Programming Fundamentals Final Exam Preparation 1

Problem 1. Secret Chat

Link: https://judge.softuni.org/Contests/Practice/Index/2307#0

You have plenty of free time, so you decide to write a program that conceals and reveals your received messages. Go ahead and type it in!

On the first line of the input, you will receive the **concealed message**. After that, until the **"Reveal"** command is given, **you will receive strings with instructions for different operations that need to be performed upon the concealed message** to **interpret it** and reveal its actual content. There are several types of instructions, split by **":|:"**

- **"InsertSpace:|:{index}"**:
  - Inserts a single **space at the given index**. The given index will always be valid.
- **"Reverse:|:{substring}"**:
  - If the message contains the given **substring**, **cut it out**, **reverse** it and **add** it at the **end** of the message.
  - If not, print only **"error"**.
  - This operation should replace only the first occurrence of the given **substring if there are two or more occurrences**.
- **"ChangeAll:|:{substring}:|:{replacement}"**:
  - Changes **all occurrences** of the given **substring** with the **replacement text**.

Input / Constraints

- On the first line, you will receive a string with a message.
- On the following lines, you will be receiving commands, split by **":|:"**.

Output

- After each **correct** set of instructions, print the resulting string.
- After the **"Reveal"** command is received, print this message:
  "You have a new text message: {message}"

Examples

Input	Output
heVVodar!gniV	hellodar!gnil
ChangeAll:|:V:|:l	hellodarling!
Reverse:|:!gnil	hello darling!
InsertSpace:|:5	You have a new text message: hello darling!
Reveal	
Comments	

Follow us:

ChangeAll:|:V:|:l

he**VV**odar!gni**V** -> he**ll**odar!gni**l** (We replace all occurrences of "V" with "l")

Reverse:|:!gnil

hellodar!gnil -> !gnil -> ling! -> hellodar**ling!** (We reverse **!gnil** to **ling!** And put it at the end of the string)

InsertSpace:|:5

hellodarling! -> hello darling! (We insert a space at index 5)

Finally, after receiving the **"Reveal"** command, we print the resulting message.

Input	Output
Hiware?uiy	Howare?uoy
ChangeAll:|:i:|:o	Howareyou?
Reverse:|:?uoy	error
Reverse:|:jd	How areyou?
InsertSpace:|:3	How are you?
InsertSpace:|:7	You have a new text message: How are you?
Reveal	

JS Examples

Input	Output
['heVVodar!gniV', 'ChangeAll:|:V:|:l', 'Reverse:|:!gnil', 'InsertSpace:|:5', 'Reveal']	hellodar!gnil hellodarling! hello darling! You have a new text message: hello darling!
Comments	

ChangeAll:|:V:|:l

he**VV**odar!gni**V** -> he**ll**odar!gni**l** (We replace all occurrences of "V" with "l")

Reverse:|:!gnil

hellodar!gnil -> !gnil -> ling! -> hellodar**ling!** (We reverse **!gnil** to **ling!** And put it in the end of the string)

InsertSpace:|:5

SoftUni

Follow us:

hellodarling! -> hello darling! (We insert a space at index 5)

Finally, after receiving the **"Reveal"** command, we print the resulting message.

Input	Output

<pre>[
  'Hiware?uiy',
  'ChangeAll:|:i:|:o',
  'Reverse:|:?uoy',
  'Reverse:|:jd',
  'InsertSpace:|:3',
  'InsertSpace:|:7',
  'Reveal'
]</pre>
	Howare?uoy
	
	Howareyou?
	
	error
	
	How areyou?
	
	How are you?
	
	You have a new text message: How are you?

Problem 2 - Mirror words

Link: https://judge.softuni.org/Contests/Practice/Index/2307#1

The SoftUni Spelling Bee competition is here. But it`s not like any other Spelling Bee competition out there. It`s different and a lot more fun! You, of course, are a participant, and you are eager to show the competition that you are the best, so go ahead, learn the rules and win!

On the first line of the input, you will be given a **text string**. To win the competition, you have to find all hidden **word pairs**, read them, and mark the ones that are **mirror images** of each other.

First of all, you have to **extract the hidden word pairs**. Hidden word pairs are:

- Surrounded by **"@"** or **"#"** (only one of the two) in the following pattern **#wordOne##wordTwo#** or **@wordOne@@wordTwo@**
- At least **3 characters long each** (**without the surrounding symbols**)
- Made up of **letters only**

If the second word, **spelled backward,** is the **same as the first word and vice versa** (**casing matters**!), they are a **match,** and you have to store them somewhere. **Examples** of mirror words:

#Part##traP# @leveL@@Level@ #sAw##wAs#

- If you don`t find any valid pairs, print: **"No word pairs found!"**
- If you find valid pairs print their count: **"{valid pairs count} word pairs found!"**
- If there are no mirror words, print: **"No mirror words!"**
- If there are mirror words print:

    "The mirror words are:

{wordOne} <=> {wordtwo}, {wordOne} <=> {wordtwo}, … {wordOne} <=> {wordtwo}"

Input / Constraints

- You will recive a string.

Output

- Print the proper output messages in the proper cases as described in the problem description.
- If there are pairs of mirror words, print them in the end, each pair separated by **", "**.
- Each pair of mirror word must be printed with **" <=> "** between the words.

Examples

Input
@mix#tix3dj#poOl##loOp#wl@@bong&song%4very$long@thong#Part##traP##@@leveL@@Level@#carrac##tu@pack@@ckap@#rr#sAw##wAs#r#@w1r

Output	Comments
5 word pairs found! The mirror words are: Part <=> traP, leveL <=> Level, sAw <=> wAs	There are 5 green and yellow pairs that meet all requirements and thus are valid. #poOl##loOp# is valid and looks very much like a mirror words pair, but it isn`t because the casings don`t match. #car#rac# "rac" spelled backward is "car", but this is not a valid pair because there is only one "#" between the words. @pack@@ckap@ is also valid, but "ckap" backward is "pakc" which is not the same as "pack", so they are not mirror words.

Input
#po0l##l0op# @bAc##cAB@ @LM@ML@ #xxxXxx##xxxXxx# @aba@@ababa@

Output	Comments
2 word pairs found! No mirror words!	"xxxXxx" backward is not the same as "xxxXxx" @aba@@ababa@ is a valid pair, but the word lengths are different - these are definitely not mirror words

Input
#lol#lol# @#God@@doG@# #abC@@Cba# @Xyu@#uyX#

Output	Comments

No word pairs found! No mirror words!	

JS Examples

Input
['@mix#tix3dj#poOl##loOp#wl@@bong&song%4very$long@thong#Part##traP##@@leveL@@Level@##car#rac##tu@pack@@ckap@#rr#sAw##wAs#r#@w1r']

Output	Comments
5 word pairs found! The mirror words are: Part <=> traP, leveL <=> Level, sAw <=> wAs	There are 5 green and yellow pairs that meet all requirements and thus are valid. #poOl##loOp# is valid and looks very much like a mirror words pair, but it isn`t because the casings don`t match. #car#rac# "rac" spelled backward is "car", but this is not a valid pair because there is only one "#" between the words. @pack@@ckap@ is also valid, but "ckap" backward is "pakc" which is not the same as "pack", so they are not mirror words.

Input
['#po0l##l0op# @bAc##cAB@ @LM@ML@ #xxxXxx##xxxXxx# @aba@@ababa@']

Output	Comments
2 word pairs found! No mirror words!	"xxxXxx" backward is not the same as "xxxXxx" @aba@@ababa@ is a valid pair, but the word lengths are different - these are definitely not mirror words

Input
['#lol#lol# @#God@@doG@# #abC@@Cba# @Xyu@#uyX#']

Output	Comments
No word pairs found! No mirror words!	

Problem 3 - Heroes of Code and Logic VII

Link: https://judge.softuni.org/Contests/Practice/Index/2303#2

You got your hands on the most recent update on the best MMORPG of all time – Heroes of Code and Logic. You want to play it all day long! So cancel all other arrangements and create your party!

On the first line of the standard input, you will receive an integer **n** – the number of heroes that you can choose for your party. On the next **n** lines, the heroes themselves will follow with their **hit points** and **mana points** separated by a single space in the following format:

"{hero name} {HP} {MP}"

-   **HP** stands for hit points and **MP** for mana points
-   a hero can have a **maximum** of **100 HP** and **200 MP**

After you have successfully picked your heroes, you can start playing the game. You will be receiving different commands, each on a new line, separated by **" – "**, until the **"End"** command is given.

There are several actions that the heroes can perform:

"CastSpell – {hero name} – {MP needed} – {spell name}"

-   If the hero has the required MP, he casts the spell, thus reducing his MP. Print this message:
    -   **"{hero name} has successfully cast {spell name} and now has {mana points left} MP!"**
-   If the hero is unable to cast the spell print:
    -   **"{hero name} does not have enough MP to cast {spell name}!"**

"TakeDamage – {hero name} – {damage} – {attacker}"

-   Reduce the hero HP by the given damage amount. If the hero is still alive (his HP is greater than 0) print:
    -   **"{hero name} was hit for {damage} HP by {attacker} and now has {current HP} HP left!"**
-   If the hero has died, remove him from your party and print:
    -   **"{hero name} has been killed by {attacker}!"**

"Recharge – {hero name} – {amount}"

-   The hero increases his MP. If it brings the MP of the hero above the **maximum value** (**200)**, MP is increased to **200**. (the MP can't go over the maximum value).
-   Print the following message:
    -   **"{hero name} recharged for {amount recovered} MP!"**

"Heal – {hero name} – {amount}"

-   The hero increases his HP. If a command is given that would bring the HP of the hero above the **maximum value (100)**, HP is increased to **100** (the HP can't go over the maximum value).
-   Print the following message:
    -   **"{hero name} healed for {amount recovered} HP!"**

Input

-   On the first line of the standard input, you will receive an integer **n**

Follow us:

- On the following **n** lines, the heroes themselves will follow with their **hit points** and **mana points** separated by a space in the following format
- You will be receiving different **commands**, each on a new line, separated by **" – "**, until the **"End"** command is given

Output

- Print all members of your party who are **still alive**, in the following format (their HP/MP need to be indented 2 spaces):

```
"{hero name}
  HP: {current HP}
  MP: {current MP}"
```

Constraints

- The starting HP/MP of the heroes will be valid, 32-bit integers will never be negative or exceed the respective limits.
- The HP/MP amounts in the commands will never be negative.
- The hero names in the commands will always be valid members of your party. No need to check that explicitly.

Examples

Input	Output

2
 	
Solmyr 85 120
 	
Kyrre 99 50
 	
Heal - Solmyr - 10
 	
Recharge - Solmyr - 50
 	
TakeDamage - Kyrre - 66 - Orc
 	
CastSpell - Kyrre - 15 - ViewEarth
 	
End
	Solmyr healed for 10 HP!
	
	Solmyr recharged for 50 MP!
	
	Kyrre was hit for 66 HP by Orc and now has 33 HP left!
	
	Kyrre has successfully cast ViewEarth and now has 35 MP!
	
	Solmyr
	
	  HP: 95
	
	  MP: 170
	
	Kyrre
	
	  HP: 33
	
	  MP: 35

Input	**Output**

Follow us:

4
 	
Adela 90 150
 	
SirMullich 70 40
 	
Ivor 1 111
 	
Tyris 94 61
 	
Heal - SirMullich - 50
 	
Recharge - Adela - 100
 	
CastSpell - Tyris - 1000 - Fireball
 	
TakeDamage - Tyris - 99 - Fireball
 	
TakeDamage - Ivor - 3 - Mosquito
 	
End
	SirMullich healed for 30 HP!
	
	Adela recharged for 50 MP!
	
	Tyris does not have enough MP to cast Fireball!
	
	Tyris has been killed by Fireball!
	
	Ivor has been killed by Mosquito!
	
	Adela
	
	  HP: 90
	
	  MP: 200
	
	SirMullich
	
	  HP: 100
	
	  MP: 40

Comments

Heal – SirMullich healed for 30 HP due to the HP max limit.

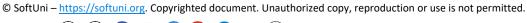Recharge – Adela recharged for 50 MP due to the MP max limit.

CastSpell – Tyris does not have enough MP to cast the spell.

TakeDamage – Tyris's HP is reduced by 99, thus becoming -5, which means he is dead.

TakeDamage – Ivor's HP is now -2, so he is dead too.

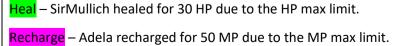After the "End" command, we print the remaining living heroes.

JS Examples

Input	Output
2 Solmyr 85 120 Kyrre 99 50 Heal - Solmyr - 10 Recharge - Solmyr - 50 TakeDamage - Kyrre - 66 - Orc CastSpell - Kyrre - 15 - ViewEarth End	Solmyr healed for 10 HP! Solmyr recharged for 50 MP! Kyrre was hit for 66 HP by Orc and now has 33 HP left! Kyrre has successfully cast ViewEarth and now has 35 MP! Solmyr HP: 95 MP: 170 Kyrre HP: 33 MP: 35

Input	Output
4 Adela 90 150 SirMullich 70 40 Ivor 1 111 Tyris 94 61 Heal - SirMullich - 50 Recharge - Adela - 100 CastSpell - Tyris - 1000 - Fireball TakeDamage - Tyris - 99 - Fireball TakeDamage - Ivor - 3 - Mosquito End	SirMullich healed for 30 HP! Adela recharged for 50 MP! Tyris does not have enough MP to cast Fireball! Tyris has been killed by Fireball! Ivor has been killed by Mosquito! Adela HP: 90 MP: 200 SirMullich HP: 100 MP: 40

Comments
Heal – SirMullich healed for 30 HP due to the HP max limit. Recharge – Adela recharged for 50 MP due to the MP max limit.

CastSpell – Tyris does not have enough MP to cast the spell.

TakeDamage – Tyris's HP is reduced by 99, thus becoming -5, which means that he is dead.

TakeDamage – Ivor's HP is now -2, so he is dead too.

After the "End" command, we print the remaining living heroes.