

# 2020保研-夏令营机试

## T1 元数据

时间限制：1.0 秒

空间限制：512 MB

### 题目描述

文件系统管理每个文件需要记录文件的相关属性，这些属性数据叫做元数据。有一个简单的文件系统，元数据需要记录文件大小。该文件系统一共管理着  $n$  个文件，编号分别为 1 到  $n$ 。该文件系统支持两种基本操作，在文件末尾追加新的数据和在文件开头删除旧的数据，前者会增加文件大小，后者会减少文件大小。

- $A\ x\ y$ ，表示第  $x$  号文件增加  $y$  字节的文件大小。
- $D\ x\ y$ ，表示第  $x$  号文件减少  $y$  字节的文件大小。

每个文件初始大小为 0 字节，该文件系统保证文件大小在  $[0, 10^9]$  字节之间。如果某个操作超出了这个区间，未超出部分会正常执行，超出部分会直接被忽略掉。现在给出  $m$  个操作记录，求出  $m$  个操作完成以后，各个文件的文件大小，并且按照从 1 到  $n$  的顺序依次输出每个文件的文件大小。

### 输入格式

从标准输入读入数据。

输入的第一行包含两个正整数  $n$  和  $m$ ，保证  $n, m \leq 10^6$ 。

接下来  $m$  行表示  $m$  个操作，保证  $1 \leq x \leq 10^6$  和  $0 \leq y \leq 10^9$ 。

### 输出格式

输出到标准输出。

输出  $n$  行，第  $i$  行一个整数表示第  $i$  号文件的文件大小。

### 样例 #1

#### 样例输入 #1

```
1 2 5
2 A 1 600000000
3 A 1 600000000
4 D 2 1
5 A 2 1
6 D 1 1000000000
```

## 样例输出 #1

1	0
2	1

## 子任务

测试点编号	$n \leq$	$m \leq$	分数
1	1	$10^6$	10
2	1	$10^6$	10
3	$10^2$	$10^2$	10
4	$10^2$	$10^2$	10
5	$10^4$	$10^4$	10
6	$10^4$	$10^4$	10
7	$10^4$	$10^4$	10
8	$10^6$	$10^6$	10
9	$10^6$	$10^6$	10
10	$10^6$	$10^6$	10

## T2 简单文件系统

时间限制： 1.0 秒

空间限制： 512 MB

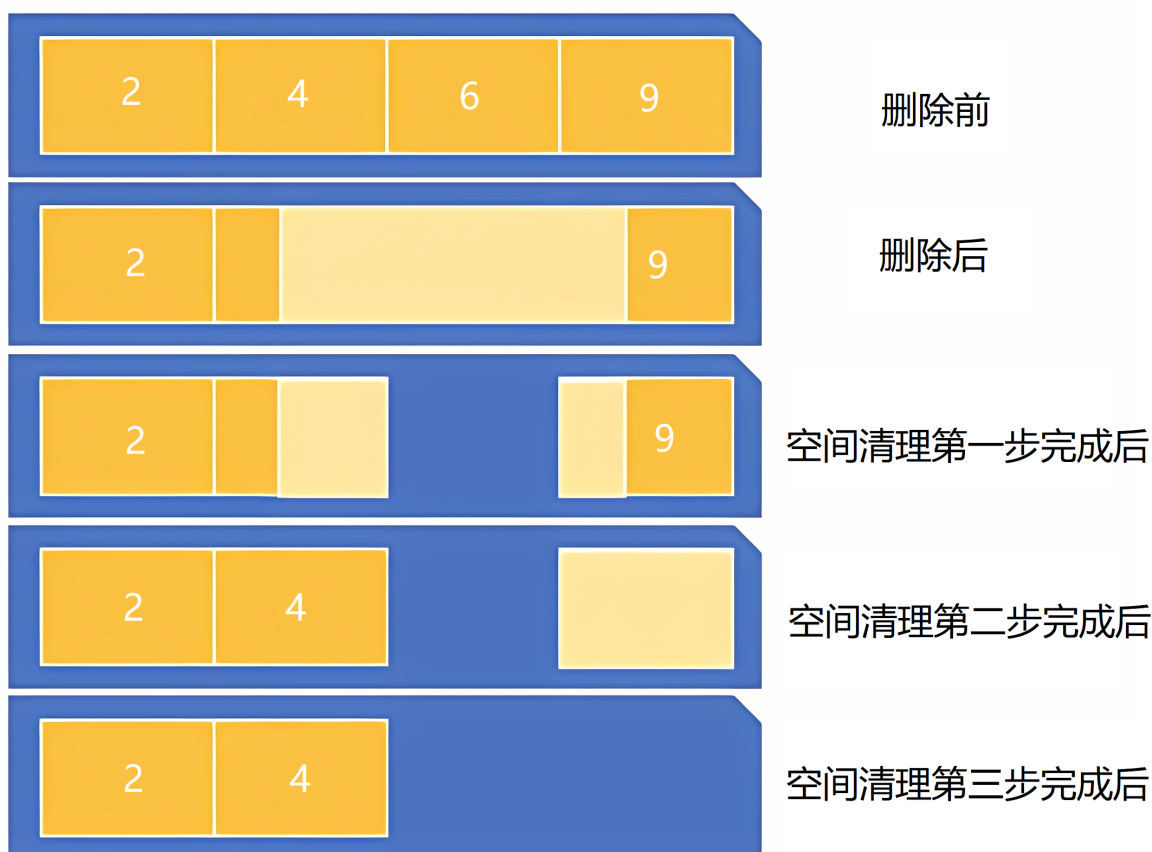
### 题目描述

在一个简单文件系统中，有  $n$  个文件，编号从 1 到  $n$ 。每个文件存储的数据量，即文件内容的长度，在该文件系统中，叫做“文件大小”，初始文件大小皆为 0 字节。学过计算机系统相关知识的同学都知道，每个文件在硬盘存储时，是以“块”为存储单元。每个文件在硬盘中拥有的存储空间，在该文件系统中，叫做“文件容量”。任何时刻文件容量一定是块大小的非负整数倍。比如，块大小为 100 字节，则任一文件在任何时刻的文件容量一定是 100 字节的非负整数倍。每个文件存储的数据量肯定是不超过其文件容量的，即文件大小不超过文件容量。该文件系统一共有  $m$  个块，编号从 1 到  $m$ ，块大小为  $b$ ，每个块有“已占用”和“未占用”两种状态。

学过计算机系统相关知识的同学都知道，文件内容在逻辑上是连续的，但是在硬盘存储上却不一定是连续的。如图所示，块大小为 100 字节，某个文件大小为 255 字节，文件容量为 300 字节。在逻辑上，文件内容是连续的，但是在硬盘中，是由 3 个编号不连续的块存储着这些数据。该文件系统支持“增改”、“删除”、“查询块”、“查询内容”四种文件操作对文件中的数据进行处理。文件操作可以从某个偏移量 `off` 开始操作。将整个文件内容看作一个数组，`off` 就是数组下标，从 0 开始。



- $A \ x \ off \ st$  表示以第  $x$  号文件的  $off$  位置为起点写入数据  $st$ 。如果  $off$  超过文件尾部，则直接从文件尾部开始写入数据。如果  $off$  及其后面存在旧数据，新数据会覆盖旧数据。在执行写入操作的过程中，如果文件容量不足，该文件系统会为该文件分配“未占用”块以增加文件容量。该文件系统按编号从小到大分配“未占用”块，然后将这些被分配的块标记为“已占用”。如果文件系统“未占用”块不足，则无法增加文件容量，该操作终止，后续部分无效。
- $D \ x \ off \ y$  表示以第  $x$  号文件的  $off$  位置为起点删除  $y$  字节数据。如果  $off$  超过文件尾部，则该操作无效。如果以第  $x$  号文件的  $off$  位置为起点，已经删除后面所有文件的内容，还未到达  $y$  字节，该操作终止，后续部分无效。数据删除后，该文件系统会进行空间清理。如图所示，删除后会有一个“空洞”。



浅黄色表示删除部分，也是未存储数据的部分

- 空间清理的第一步是，找出数据全部被删除的“已占用”块，将其标记为“未占用”，文件容量也随之减小。如图所示，删除第 6 块。
- 空间清理的第二步是，连接  $off$  前后的数据。假设  $off$  前最后一块数据对应第  $bo_0$  块， $off$  开始的数据对应第  $bo_1, bo_2, \dots, bo_e$  块。按照  $i = 0, 1, \dots, e$  的顺序处理第  $bo_i$  块数据。先对第  $bo_i$  块内部数据  $bo_i^0, bo_i^1, \dots, bo_i^{b-1}$  先进行处理。块内位置  $bo_i^l, bo_i^{l+1}, \dots, bo_i^r$  ( $0 \leq l \leq r \leq b-1$ ) 没有数据。如果  $r < b-1$ ，从  $bo_i^{r+1}$  开始将数据前移，否则直接从后面块开始将数据前移，直到第  $bo_i$  块内存满数据，或者后面块没有数据皆为空。注意，在数据移动过程中，原有位置的数据移动到新位置，原有位置就不再保留数据。第  $bo_i$  块处理完成后，继续处理后一块数据。处理完第  $e$  块数据，继续进行空间清理的第三步。如图所示，将第 9 块数据前移到第 4 块。

- 空间清理的第三步是，继续找出数据全部被删除的“已占用”块，将其标记为“未占用”，文件容量也随之减小。如图所示，删除第 9 块。
- B**  $x$  表示查询第  $x$  号文件拥有的“已占用”块。先返回一个块数，再按文件先后顺序输出“已占用”块的编号。
- C**  $x$   $off$   $y$  表示以第  $x$  号文件的  $off$  位置为起点查询  $y$  字节数据。如果  $off$  超过文件尾部，返回 -2。如果  $off$  开始的文件内容不足  $y$  字节，则返回 -1，否则输出对应的  $y$  字节数据。

现在给出  $k$  个操作记录，要求实现这个简单文件系统，并输出查询操作的返回结果。保证写入的数据仅有大小写字母，没有其他字符，每个字母占 1 字节的空间。保证  $off$  一定是非负整数，并且不超过  $10^9$ 。保证 **A** 操作的总写入数据量不超过  $10^6$  字节。

题目保证任意时刻任意文件的大小不超过 1500。

## 输入格式

从标准输入读入数据。

输入的第一行包含四个正整数  $n, m, k, b$ ，表示  $n$  个文件， $m$  个块， $k$  个操作，以及块大小  $b$ 。保证  $n \leq 10^3$ ， $m \leq 10^5$ ， $k \leq 10^5$ ， $b \leq 10^2$ 。

接下来  $k$  行表示  $k$  个操作，保证  $1 \leq x \leq n$ ， $0 \leq offset \leq 10^9$ ， $1 \leq y \leq 10^9$ ，**A** 操作中所有  $st$  长度之和不超过  $10^6$  字节，且  $st$  中仅有大小写字母，没有其他字符。

## 输出格式

输出到标准输出。

每个查询操作输出一行，一共若干行。

## 样例 #1

### 样例输入 #1

```

1 2 6 28 3
2 A 1 2 aAbB
3 B 1
4 C 1 0 4
5 C 1 2 2
6 C 1 2 3
7 A 1 2 cCdDeEfFgGhHiIjJkk
8 B 1
9 C 1 0 18
10 C 1 2 8
11 A 2 0 1L
12 B 2
13 C 2 0 1
14 D 1 4 12
15 B 1
16 C 1 0 18
17 C 1 0 6
18 A 2 0 1L
19 B 2
20 C 2 0 1
21 D 2 2 20
22 B 2
23 C 2 0 1

```

```

24 | C 2 0 2
25 | D 2 1 20
26 | B 2
27 | C 2 0 2
28 | C 2 0 1
29 | C 2 100 1

```

## 样例输出 #1

```

1 | 2 1 2
2 | aAbB
3 | bB
4 | -1
5 | 6 1 2 3 4 5 6
6 | aAcCdDeEfGgHhIiJ
7 | cCdDeEfF
8 | 0
9 | -2
10 | 2 1 2
11 | -1
12 | aAcCjJ
13 | 1 3
14 | 1
15 | 1 3
16 | 1
17 | 1L
18 | 1 3
19 | -1
20 | 1
21 | -2

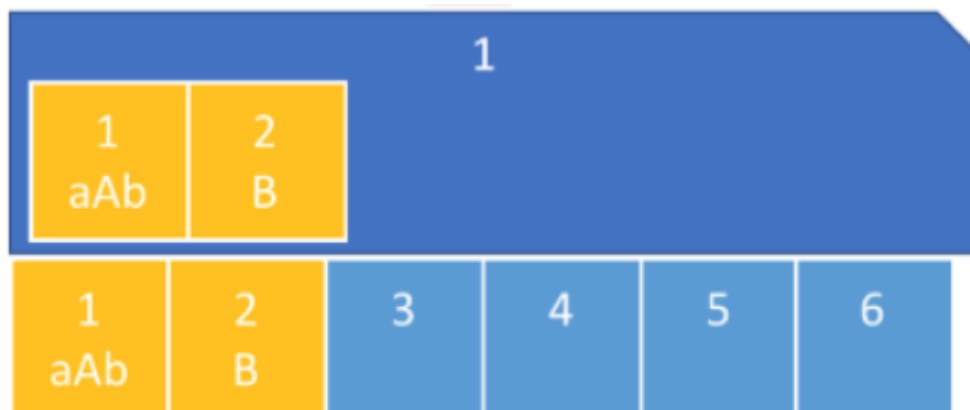
```

## 样例解释 #1

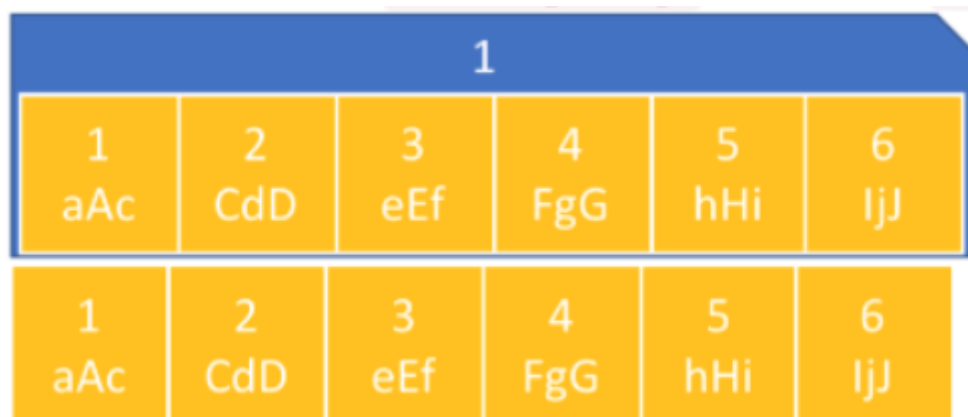
### 样例 1 解释

一共有 2 个文件，6 个块，28 个操作，块大小为 3。

1 号文件先写入数据 `aAbB`，拥有 1, 2 号块，如下图所示。

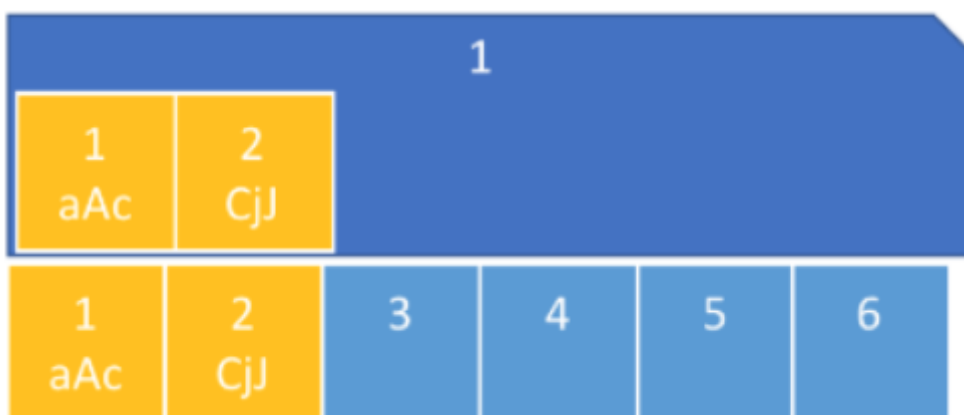


1 号文件然后写入 `cCdDeEfGgHhIiJ`，实际数据为 `aAcCdDeEfGgHhIiJ`，拥有 1, 2, 3, 4, 5, 6 号块，如下图所示。

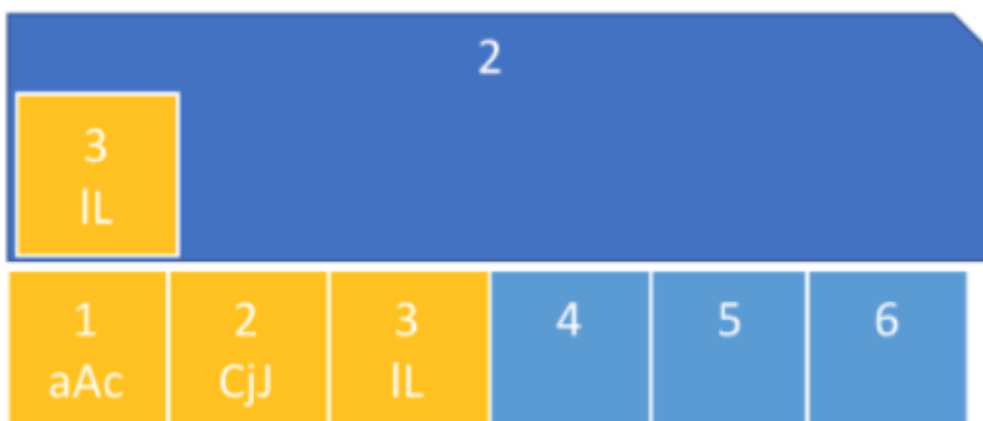


2 号文件先写入数据 1L，因为空间不足，写入失败。

1 号文件删除数据，实际数据为 aAcCjJ，如下图所示，3, 4, 5, 6 号块被回收。

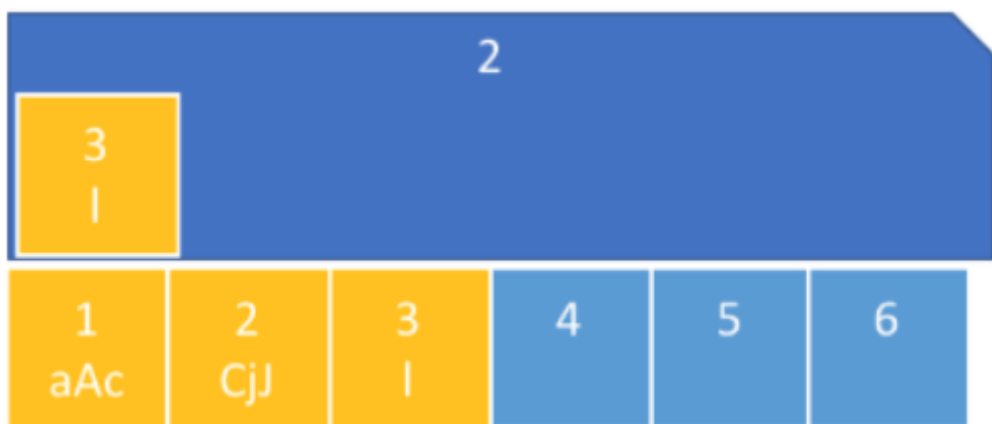


2 号文件再写入数据 1L，拥有 3 号块，如下图所示。



2 号文件删除数据，实际数据为 1L。

2 号文件删除数据，实际数据为 1，如下图所示。



## 样例 #2

### 样例输入 #2

```
1 2 8 22 3
2 A 2 0 aaa
3 B 2
4 C 2 0 3
5 A 1 10 aAbBcCdDeEfGhHiIjJ
6 B 1
7 C 1 0 20
8 A 1 1 JiIiHhGgFfEeDdCcBbAa
9 B 1
10 C 1 0 20
11 C 1 0 21
12 D 2 0 3
13 B 2
14 C 2 0 1
15 D 1 9 9
16 B 1
17 C 1 0 9
18 C 1 0 12
19 C 1 9 3
20 C 1 8 4
21 D 1 0 100
22 B 1
23 C 1 0 1
```

### 样例输出 #2

```
1 1 1
2 aaa
3 7 2 3 4 5 6 7 8
4 aAbBcCdDeEfGhHiIjJ
5 7 2 3 4 5 6 7 8
6 aJiIiHhGgFfEeDdCcBbA
7 aJiIiHhGgFfEeDdCcBbAa
8 0
9 -2
10 4 2 3 4 8
11 aJiIiHhGg
12 aJiIiHhGgbAa
13 bAa
```

14	gbAa
15	0
16	-2

## 子任务

测试点编号	$n \leq$	$m \leq$	$k \leq$	$b \leq$	操作说明	分数
1	$10^2$	$10^2$	$10^3$	$10^2$	只有 AB 操作	10
2	$10^2$	$10^2$	$10^3$	$10^2$	只有 AC 操作	10
3	$10^2$	$10^2$	$10^3$	$10^2$	只有 ABD 操作	10
4	$10^2$	$10^2$	$10^3$	$10^2$	只有 ACD 操作	10
5	$10^2$	$10^2$	$10^3$	$10^2$	所有操作	10
6	$10^3$	$10^5$	$10^5$	$10^2$	只有 AB 操作	10
7	$10^3$	$10^5$	$10^5$	$10^2$	只有 AC 操作	10
8	$10^3$	$10^5$	$10^5$	$10^2$	只有 ABD 操作	10
9	$10^3$	$10^5$	$10^5$	$10^2$	只有 ACD 操作	10
10	$10^3$	$10^5$	$10^5$	$10^2$	所有操作	10

## T3 连通图的方案数

时间限制： 1.0 秒

空间限制： 512 MB

### 题目描述

有一张  $n$  个点的无向图。总共有  $m$  条边，其中有些边已经存在于图中，剩下的边是可以加入的边，问有多少种加边的方案使得加边后的图是连通图。

对于两种加边的方案，如果存在至少一条边在其中一种方案加入了，但在另一种方案里没有加入，则这两种方案被视为不同方案。

在最后一档子任务中保证由已经存在的边构成的子图的连通块个数不大于 15。

输出结果需要对 998244353 取模。

### 输入格式

从标准输入读入数据。

输入的第一行包含两个整数  $n, m$ ，表示  $n$  个点， $m$  条边，保证  $n \leq 10^5$ ， $m \leq 4 \times 10^5$ 。

接下来  $m$  行，每行三个整数  $x, y, z$ ，表示在  $x$  和  $y$  之间有一条无向边，如果  $z = 0$  则这条边已经存在，否则  $z = 1$  则这条边可以加入。保证  $1 \leq x, y \leq n$ ， $0 \leq z \leq 1$ 。

### 输出格式

输出到标准输出。



输出一个整数，表示方案数对 998244353 取模的结果。

## 样例 #1

---

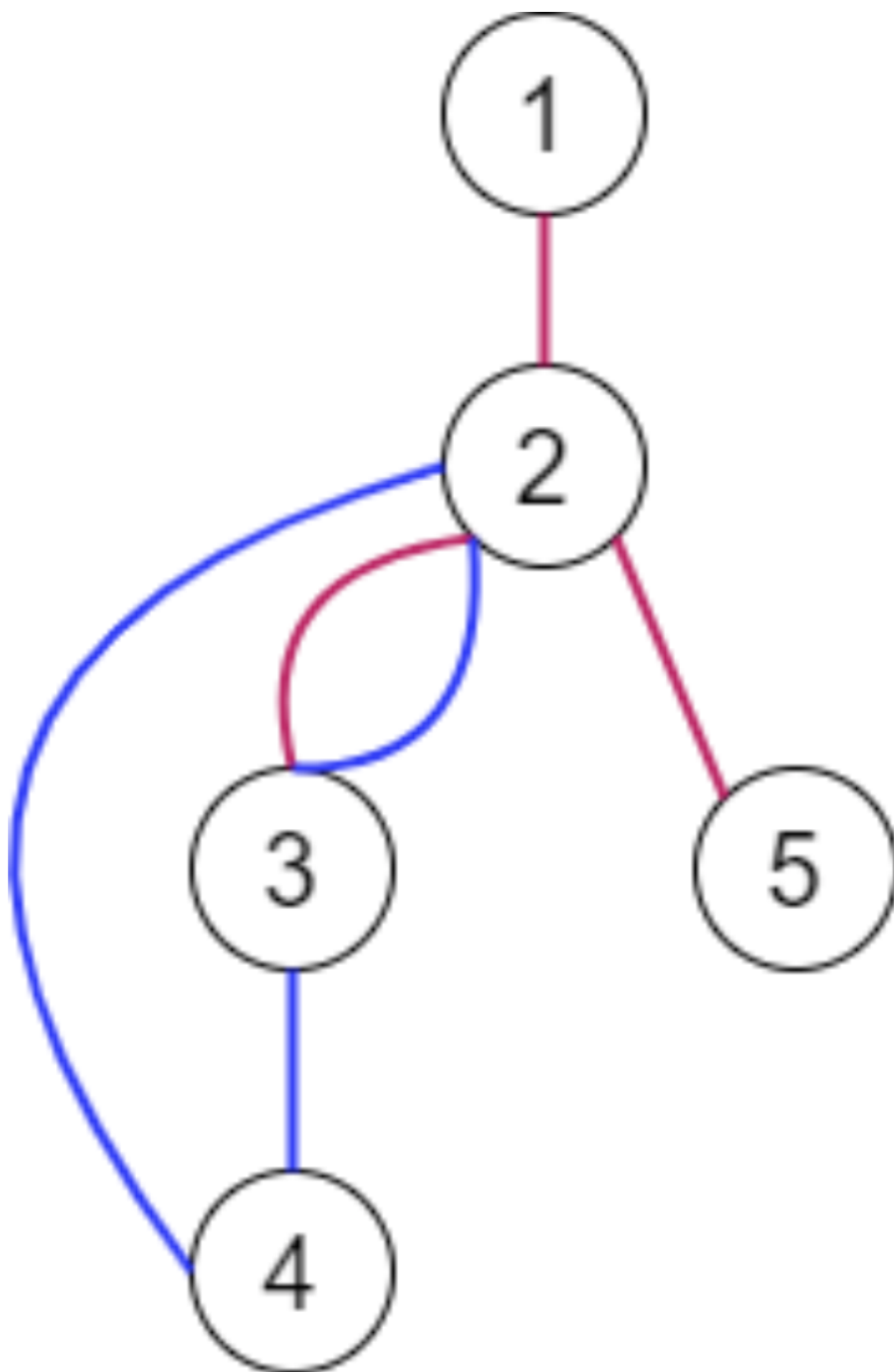
### 样例输入 #1

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

### 样例输出 #1

1	6
---	---

### 样例解释 #1



图中红色的边为已经存在的边，蓝色的边为可以加入的边。

输入的第 6 条边和第 4 条边两条边中必须存在一条，而输入的第 3 条边可有可无，所以一共有 6 种方案。

## 子任务

对于所有数据保证  $n \leq 10^5$ ,  $m \leq 4 \times 10^5$ 。

子任务 1 (13 分) : 保证  $m \leq 18$ 。

子任务 2 (15 分) : 保证  $m = n - 1$ 。

子任务 3 (19 分) : 保证  $n \leq 1000$ ,  $m \leq 4000$ , 且由已经存在的边构成的子图的连通块个数不大于 10。

子任务 4 (24 分) : 保证由已经存在的边构成的子图的连通块个数不大于 13。

子任务 5 (29 分) : 保证由已经存在的边构成的子图的连通块个数不大于 15。