

预览输出

JSOI2019 Round 2

时间：2019 年 4 月 28 日 08:00 ~ 13:00

题目名称	精准预测	神经网络	节日庆典
题目类型	传统型	传统型	传统型
可执行文件名	<code>predict.exe</code>	<code>neural.exe</code>	<code>celebrate.exe</code>
输入文件名	<code>predict.in</code>	<code>neural.in</code>	<code>celebrate.in</code>
输出文件名	<code>predict.out</code>	<code>neural.out</code>	<code>celebrate.out</code>
每个测试点时限	2.0 秒	3.0 秒	1.0 秒
内存限制	1024 MiB	1024 MiB	1024 MiB
测试点数目	10	20	10
测试点是否等分	是	是	是
预测试点数目	2	2	2

提交源程序文件名

对于 C++ 语言	<code>predict.cpp</code>	<code>neural.cpp</code>	<code>celebrate.cpp</code>
-----------	--------------------------	-------------------------	----------------------------

编译选项

对于 C++ 语言	<code>-O2 -std=c++11 -Wl,--stack=1073741824</code>
-----------	--

精准预测 (predict)

【题目背景】

JYY 和他的火星探险队再次登录火星小镇，并且打算把机器学习知识传授给火星星人，从而提高火星人的生活效率。但智力有限的火星星人纷纷表示不相信计算机科学。为了让火星星人彻底信服，JYY 的探险队找到了他们之前关于火星详细的数据记录，并且训练了一个预测模型，这个模型能准确地预测出火星星人在未来的生死情况。

【题目描述】

目前，火星小镇上有 n 个居民（编号 $1, 2, \dots, n$ ）。机器学习算法预测出这些居民在接下来 T 个时刻（编号 $1, 2, \dots, T$ ）的生死情况，每条预测都是如下两种形式之一：

- **难兄难弟** $0\ t\ x\ y$: 在 t 时刻，如果 x 是死亡状态，那么在 $t+1$ 时刻， y 是死亡状态。（注意，当 x 在 t 时刻是生存状态时，该预测也被认为是正确的）
- **死神来了** $1\ t\ x\ y$: 在 t 时刻，如果 x 是生存状态，那么在 t 时刻， y 是死亡状态。（注意，当 x 在 t 时刻是死亡状态时，该预测也被认为是正确的）

（注意本题是对某个时刻进行生死状态的预测，如果某个人在 t 时刻是生存状态，在 $t+1$ 时刻是死亡状态，你可以认为是在 t 到 $t+1$ 这段时间内发生了某个事件导致其死亡。）

虽然 JYY 对自己从大数据中统计得到的模型非常自信，但火星星人看到这些预测吓了一跳，表示实在难以接受这种设定，更是认为计算机科学是可怕的邪教，打破了他们平静的生活。为了安抚火星星人的情绪，JYY 打算从这些预测结果中推导出一些火星星人更容易接受的事实，从而安抚火星星人的情绪。

具体来说，JYY 首先假设对火星星人生死的预测全部正确。在此基础上，JYY 希望为小镇上的每个居民 k 分别计算有多少个火星星人有可能和他一起活到第 $T+1$ 时刻。换言之，JYY 希望为每个火星星人 k 计算

$$\sum_{1 \leq i \leq n, i \neq k} Live(k, i),$$

其中 $Live(i, j) = 1$ 表示编号为 i 和 j 的火星星人可以同时在第 $T+1$ 时刻处于生还状态，否则 $Live(i, j) = 0$ 。

注意火星星人是不能够复活的。一个火星星人可能在时刻 1 就处于死亡状态，也有可能发生预测未覆盖的死亡情况（火星星人在任何时候都可能死亡，但任意时刻观察到火星星人的状态要么活着，要么死亡）。最后，注意到 $Live$ 是为每一对火星星人分别定义的，因此 $Live(x, y) = Live(y, z) = 1$ 并不意味着 $Live(x, z) = 1$ 。

【输入格式】

从文件 `predict.in` 中读入数据。

输入第一行包含三个整数 T, n, m 。

接下来有 m 行，每行表示一条预言，每条预言第一个整数 c 表示预言的类型：

- $c = 0$ ：接下来读入 t, x, y

- $c = 1$ ：接下来读入 t, x, y

输入数据保证 $1 \leq t \leq T, 1 \leq x, y \leq n$ 。本题内存限制 1024MiB 包含进程中运行库、堆栈等所有内存，请特别留意。

【输出格式】

输出到文件 *predict.out* 中。

输出 n 个数表示答案，用空格分割。

【样例 1 输入】

```
1 3 3 2
2 0 2 1 3
3 1 1 2 3
```

【样例 1 输出】

```
1 2 1 1
```

【样例 1 解释】

如果编号为 2 的火星人类活到 $T + 1$ 时刻，意味着在 1 时刻他也是活着的，由于第二条预言，会观察到编号为 3 的火星人在时刻 1 是死亡状态，所以编号为 2 和 3 的火星人类不能同时活到 $T + 1$ 时刻，所以 $Live(2, 3) = 0$ 。

而编号为 1 的火星人类能和其他两人的某一个活到 $T + 1$ 时刻。注意当 1 和 3 同时活着的时候，第一条预言是正确的。所以有 $Live(1, 2) = 1, Live(1, 3) = 1$ 。

【样例 2】

见选手目录下的 *predict/predict2.in* 与 *predict/predict2.ans*。

【子任务】

测试点	T	n	m
1	≤ 2	≤ 10	≤ 10
2,3	$\leq 10^2$	$\leq 10^2$	≤ 200
4	$\leq 10^6$	$\leq 3,000$	$\leq 6,000$
5,6	$\leq 4 \times 10^4$	$\leq 10^4$	$\leq 2 \times 10^4$
7	$\leq 10^6$	$\leq 3 \times 10^4$	$\leq 6 \times 10^4$
8		$\leq 4 \times 10^4$	$\leq 8 \times 10^4$
9,10		$\leq 5 \times 10^4$	$\leq 10^5$

神经网络 (neural)

【题目背景】

火星探险队发现，火星人的思维方式与人类非常不同，是因为他们拥有与人类很不一样的神经网络结构。为了更好地理解火星人的行为模式，JYY 对小镇上火星人的大脑进行了扫描，得到了一些重要数据。

【题目描述】

火星人在出生后，神经网络可以看作是一个由若干无向树 $\{T_1(V_1, E_1), T_2(V_2, E_2), \dots, T_m(V_m, E_m)\}$ 构成的森林。随着火星人年龄的增长，神经连接的数量也不断增长。初始时，神经网络中生长的连接 $E^* = \emptyset$ 。神经网络根据如下规则生长：

- 如果节点 $u \in V_i, v \in V_j$ 分别属于不同的无向树 T_i 和 T_j ($i \neq j$)，则 E^* 中应当包含边 (u, v) 。

最终，在不再有神经网络连接可能生长后，神经网络之间的节点连接可以看成是一个无向图 $G(V, E)$ ，其中

$$V = V_1 \cup V_2 \cup \dots \cup V_m, E = E_1 \cup E_2 \cup \dots \cup E_m \cup E^*.$$

火星人的决策是通过在 $G(V, E)$ 中建立环路完成的。针对不同的外界输入，火星人会建立不同的神经连接环路，从而做出不同的响应。为了了解火星人行为模式的复杂性，JYY 决定计算 G 中哈密顿回路的数量。

$G(V, E)$ 的哈密顿回路是一条简单回路，从第一棵树的第一个节点出发，恰好经过 V 中的其他节点一次且仅一次，并且回到第一棵树的第一个节点。

【输入格式】

从文件 `neural.in` 中读入数据。

第一行读入 m ，表示火星人神经网络初始时无向树的数量。

接下来输入有 m 部分，第 i 部分描述了树 T_i 。

对于 T_i ，输入的第一行是树 $T_i(V_i, E_i)$ 中节点的数量 k_i 。假设 $V_i = \{v_1, v_2, \dots, v_{k_i}\}$ 。

接下来 $k_i - 1$ 行，每行两个整数 x, y ，表示该树节点 v_x, v_y ($1 \leq x, y \leq k_i$) 之间有一条树边，即 $(v_x, v_y) \in E_i$ 。

【输出格式】

输出到文件 `neural.out` 中。

因为哈密顿回路的数量可能很多，你只需要输出一个非负整数，表示答案对 998244353 取模后的值。

【样例 1 输入】

```

1 2
2 3
3 1 2
4 1 3
5 2
6 1 2

```

【样例 1 输出】

```

1 12

```

【样例 1 解释】

在这个样例中, G 中一共有 5 个点, 只有第一棵树的 2, 3 节点之间没有边, 其他任意两个点之间都存在边. 这样的图哈密顿回路个数为 12.

【样例 2】

见选手目录下的 *neural/neural2.in* 与 *neural/neural2.ans*。

【子任务】

测试点	$\sum_{i=1}^m k_i$	m	k_i	树形态限制
1	≤ 10	≤ 3	无限制	无限制
2	≤ 15	≤ 4		
3	≤ 600	≤ 300	≤ 2	
4,5,6	≤ 900		≤ 3	
7,8,9,10	2, 500	50	50	
11,12,13,14	$\leq 5,000$	≤ 300	无限制	每棵树都是链
15,16,17,18,19,20				无限制

节日庆典 (celebrate)

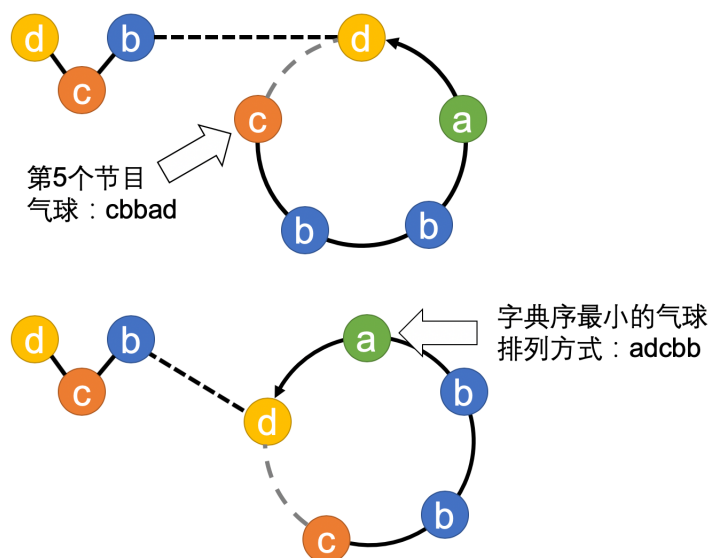
【题目背景】

JYY 和探险队顺利完成了火星上的任务。在离开前，探险队正好赶上了火星一年一度最盛大的节日“气球节”。然而，火星遇到了每年一度的麻烦：怎样最美观地摆放气球。JYY 决定请你设计算法帮助火星解决这个问题。

【题目描述】

在庆典开始前，火星会把气球准备好并串在一根绳子上。气球按顺序排列可以看成是一个由小写字母组成的长度为 n 的字符串 S 。然后，火星会按照字符串的顺序逐个把气球加入到一个庆典的圆环上，并且表演一个节目庆祝。

下图展示了一串气球 `cbbad` 在进行到第 5 个节目时的情形，此时在庆典环上的气球是 `cbbad`。



为了让每个节目都更好看，火星希望在每个节目开始前调整气球在环上的顺序，使得每个节目的气球排布都最美观。对于一组气球（一个字符串），火星认为最美观的字符串是庆典圆环上按绳子方向读出字典序最小的字符串，例如对于 `cbbad`，共有 5 个读出字符串的位置：

- `cbbad` ($i = 1$)
- `bbadc` ($i = 2$)
- `badcb` ($i = 3$)
- `adcb` ($i = 4$)
- `dcbb` ($i = 5$)

如果有多个字典序最小的字符串，火星人希望找出离绳头最近的那个（即 i 最小的那个）。更严谨地说，对于字符串 T ，定义

$$T_i = T[i \dots |T|] :: T[1 \dots i - 1] \quad (1 \leq i \leq |T|),$$

其中 $::$ 是字符串的拼接操作。定义 $f(T)$ 为最小的 i ($1 \leq i \leq |T|$) 满足 $T_i = \min\{T_1, T_2, \dots, T_{|T|}\}$ 。

JYY 希望你帮助他设计一个算法，让火星人每个节目的气球排列都最美观，即对于给定字符串 S 的每一个前缀 $S[1 \dots i]$ ($1 \leq i \leq |S|$)，求出 $f(S[1 \dots i])$ 。

【输入格式】

从文件 *celebrate.in* 中读入数据。

输入文件只有一行，该行包括一个字符串 S 。

【输出格式】

输出到文件 *celebrate.out* 中。

在一行中对于每个 i ($1 \leq i \leq |S|$)，输出一个整数 $f(S[1 \dots i])$ 。输出的数字之间以空格分隔。

【样例 1 输入】

```
1 abaacaba
```

【样例 1 输出】

```
1 1 1 3 3 3 6 3 8
```

【样例 2】

见选手目录下的 *celebrate/celebrate2.in* 与 *celebrate/celebrate2.ans*。

【子任务】

测试点	n	是否随机生成?	生成限制
1	≤ 20	否	无
2,3	$\leq 10^3$	是	字符集为 $\{a, b, c\}$
4	$\leq 2 \times 10^5$		
5	$\leq 5 \times 10^4$	否	每个字符出现次数相等
6,7	$\leq 10^6$		无
8,9,10	$\leq 3 \times 10^6$		