个事件, 其中 c 为字符, 要么是“+”要么是“-”, 而 u 和 v 为吵架(用“-”表示)或和好(用“+”表示)的两位同学的编号。

数据保证两位刚和好的同学之前处于吵架状态, 而刚吵架的同学之前处于和好状态。

【输出格式】

输出到文件 sport.out 中。

每组数据输出 M 行, 每行包含 $N/2$ 个数字, 第 k 个数字表示派出 k 组同学参赛的方案数对 10^9+7 取模后是多少。

【样例输入】

```
1
4 8
+ 1 2
+ 3 4
+ 1 3
+ 2 4
- 1 2
- 3 4
+ 1 2
+ 3 4
```

【样例输出】

```
1 0
2 1
```

3 1
4 2
3 1
2 1
3 1
4 2

【数据范围与约定】

对于全部测试数据，满足 $T \leq 10$ 。

测试点	N	M
1	≤ 6	≤ 20
2		
3		
4	≤ 6	≤ 30000
5		
6	≤ 10	≤ 100
7		
8	≤ 10	≤ 30000
9		
10		