Linker

Computer programs are usually composed of multiple modules. Compilation of such programs is often performed in two steps.

First, each source module is compiled separately into an object module. The object module consists of three parts:

- compiled program code,
- export table,
- import table.

The export table contains a list of symbols (e.g. variable or function names) that are defined in this module and may be accessed from other modules. The import table contains a list of external symbols this module depends on.

After all the modules are compiled, a special program called linker assembles them into an executable program.

Each program has an entry point — the place at which the execution of the program starts. The linker may perform an optimization by including into the executable only these modules on which the module containing the entry point depends. We will call the rest of the modules redundant.

Your task is to write a program that is given a description of the modules and finds:

- all non-redundant modules,
- all duplicate exports — export symbols that are exported from multiple modules and are imported into at least one non-redundant module,
- all unresolved imports — import symbols that are imported into at least one non-redundant module and are not exported from any module.

If the entry point is not exported from any module, we will call it an unresolved import as well.

Input. On the first line of the input file `LINKER.IN` is the name of the entry point. On the next line is the number of modules, N ($1 \leq N \leq 100$). The rest of the file contains N blocks describing the modules. The first line of a block contains the name of the module. The next line contains the number of exports, E ($0 \leq E \leq 100$). It is followed by E lines, each containing a distinct export symbol. The next line contains the number of imports, I ($0 \leq I \leq 100$). It is followed by I lines, each containing a distinct import symbol. All the names in this task are non-empty strings of length at most 30 and consist of latin lettes and digits.

Output. The first line of the output file `LINKER.OUT` must contain the number of non-redundant modules M_R, followed by M_R lines, each containing the name of a non-redundant module. The next line must contain the number of duplicate export symbols M_E, followed by M_E lines, each containing one duplicate export symbol. The next line must contain the number of unresolved import symbols M_I, followed by M_I lines, each containing one unresolved import.

Sample.

LINKER.IN

```
entry
4
main
1
entry
1
work
first
1
work
1
input
second
1
work
1
output
redundant
1
foo
1
bar
```

LINKER.OUT

```
3
main
first
second
1
work
2
input
output
```