

《程序竞赛基础》期末大作业

一、作业题目

题目 1：CCF 相关 - 题号 T120（占三题总分 40%）

题目链接：<http://118.190.20.162/view.page?gpid=T120>

试题编号：	202012-4
试题名称：	食材运输
时间限制：	1.0s
内存限制：	512.0MB
	<p>题目背景</p> <p>在T市有很多个酒店，这些酒店对于不同种类的食材有不同的需求情况，莱莱公司负责每天给这些酒店运输食材。</p> <p>由于酒店众多，如何规划运输路线成为了一个非常重要的问题。你作为莱莱公司的顾问，请帮他们解决这个棘手的问题。</p> <p>题目描述</p> <p>T市有 N 个酒店，这些酒店由 $N - 1$ 条双向道路连接，所有酒店和道路构成一颗树。不同的道路可能有不同的长度，运输车通过该道路所需要的时间受道路的长度影响。</p> <p>在T市，一共有 K 种主流食材。莱莱公司有 K 辆车，每辆车负责一种食材的配送，不存在多辆车配送相同的食材。</p> <p>由于不同酒店的特点不同，因此不同酒店对食材的需求情况也不同，比如可能 1 号酒店只需要第 1, 5 种食材，2 号酒店需要全部的 K 种食材。</p> <p>莱莱公司每天给这些公司运输食材。对于运输第 i 种食材的车辆，这辆车可以从任意酒店出发，然后将食材运输到所有需要第 i 种食材的酒店。假设运输过程中食材的装卸不花时间，运输车足够大使得其能够在出发时就装满全部所需食材，并且食材的重量不影响运输车的速度。</p> <p>为了提高配送效率，这 K 辆车可以从不同的酒店出发。但是由于T市对于食品安全特别重视，因此每辆车在配送之前需要进行食品安全检查。鉴于进行食品安全检查的人手不足，最多可以设置 M 个检查点。</p> <p>现在莱莱公司需要你制定一个运输方案：选定不超过 m 个酒店设立食品安全检查点，确定每辆运输车从哪个检查点出发，规划每辆运输车的路线。</p> <p>假设所有的食材运输车在进行了食品安全检查之后同时出发，请制定一个运输方案，使得所有酒店的等待时间的最大值最小。酒店的等待时间从运输车辆出发时开始计算，到该酒店所有需要的食材都运输完毕截至。如果一个酒店不需要任何食材，那么它的等待时间为 0。</p>

输入格式

从标准输入读入数据。

输入的第一行包含 3 个正整数 N, M, K ($1 \leq N \leq 10^2, 1 \leq M \leq K \leq 10$)，含义见题目描述。

接下来 N 行，每行包含 K 个整数。每行输入描述对应酒店对每种食材的需求情况，1 表示需要对应的食材，0 表示不需要。

接下来 $N - 1$ 行，每行包含 3 个整数 u, v, w ，表示存在一条通行时间为 w 的双向道路连接 u 号酒店和 v 号酒店。保证输入数据是一颗树，酒店从 1 编号到 N ，保证 $1 \leq u, v \leq N$ 并且 $1 \leq w \leq 10^6$ 。

输出格式

输出到标准输出。

输出一个整数，表示在你的方案中，所有酒店的等待时间的最大值。

样例1输入

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

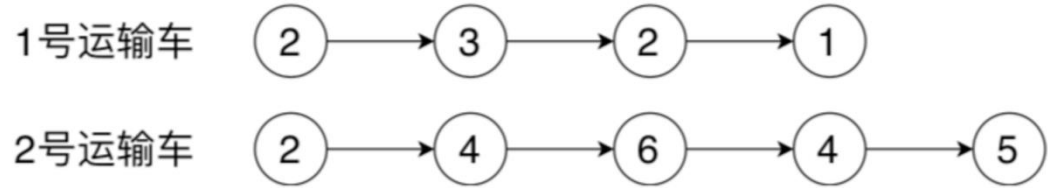
样例1输出

```
15
```

样例1解释

● 需要对应的食材
⊖ 不需要对应的食材

样例1的输入数据如上图。由于限制了最多只能设置 1 个检查点，因此可以设置两辆运输车的路径如下：



在 2 号酒店设置检查点，最晚拿到所有食材的酒店为 3 号酒店，等待时间为 9。

样例2输入

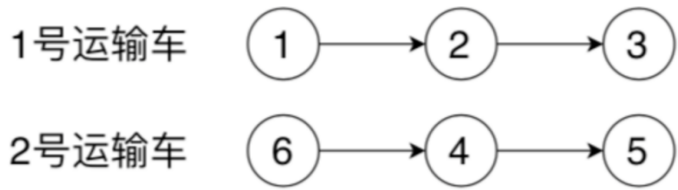
```
6 2 2
1 0
0 0
1 0
0 1
0 1
0 1
1 2 7
2 3 2
2 4 4
4 5 5
4 6 3
```

样例2输出

9

样例2解释

样例2的输入数据和样例1几乎完全相同，唯一的区别在于样例2中允许最多设置 2 个检查点。我们可以设置两辆运输车的路径如下：



在 1 号酒店和 6 号酒店设置检查点，最晚拿到所有食材的酒店为 5 号酒店，等待时间为 15 。

子任务

本题目数据规模如下：

分数占比	N	M	K	特殊性
30%	$\leq 10^2$	$= K$	≤ 10	保证输入数据是一条链，且 $u + 1 = v$
40%				无
30%		$\leq K$		

题目 2：CCF 相关 - 题号 T162（占三题总分 40%）

题目链接：<http://118.190.20.162/view.page?gpid=T162>

试题编号：	202303-4
试题名称：	星际网络II
时间限制：	2.0s
内存限制：	1.0GB
<p>问题描述</p> <p>随着星际网络的进一步建设和规模的增大，一个新的问题出现在网络工程师面前——地址空间不够用了！原来，星际网络采用了传统的IPv6协议，虽然有 2^{128} 级别的可用地址数量，但面对广袤无垠的宇宙和爆炸式增长的网络用户数，如此庞大的地址空间也面临了用尽的那一天。</p> <p>新的通信协议的研发工作交给了著名的网络科技圣地——西西艾弗星。最终，经过2333年的不懈努力，西西艾弗星的工程师们设计出了一种新的协议——“西西艾弗IP协议”，又称IPxxaf。</p> <p>在IPxxaf协议中，一个地址由 n 位二进制位组成，其中 n 是 16 的倍数。日常表示一个地址时，采用类似IPv6协议的十六进制表示法，每 4 位用 <code>:</code> 隔开。如 $n = 32$ 时，地址为 <code>2a00:0001</code>，即表示一个二进制为 <code>0010 1010 0000 0000 0000 0000 0000 0001</code> 的地址。注意不会出现IPv6中省略每组的前导 0 或用 <code>::</code> 省略一段 0 的情况。</p> <p>为方便起见，记 $num(s)$ 为地址 s 按高位在前、低位在后组成的 n 位二进制数，称一段“连续的地址”为 $num(s)$ 成一段连续区间的一系列地址。</p> <p>西西艾弗星的网络管理员负责地址的分配与管理。最开始，整个地址空间都是未分配的。用户可以随时向管理员申请一些地址：</p> <p>1 id l r：表示用户 id 申请地址在 $l \sim r$ 范围内（包含 l 和 r，下同）的一段连续地址块。</p> <p>在地址申请操作中，管理员需要先检查地址是否可用。如果用户申请的地址全部未被分配，则检查通过；若地址中存在已经分配给其他用户的地址，则检查失败。</p> <p>但有一种特殊情况：申请的地址中没有已经分配给其他用户的地址，但含有一些先前已分配给该用户本人的地址。此时可以认为检查通过，但若申请的地址先前已全部分配给该用户则检查失败。</p> <p>如果上述检查通过，则管理员向用户返回 YES，并将申请的地址分配给该用户；若不通过，则向用户返回 NO，同时不改变现有的地址分配。</p> <p>网络管理员要定期检查地址的分配情况，具体而言有如下两种操作：</p> <p>2 s：检查地址 s 被分配给了哪个用户。若未被分配，则结果为 0。</p> <p>3 l r：检查 $l \sim r$ 范围内的所有地址是否完整地分配给了某个用户。若是，回答该用户的编号；若否，回答 0。</p> <p>在整个网络的运行过程中，共出现了 q 次申请地址和检查地址分配的操作。作为西西艾弗星的一名重要的网络技术顾问，你要帮网络管理员依次处理每个操作，并回答相应的结果。</p>	

问题描述:

输入格式

从标准输入读入数据。

第一行，2 个正整数 n, q 。

接下来 q 行，每行一个操作，格式如上所述，其中的 id 为正整数， l, r, s 均为IPxxaf地址串，其中十六进制均用数字和小写字母表示。

输出格式

输出到标准输出。

输出 q 行，每行一个非负整数或字符串，表示此次操作的结果。

其中，对于操作 1，输出 YES 或 NO；对于操作 2,3，输出一个非负整数。

样例输入1

```
32 12
1 1 0001:8000 0001:ffff
2 0001:a000
3 0001:c000 0001:ffff
1 2 0000:0000 000f:ffff
2 0000:1000
1 1 0001:8000 0001:8fff
1 2 0000:0000 0000:ffff
2 0000:1000
1 1 0002:8000 0002:ffff
3 0001:8000 0002:ffff
1 1 0001:c000 0003:ffff
3 0001:8000 0002:ffff
```

样例输出1

```
YES
1
1
NO
0
NO
YES
2
YES
0
YES
1
```

样例解释

第 4 个操作时，由于用户 2 申请的部分地址已被分配给用户 1，因此申请不通过；
第 6 个操作时，由于用户 1 申请的全部地址已被分配给用户 1，因此申请不通过；
第 11 个操作时，用户 1 申请的部分地址已被分配给用户 1，其余地址尚未被分配，申请通过；

数据范围

对于所有数据， $1 \leq n \leq 10^4$ ， $1 \leq q \leq 10^4$ ， $1 \leq a_i \leq 10^9$ ， $1 \leq b_i \leq 10^9$ ， $1 \leq c_i \leq 10^9$ ， $1 \leq d_i \leq 10^9$ ， $1 \leq e_i \leq 10^9$ ， $1 \leq f_i \leq 10^9$ ， $1 \leq g_i \leq 10^9$ ， $1 \leq h_i \leq 10^9$ ， $1 \leq i \leq n$ ，对于操作 1，保证 $1 \leq a_i \leq 10^9$ 。

测试点编号	$n \leq$	$q \leq$	特殊性质
1 ~ 4	16	200	无
5 ~ 6	64	200	无
7 ~ 9	512	200	无
10 ~ 11	16	20000	无
12 ~ 13	64	50000	无
14 ~ 16	512	50000	所有操作 1 的 id 互不相同
17 ~ 20	512	50000	无

题目 3: ACM 相关 - Codeforces 1786F (占三题总分 20%)

题目链接: <https://codeforces.com/contest/1786/problem/F>

F. Wooden Spoon

time limit per test: 4 seconds
memory limit per test: 512 megabytes
input: standard input
output: standard output

2^n people, numbered with distinct integers from 1 to 2^n , are playing in a single elimination tournament. The bracket of the tournament is a full binary tree of height n with 2^n leaves.

When two players meet each other in a match, a player with the **smaller** number always wins. The winner of the tournament is the player who wins all n their matches.

A virtual consolation prize "Wooden Spoon" is awarded to a player who satisfies the following n conditions:

- they lost their first match;
- the player who beat them lost their second match;
- the player who beat that player lost their third match;
- ...;
- the player who beat the player from the previous condition lost the final match of the tournament.

It can be shown that there is always exactly one player who satisfies these conditions.

Consider all possible $(2^n)!$ arrangements of players into the tournament bracket. For each player, find the number of these arrangements in which they will be awarded the "Wooden Spoon", and print these numbers modulo 998 244 353.

Input

The only line contains a single integer n ($1 \leq n \leq 20$) — the size of the tournament.

There are 20 tests in the problem: in the first test, $n = 1$; in the second test, $n = 2$; ...; in the 20-th test, $n = 20$.

Output

Print 2^n integers — the number of arrangements in which the "Wooden Spoon" is awarded to players $1, 2, \dots, 2^n$, modulo 998 244 353.

Examples

input	Copy
1	
output	Copy
0 2	

input	Copy
2	
output	Copy
0 0 8 16	

input	Copy
3	
output	Copy
0 0 0 1536 4224 7680 11520 15360	

Note

In the first example, the "Wooden Spoon" is always awarded to player 2.

In the second example, there are 8 arrangements where players 1 and 4 meet each other in the first match, and in these cases, the "Wooden Spoon" is awarded to player 3. In the remaining 16 arrangements, the "Wooden Spoon" is awarded to player 4.

二、评分细则

1.网页测评给分（占总分的 60%）：

100 分：代码通过全部测试用例。

80 分：代码未通过全部测试用例，但通过了大部分测试用例。

60 分：代码通过少部分测试用例。

40 分：代码未通过任何测试用例，但给出了合理的思路。

0 分：代码未通过任何测试用例，且没有给出合理的思路。

2.报告详细程度（占总分的 20%）：

100 分：报告内容详尽、结构清晰，涵盖问题描述、解题思路、算法分析、代码实现，上传网页测评分数截图等关键内容。

80 分：报告内容较为详尽，但在某些关键部分有所遗漏或表述不清。

60 分：报告内容基本涵盖问题描述、解题思路和算法分析，但缺乏详细的代码实现说明。

40 分：报告内容较为简略，只涉及问题描述和解题思路，缺乏算法分析和代码实现以及网页测评分数截图。

0 分：未提交报告或报告内容极为不完整。

3.代码与网络上代码及其他同学代码的重合程度（占总分的 20%）：

与网络上的代码和其他同学的代码进行对比。如果代码相似度较高，根据重合程度的具体情况给出扣分。

三、交作业方式及时间

1. 提交题目的源代码文件、解题报告（word 版本）；
2. 截至时间：6 月 27 日
3. 提交到教学云平台