Problem 1 - Internship

A software company is looking for its next star programmer, so they organize an interview in several rounds. In the first part, the candidates have to solve a few **Problems, ordered in a stack,** one after another, reading them from the console.

The number of Candidates may **not be the same** as the number of Problems prepared. The Candidates are solving the problems in a queue **one by one**, but only **if their name is valid**. A valid Candidate name consists of **First and Last name in proper casing** (e.g. Ivan Ivanov - is valid name; invalid names - ivan IVanov, Ivan ivanov...).

Each Candidate **takes the topmost** Problem to solve. If the sum of **Candidate`s name** letters (ASCII value) **is greater** from the sum of the **Problem`s letters** (ASCII value) => the **Problem is solved**. Otherwise, the Problem is **unsolved**. Don't forget to include space characters.

If a **Problem is solved,** it is **removed** from the stack with problems, and the Candidate who solved it goes to the end of the queue. You should write on the console: **`"{candidate} solved {problem}."`**

If a **Problem is unsolved**, it goes to the bottom of the stack and the **candidate fails the interview**, so he leaves in tears. You should write on the console: **`"{candidate} failed {problem}."`**

If there is **only one candidate left**, the program ends and you should write on the console: **`"{candidate} gets the job!"`**

In case there are no more tasks to solve you should print on the console the names of the candidates, separate by comma.

Input / Constraints

The input data should be read from the console.

- **First line** – **n** – Problems count – a 32-bit integer in the range [0 … 100].
- **Second line** – **m** – Candidates count – a 32-bit integer in the range [0 … 100].
- **Next n-lines** – a Problem – text in the range [1 … 100 chars].
- **Next m-lines** – a Candidate name – text in the range [1 … 100 chars].
- Allowed working time for your program: 0.1 seconds.
- Allowed memory: 16 MB.

Output

Print on the console each message in the above specified format.

Examples

Input	Output	Comments
5 5 Sum Bytes Word Count Text Filter Palindromes Hot Potato Filip Yordanov Vladimir Georgiev	Filip Yordanov solved Hot Potato. Vladimir Georgiev solved Palindromes. Borislava Pesheva solved Text Filter. Maria Mileva solved Word Count. Stanislav Stoyanov solved Sum Bytes. Filip Yordanov, Vladimir Georgiev, Borislava Pesheva, Maria Mileva, Stanislav Stoyanov	Filip Yordanov take the Problem – "Hot Potato". The sum of his name letters is 1382, the sum of Problem's name letters is – 962 -> the Problem is solved. Filip Yordanov goes to the end of the queue and the Problem is removed from the stack. Print –

Borislava Pesheva Maria Mileva Stanislav Stoyanov		“Filip Yordanov solved Hot Potato.” Vladimir Georgiev take the Problem – “Palindomes”….etc. Finally, there are no tasks left and we print the name of Candidates.
4 5 Greeting Snake Array Sum of All Elements of Matrix Pesho Goshov Georgi Ivanov Petyr Petrov Ivan ivanov Simo Simov	Pesho Goshov failed Sum of All Elements of Matrix. Georgi Ivanov solved Array. Petyr Petrov solved Snake. Simo Simov solved Greeting. Georgi Ivanov failed Sum of All Elements of Matrix. Petyr Petrov failed Sum of All Elements of Matrix. Simo Simov gets the job!	