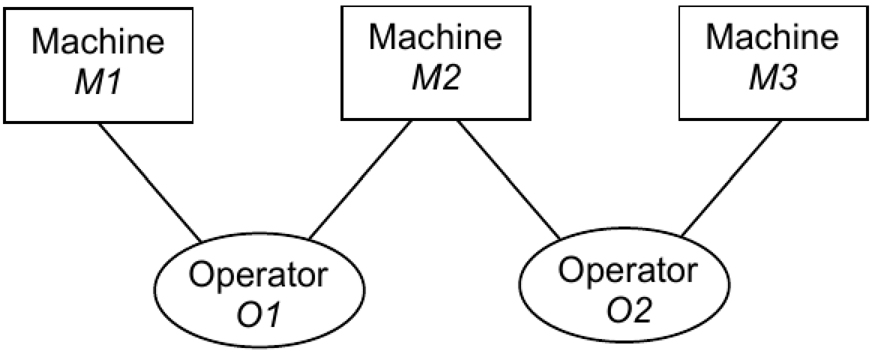
2213 Conflict Sets in Resource Sharing

In a manufacturing simulation, a *process route* specifies the sequence of various processing steps required to transform raw materials into finished products. Processing steps are arranged in the order in which they occur in the manufacturing of the product. Each processing step is represented by a record that contains information for that step. The record includes the *machine* and *operators* required for the step and the *processing time*. For example, test case 1 in the sample input specifies a process route with five processing steps. Machine $M1$ and operator $O1$ are used in the first step which requires a processing time of 0.1 time unit. Exactly one machine is used per processing step. However, zero, one or multiple operators may be required in a given processing step.

It is possible for two different machines to require the same operator to perform the operation (e.g., both processing steps 1 and 3 in test case 1 require operator $O1$ and both processing steps 3 and 4 require operator $O2$). In addition, different operators may be required to operate the machine in a given processing step (e.g., processing steps 3 in test case 1 uses both $O1$ and $O2$). This leads to situations as shown in the following figure, where machine $M1$ and $M2$ share operator $O1$, and machine $M2$ and $M3$ share operator $O2$.



Machines and operators are modeled as resources in simulation. The relationship between machines and operators can be defined as follows: machine X and operator Y are related to each other (written as $X \leftrightarrow Y$) if X and Y are required in the same processing step. Note that relation $\leftrightarrow$ is symmetric. An equivalence class can then be defined by the transitive closure of this relation (written as $\leftrightarrow^+$). For resources X and Y,

$X \leftrightarrow^+ Y$, if $X \leftrightarrow Y$; or
$X \leftrightarrow^+ Y$, if there exists a resource Z and $X \leftrightarrow^+ Z$, then $Z \leftrightarrow^+ Y$.

The edges shown in the above graph represent relation $\leftrightarrow$. Any two resources connected by a path must be in the same equivalence class.

To easily resolve the resource requirements in the simulation, it requires that machines in the same equivalence class are put together to form a *conflict set*. Thus, in the above example, $M1$, $M2$ and $M3$ form a conflict set.

In this problem, you are requested to develop a program that generates the following information from a process route:

1. *Conflict Sets*: Each conflict set consists of a machine or a group of machines according to the above definition. For test case 1 in the sample input, there are two conflict sets: $CS1$ and $CS2$. $CS1 = \{M1, M2, M3\}$ and $CS2 = \{M4\}$.

2. *Connectivity between Conflict Sets*: There is a direct link from conflict set CS' to conflict set CS" if there exist machines $M' \in CS'$ and $M" \in CS"$ so that M' and $M"$ are used in two consecutive processing steps. For test case 1, there is a link from $CS1$ to $CS2$, since $M1 \in CS1$, $M4 \in CS2$, and $M1$ and $M4$ are used in two consecutive processing steps (i.e., steps 1 and 2). Similarly, there is also a link from $CS2$ to $CS1$.

3. *Weight of Conflict Set*: The weight of a conflict set is defined as the aggregated processing time of all the steps involved by the machines in the conflict set. For example, the weight of conflict set $CS1$ is 2.6 which is the summation of processing times at steps 1, 3, 4 and 5 (i.e., all steps involved by machines $M1$, $M2$ and $M3$).

Input

The input consists of a number of test cases, each of which describes a process route and ends with a '#' sign. A process route consists of a number of lines, each of which describes the machine, operators and processing time required for a processing step. It is given in the following format:

step_id machine_id list_of_operator_ids processing_time

Step identifiers are specified using integers. Identifiers of machines and operators are specified using integers prefixed by characters 'M' and 'O' (letter, not zero), and processing times are specified using floating-point numbers (with a single decimal place). Note that there is exactly one machine per processing step. However, zero, one or multiple operators may be required at a given processing step.

Output

The definition of the conflict sets should be printed according to the following format (see the sample output below):

Total number of conflict sets
List of machine identifiers in each conflict sets

The connectivity between conflict sets and the weight of conflict set should be printed as a *connectivity matrix* according to the following definition (see the sample output below):

CM is a connectivity matrix.
$CM(i, i)$ contains the weight of conflict set i.
$CM(i, j)$, $i \neq j$, contains '1' if there is a direct link from conflict set i to conflict set j; otherwise it contains '0'.

Sample Input

```
1 M1 O1 0.2
2 M4 O3 1.0
3 M2 O1 O2 1.2
4 M3 O2 0.2
5 M1 1.0
#
1 M1 O1 0.5
2 M2 O1 0.5
3 M3 1.0
#
```

Sample Output

```
Test Case 1:
Total number of conflict sets: 2
Conflict Set 1: M1, M2, M3
Conflict Set 2: M4
Connectivity Matrix:
2.6, 1
1, 1.0

Test Case 2
Total number of conflict sets: 2
Conflict Set 1: M1, M2
Conflict Set 2: M3
Connectivity Matrix
1.0 1
0 1.0
```