

华师一 NOIP2020 摸底试题

第一试

2020 年 11 月 14 日 8:00~12:30

(请选手务必仔细阅读此页内容)

一. 题目概况

中文题目名称	求 x 的值	重排序列	GGO 的手游	GKK 的游戏
英文题目名称	a	typeset	ggo	gkk
可执行文件名	a	typeset	ggo	gkk
输入文件名	a.in	typeset.in	ggo.in	gkk.in
输出文件名	a.out	typeset.out	ggo.out	gkk.out
每个测试点时限	1 秒	2 秒	1 秒	1 秒
内存上限	512MB	256MB	256MB	256MB
测试点数目	10	10	20	25
每个测试点分值	10	10	5	4
结果比较方式	全文比较	全文比较	全文比较	全文比较
题目类型	传统	传统	传统	传统

二. 提交源程序文件名

对于 C 语言	a.c	typeset.c	ggo.c	gkk.c
对于 C++ 语言	a.cpp	typeset.cpp	ggo.cpp	gkk.cpp

三. 编译命令

对于 C 语言	gcc -o a a.c -lm	gcc -o typeset typeset.c -lm	gcc -o ggo ggo.c -lm	gcc -o gkk ggo.c -lm
对于 C++ 语言	g++ -o a a.cpp -lm	g++ -o typeset typeset.cpp -lm	g++ -o ggo ggo.cpp -lm	g++ -o gkk gkk.cpp -lm

四. 注意事项:

1. 文件名(程序名和输入输出文件名)必须使用小写。
2. 选手提交以自己编号命名的文件夹, 文件夹内包含 4 个源文件(.c, .cpp, .pas, .), 并在文件夹下建立 4 个相应的子目录, 并将 4 个对应的源程序分别放入对应的子文件夹中, 所有名字必须使用小写; 例如:

```
A001          编号
|---a
|      |---a. cpp
|---typeset
|      |---typeset. cpp
|---ggo
|      |---ggo.cpp
|---gkk
|      |---gkk.cpp
|---a. cpp
|---typeset. cpp
|---ggo.cpp
|---gkk.cpp
```

3. 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`，`Maxlongint=2147483647`。
4. 题目简单，请认真对待，争取三位数。
5. 每道题源代码长度限制均为 `50KB`。
6. 每道题的数据都有一定梯度。请尽量优化算法，争取拿高分。
7. 评测在 `linux` 系统下全国评测机和 `windows` 下的 `lemon` 分别评测，所以放在外面的代码是在 `lemon` 下评测用，放在各自子目录下的是在国评系统下评测用。
8. 评测时的栈空间为内在空间，并且不开启 `-O2` 优化开关。
9. 建议最后 `10` 分钟不要再编程，检查一下提交的文件夹中的代码是否符合要求，检查文件名，输入输出文件名，数据类型，数据精度，空间限制，赋初值等是否按试卷上的要求来做的，一定要杜绝一切的不小心的人为错误，显然这种错误是致命的。
10. 做题时，审题是关键，必须深入与全面，学过的知识与做过的题都是分析问题的有利武器；编写代码要细致，多写函数，便于调试，只有这样，才能达到你的期望。

伢子妹子们，杜绝非知识类错误！！！！

伢子妹子们，必须杜绝答案错误！！！！

它与每题得分绝对不是一名空话！！！！

1 a

1.1 Description

定义函数 $f(x)$ 为组成 x 的各位数字之和，如 $f(2018) = 2 + 0 + 1 + 8 = 11$ 。
给定一个正整数 n 求所有满足条件 $f(x) + x = n$ 的 x 。

1.2 Input

共一行，一个正整数 n 。

1.3 Output

第一行一个整数 ans 表示有多少个整数 x 满足条件。
下接 ans 行，每行一个整数表示满足条件的 x 。

1.4 Samples

见选手目录下的 `a/a1.in-a3.in` 与 `a/a1.out-a3.out`。

1.5 Constraints

对于 30% 的数据满足： $n \leq 1000$

对于 100% 的数据满足： $n \leq 10^{18}$

注意：输出结果时 x 按从小到大的顺序输出。

2. Typeset

Typeset

Description

给你一个长度为 n 的序列，要求你重新排列，求是否存在一种排列方案满足任意两个相邻的数的积都是 2^m 的倍数。如果存在则输出 *Yes*，否则输出 *No*，有多组数据。

Input Format

第一行一个数 T 表示数据组数。

对于每组数据，第一行两个数 n, m ，第二行 n 个数 a_1, a_2, \dots, a_n 表示序列。

Output Format

输出共 T 行，对于每组数据，输出 *Yes* 或 *No*。

Sample 1

Input:

```
4
3 2
1 10 100
4 2
1 2 3 4
3 3
1 4 1
6 2
2 7 1 8 2 8
```

Output:

```
Yes
No
No
Yes
```

Sample2

Input:

```
5
10 2
136 68 26 58 124 136 26 31 34 62
10 3
152 22 58 17 92 17 5 29 28 58
10 1
17 232 3 152 116 38 6 14 6 19
10 2
24 88 58 6 13 68 38 13 6 5
10 3
13 5 23 7 24 152 76 24 184 31
```

Output:

```
Yes
No
Yes
Yes
No
```

Constraints

对于所有数据 $T \leq 5, 1 \leq a_i \leq 10^9$ 。

对于前 30% 的数据, 满足 $1 \leq n \leq 10, 1 \leq m \leq 5$ 。

对于 100% 的数据, 满足 $1 \leq n \leq 10^5, 1 \leq m \leq 20$ 。

3. GGO 手游

【问题描述】

酒足饭饱之后（没有后三个字），紫萱学姐打开了手机上的一款游戏，叫做 GwentGrandOrder，简称 GGO，但是由于光庭总菊的要求，手机游戏中不得出现英文，所以就有一个很 low 的中文名：昆特-冠位指定。

看到学姐玩这个游戏，学长也产生了浓厚的兴趣并开始了围观，学姐决定给学长展示一下自己的牌技。

这是一款卡牌游戏，每个人用自己的卡组进行对战，游戏开始时每个人从牌堆中抽取若干张牌，然后依次打出手牌，为了简化这个游戏，我们假定牌堆中只有以下三种牌：

1. 单位牌，分为近战/远程/攻城/敏捷单位四种牌，每张牌都有自己的力量值，前三种单位牌使用时将该牌置于己方战场中的对应排，敏捷单位可以置于近战或远程排，**放置后不可移动且受该排的特殊牌影响。**

2. 特殊牌，分为霜霰/浓雾/地形雨/史凯利杰风暴/天晴五种天气牌和领导号角，前四种天气牌的效果分别为将双方的所有**近战/远程/攻城/远程和攻城单位**力量降为 1，天晴的效果为驱散当前所有天气牌效果。领导号角可以在**最终计算力量值**时将本方其中一排所有单位牌的力量翻倍。

3. 英雄牌，与单位牌使用方法相同，但是**英雄牌的力量值不受任何特殊卡牌影响**而增加或减少。

单位牌和英雄牌可能带有“间谍”属性，带有该属性的单位牌或英雄牌在使用时将会被置于对方战场，同时使用者将会从牌堆中抽取两张牌。

双方出牌结束或无牌可出时，计算双方场上所有牌的力量值之和，力量值大的一方获胜。

注意：如果一张单位牌同时受到负面天气和领导号角的影响，那么这张牌的力量值应该为 2。同时一张力量值为 0 的单位牌受到负面天气影响时力量值不会变为 1。

假设由于某种原因，对手已经打出 m 张单位牌（以总牌数和力量之和的形式表示，**敌方敏捷单位以近战或远程方式直接使用，即描述敌方单位或英雄牌时不会出现敏捷属性**）和特殊牌并结束出牌。紫萱学姐的卡组中有 n 张牌，而且可以从中抽取 k 张作为初始手牌，由于学长的欧气加持，紫萱学姐**每次抽牌时都可以抽到自己想抽的那一张牌**。但是这是一个氪金看脸游戏，每张牌都有一定的稀有度，作为一个在 NOI 上用尽人品从亚变非的新晋酋长，紫萱学姐希望能用尽量不稀有的卡牌战胜对手，来证明这是一个技术游戏。请你帮助紫萱学姐设计一个方案使得所使用的牌中稀有值的最大值尽可能小。由于她所使用的牌组所属阵营尼弗迦德的特性，在双方力量值相等时判定学姐胜利。

【输入】

第一行三个正整数 n, m 和 k ，意义如题目所示。

接下来 m 行每行描述一张牌，代表对方所出的所有牌。

接下来 n 行每行描述一张牌，代表紫萱学姐的牌堆。

每张牌用若干个整数表示，第一个整数代表该牌的稀有度 v_i ，第二个整数代表该牌类型（1 为单位牌，2 为特殊牌，3 为英雄牌）。

若该牌为单位牌或英雄牌，接下来三个自然数代表该牌的种类（1~4 分别为近战/远程/攻城/敏捷）和力量值，以及该牌是否为间谍牌（0 为正常单位或英雄，1 为间谍牌）。

若该牌为特殊牌，接下来一个整数代表该牌的类型，0~5 分别为领导号角/霜霰/浓雾/地形雨/史凯利杰风暴/天晴。若该牌为领导号角且是对方所出的牌，接下来一个正整数代表此牌所作用的位置，1~3 分别对应近战/远程/攻城。

【输出】

一个正整数，为最优解中所使用的卡牌稀有值最大值，如果无论如何学姐也无法赢得这场游戏，输出“SingleDog”（不含引号）。

【输入输出样例】

Input	Output
5 5 1 1 1 1 5 0 1 1 1 5 0 1 1 3 10 0 1 3 1 5 1 1 2 0 1 7 3 1 15 0 8 1 1 5 1 9 2 1 10 1 2 10 0 11 2 0	9

【样例解释】

样例中游戏开始时，敌方战场中有两个力量为 5 的近战单位和一个力量为 10 的攻城单位，且敌方近战排存在领导号角。我方战场有一个力量为 5 的近战间谍英雄。

我方手牌为力量为 15 的近战英雄杰日天、力量为 5 的近战间谍单位、霜霰、力量为 10 的远程单位、领导号角。

开始时抽取间谍牌（稀有度为 8），并将其置于对方战场，抽取英雄牌（稀有度为 7）和霜霰（稀有度为 9）打出并结束回合。

由于英雄不受特殊牌影响，我方力量总和为 $15+5=20$ 。

由于霜霰和领导号角的双重影响，敌方力量总和为 $2+2+2+10=16$

我方胜利，所用牌稀有度最大值为 9，为最优解。

【数据范围及约束】

对于测试点 1/2， $n, m \leq 10$ 。

对于测试点 3/4， $n, m \leq 1000$ 。

对于测试点 5/6， $n, m \leq 100000$ 。

对于测试点 1/3/5，不含间谍牌。

对于测试点 1/2/3/4/5/6，不含特殊牌

对于测试点 7/8， $n, m \leq 10000$ 。

对于测试点 9/10， $n, m \leq 100000$ 。

对于所有测试点， $v_i \leq 10^9$ ，所有单位的力量值 ≤ 10000 。

关于 GGO 手游的理解细节:

1. 我方放出的「间谍」在结算时算对面的力量。而对面放出的「间谍」算我方的。
2. 用了天晴之后还可以继续使用「负面天气」。
3. 对面的出牌视为按顺序出，也就是说对面如果用天晴那他前面用的「负面天气」就都没用了。
4. 「号角」不能叠加，每一排只能用一個。
5. 我方的「号角」可以作用在对方派过来的「间谍」上。
6. 使用天气之后，除非使用「天晴」，在对应的行「出牌」都会受到天气影响。

4 GKK

Description

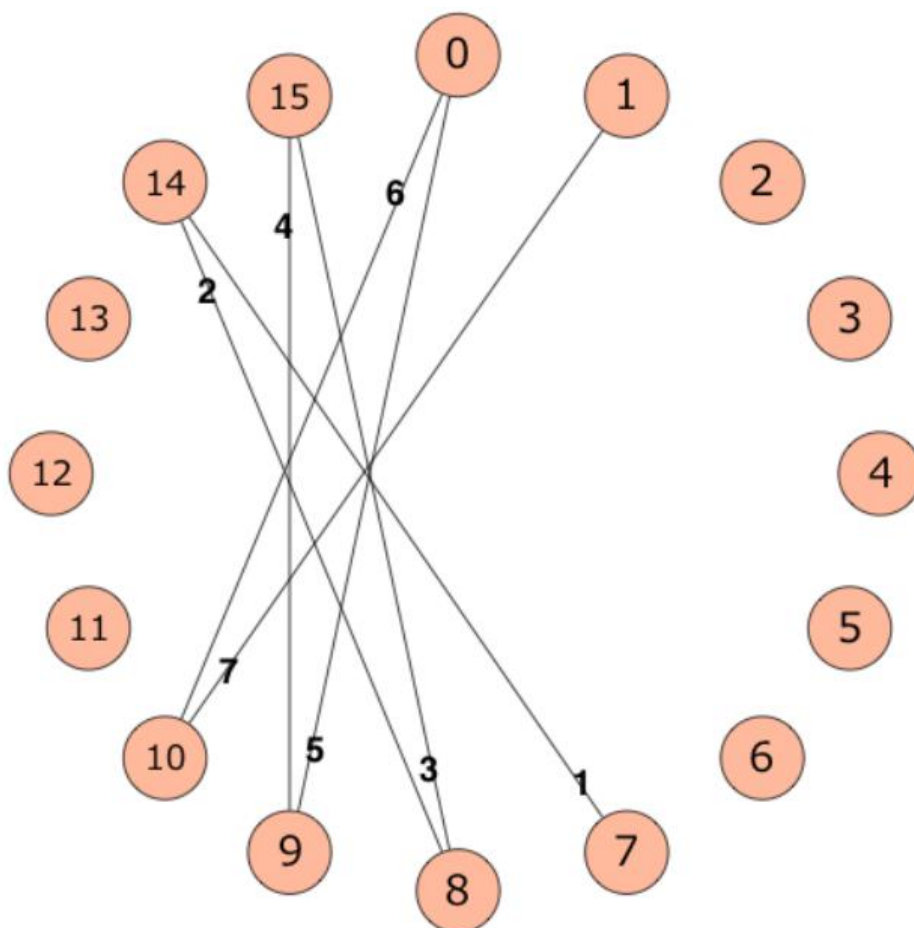
GKK是一个喜欢环上游戏的男孩。

现在有一张 n 个点的组成的图，每个点的编号为 0 到 $n-1$ 。你有 m 次操作，每一个操作有三个参数 a, b, c 。操作的意义如下：

- 在编号为 $a+0, b+0$ 的点之间连一条权值为 $c+0$ 的边。
- 在编号为 $b+0, a+1$ 的点之间连一条权值为 $c+1$ 的边。
- 在编号为 $a+1, b+1$ 的点之间连一条权值为 $c+2$ 的边。
- 在编号为 $b+1, a+2$ 的点之间连一条权值为 $c+3$ 的边。
- 在编号为 $a+2, b+2$ 的点之间连一条权值为 $c+4$ 的边。
- 在编号为 $b+2, a+3$ 的点之间连一条权值为 $c+5$ 的边。
- ...

其中，点的编号都是模 n 意义下的，即 n 号点与 0 号点等价， $2n-1$ 号点与 $n-1$ 号点的等价。

为了方便理解，下图为当 $n=16$, $a_1=7$, $b_1=14$, $c_1=1$ 首先加入的 7 条边。



GKK想知道，在所有操作进行完毕之后，求出原图的最小生成树的各边权值之和。他把这个问题抛给了你。

最小生成树的定义：在原图中选出 $n - 1$ 条边，使得图联通并且所选的边的权值之和最小。

Input Format

第一行两个整数 n, m 表示点数和边数。

下接 m 行，每行三个数 u, v, w 表示一次操作。

Output Format

一行一个整数 ans 表示答案。

Sample 1

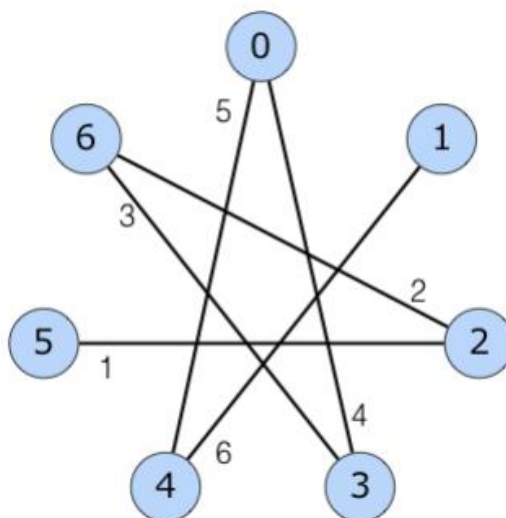
Input:

```
7 1
5 2 1
```

Output:

```
21
```

Explanation: 下图是该图的最小生成树， $1 + 2 + 3 + 4 + 5 + 6 = 21$ 。



Sample2

Input:

```
2 1
0 0 1000000000
```

Output:

```
1000000001
```

Hint: 注意可以存在自环和重边。

Sample3

Input:

```
5 3
0 1 10
0 2 10
0 4 10
```

Output:

```
42
```

Constraints

Subtask1 满足 $n \times m \leq 4 \times 10^5$, 30pts

Subtask2 满足所有边权相等, 10pts。

Subtask3 满足边权可以构成一个等差数列, 公差为 2, 20pts。

Subtask4 满足 $2 \leq n \leq 2 \times 10^5$, $1 \leq m \leq 2 \times 10^5$, 40pts。