Problem 1 - Password Reset

Problem for exam preparation for the Programming Fundamentals Course @SoftUni.
Submit your solutions in the SoftUni judge system at https://judge.softuni.org/Contests/Practice/Index/2303#0.

Yet again, you have forgotten your password. Naturally, it's not the first time this has happened. Actually, you got so tired of it that you decided to help yourself with an intelligent solution.

Write a password reset program that performs a series of commands upon a predefined string. First, you will receive a string, and afterward, until the command **"Done"** is given, you will be receiving strings with commands split by a single space. The commands will be the following:

- **"TakeOdd"**
  - Takes only the characters at **odd indices** and **concatenates** them to obtain the **new raw password** and then **prints** it.
- **"Cut {index} {length}"**
  - Gets the substring with the **given length** starting from the **given index** from the password and removes its **first occurrence**, then **prints** the password on the console.
  - The given index and the length will **always** be **valid**.
- **"Substitute {substring} {substitute}"**
  - If the raw password contains the given substring, replaces all of its occurrences with the substitute text given and prints the result.
  - If it doesn't, prints **"Nothing to replace!"**.

Input

- You will be receiving strings until the **"Done"** command is given.

Output

- After the **"Done"** command is received, print:
  - **"Your password is: {password}"**

Constraints

- The indexes from the "**Cut {index} {length}**" command will always be valid.

Examples

Input	Output
Siiceercaroetavm!:?:ahsott.:i:nstupmomceqr TakeOdd Cut 15 3 Substitute :: - Substitute | ^ Done	icecream::hot::summer icecream::hot::mer icecream-hot-mer Nothing to replace! Your password is: icecream-hot-mer

Comments
TakeOdd -> We only take the chars at odd indices 1, 3, 5 etc. Siiceercaroetavm!:?:ahsott.:i:nstupmomceqr -> icecream::hot::summer **Cut 15 3 ->** We cut a substring starting at index 15 with length 3, then remove it from the raw password: icecream::hot::summer -> sum **Substitute :: - ->** We replace "::" with "-": icecream::hot::summer -> icream-hot-summer **Substitute | ^ -> "|"** is not found anywhere in the raw password, so we print **"Nothing to replace!"** Finally, after receiving the **"Done"** command, we print the resulting password in the proper format.

Input	Output
up8rgoyg3r1atmlmpiunagt!-irs7!1fgulnnnqy TakeOdd Cut 18 2 Substitute ! *** Substitute ? .!. Done	programming!is!funny programming!is!fun programming***is***fun Nothing to replace! Your password is: programming***is***fun

JS Examples

Input	Output
(["Siiceercaroetavm!:?:ahsott.:i:nstupmomceqr", "TakeOdd", "Cut 15 3", "Substitute :: -", "Substitute | ^", "Done"])	icecream::hot::summer icecream::hot::mer icecream-hot-mer Nothing to replace! Your password is: icecream-hot-mer

Comments
TakeOdd -> We only take the chars at odd indices 1, 3, 5 etc. Siiceercaroetavm!:?:ahsott.:i:nstupmomceqr -> icecream::hot::summer **Cut 15 3 ->** We cut a substring starting at index 15 with length 3, then remove it from the raw password: icecream::hot::summer -> sum

Substitute :: - -> We replace "::" with "-":

icecream::hot::summer -> icream-hot-summer

Substitute | ^ -> "|" is not found anywhere in the raw password, so we print **"Nothing to replace!"**

Finally, after receiving the **"Done"** command, we print the resulting password in the proper format.

Input	Output

`(["up8rgoyg3r1atmlmpiunagt!-irs7!1fgulnnnqy",`
`"TakeOdd",`
`"Cut 18 2",`
`"Substitute ! ***",`
`"Substitute ? .!.",`
`"Done"])`
	`programming!is!funny`
	`programming!is!fun`
	`programming***is***fun`
	`Nothing to replace!`
	`Your password is:`
	`programming***is***fun`

Problem 2 - Fancy Barcodes

Problem for exam preparation for the Programming Fundamentals Course @SoftUni.
Submit your solutions in the SoftUni judge system at https://judge.softuni.org/Contests/Practice/Index/2303#1.

Your first task is to determine if the given sequence of characters is a **valid** barcode or **not**.

Each line must not contain anything else but a valid barcode. A barcode is **valid** when:

- It is surrounded by a **"@"** followed by one or more **"#"**
- It is **at least 6 characters long** (without the surrounding **"@"** or **"#"**)
- **It starts** with a **capital letter**
- It contains **only letters** (lower and upper case) **and digits**
- **It ends** with a **capital letter**

Examples of valid barcodes: **@#FreshFisH@#**, **@###Brea0D@###**, **@##Che46sE@##**, **@##Che46sE@###**

Examples of invalid barcodes: **##InvaliDiteM##**, **@InvalidIteM@**, **@#Invalid_IteM@#**

Next, you have to determine the **product group** of the item from the **barcode**. The product group is obtained by **concatenating all the digits** found in the barcode. If there are **no digits** present in the barcode, the **default** product group is **"00".**

Examples:

@#FreshFisH@# -> product group: 00

@###Brea0D@### -> product group: 0

@##Che4s6E@## -> product group: 46

Input

On the first line, you will be given an integer **n** – the count of barcodes that you will be receiving next.

On the following **n** lines, you will receive different strings.

Output

For each barcode that you process, you need to print a message.

If the barcode is invalid:

- **"Invalid barcode"**

If the barcode is valid:

- **"Product group: {product group}"**

Examples

Input	Output
3 @#FreshFisH@#	Product group: 00 Product group: 0

@###Brea0D@### @##Che4s6E@##	Product group: 46

Input	Output
6 @###Val1d1teM@### @#ValidIteM@# ##InvaliDiteM## @InvalidIteM@ @#Invalid_IteM@# @#ValiditeM@#	Product group: 11 Product group: 00 Invalid barcode Invalid barcode Invalid barcode Product group: 00

JS Examples

Input	Output
(["3", "@#FreshFisH@#", "@###Brea0D@###", "@##Che4s6E@##"])	Product group: 00 Product group: 0 Product group: 46

Input	Output
(["6", "@###Val1d1teM@###", "@#ValidIteM@#", "##InvaliDiteM##", "@InvalidIteM@", "@#Invalid_IteM@#", "@#ValiditeM@#"])	Product group: 11 Product group: 00 Invalid barcode Invalid barcode Invalid barcode Product group: 00

Problem 3 - Heroes of Code and Logic VII

Problem for exam preparation for the Programming Fundamentals Course @SoftUni.
Submit your solutions in the SoftUni judge system at https://judge.softuni.org/Contests/Practice/Index/2303#0.

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**
- 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**

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.