Mini Exam: Sets and Dictionaries

You can check your solutions here: https://judge.softuni.bg/Contests/3178/Additional-Exercises.

1.Repeated Names

You will be given a sequence of **N names.** You need to keep a collection only of the **unique** ones. On the **first** line you will be given an integer **N**, it represents the count of names that you will receive. On the next **N** lines you will receive **one** name **per line**. Print the collection on the console in **order** of **insertion**:

Examples

Input	Output
6 Pesho Pesho Pesho Pesho Ivan Stamen	Pesho Ivan Stamen

2.Read and Count

You will receive **text** from the console. Your program has to read the text and **count** the **appearance** of **each** character. The results should be printed in **alphabetical** (lexicographical) order.

Examples

Input	Output
This is Sparta!	: 2 time/s !: 1 time/s S: 1 time/s T: 1 time/s a: 2 time/s h: 1 time/s i: 2 time/s p: 1 time/s r: 1 time/s s: 2 time/s t: 1 time/s
Houston, we have a problem.	: 4 time/s ,: 1 time/s .: 1 time/s H: 1 time/s a: 2 time/s b: 1 time/s e: 3 time/s h: 1 time/s l: 1 time/s

	m: 1 time/s n: 1 time/s o: 3 time/s p: 1 time/s r: 1 time/s s: 1 time/s t: 1 time/s u: 1 time/s v: 1 time/s w: 1 time/s

3.Classification

You are employed as an intern and your first job is to write a program that **classify** job candidates according to the points of the **interview tasks** and their **exam results**. You will receive **input** in the format **"{contest}:{password contest}"** until you receive **"the contests are ended"**. Save that data because **you will need it later**. After that until you receive **"the submissions are ended"** you will be given information for the candidates in format **"{contest}=>{password}=>{candidateName}=>{points}"**

Here is what you need to do:

- Check if the **contest is valid (if you received it in the first type of input)**
- Check if the **password is correct for the given contest**
- Save the candidate with the **contest** they take part in **(a candidate can take part in many contests)** and the points the candidate has in the **given contest**. If you receive the **same contest** and the **same candidate, update** the **points only if the new ones** are **more than** the **older ones.**

At the end you have to print the info for the best candidate, with the **most points** in the format: **"Candidate number one is {candidate} with total {total points} points."**. After that print **all candidates alphabetical ordered** by their **names**. For **each** candidate, print **each contest** and the **points** in **descending** order in the following format:

"{candidate1 name}

 # {contest1} -> {points}

 # {contest2} -> {points}

{candidate2 name}

…"

Input

- You will be receiving strings, until “the contests are ended” or “the submissions are ended”.

Output

- At **first** print the candidate who’s **number one** in the **described** format.
- On the **next** lines print all candidates ordered as mentioned above.

Constraints

- There will be **no** two **equal contests**.
- The **strings** may contain any ASCII character except from **(:, =, >).**
- The **numbers** will be in range **[0 - 10000]**.
- The **second** input will be always **valid**.
- There will be no **2** or **more** candidates with **same sum of points**.

Examples

Input	Output
Part One Interview:success Js Fundamentals:Pesho Programming Fundamentals:thePass Algorithms:fun the contests are ended Programming Fundamentals=>thePass=>Ivan=>350 Algorithms=>fun=>Ivan=>380 Part One Interview=>success=>Peter=>120 Java Basics Exam=>pesho=>Petkan=>400 Part One Interview=>success=>Ivan=>220 OOP Advanced=>password123=>BaiIvan=>231 Programming Fundamentals=>thePass=>Ivan=>250 Programming Fundamentals=>thePass=>Peter=>200 Js Fundamentals=>Pesho=>Ivan=>400 the submissions are ended	Candidate number one is Ivan with total 1350 points. Ranking: Ivan # Js Fundamentals -> 400 # Algorithms -> 380 # Programming Fundamentals -> 350 # Part One Interview -> 220 Peter # Programming Fundamentals -> 200 # Part One Interview -> 120
Java Advanced:somePass Part Two Interview:successPass Math Concept:asdasd Java Web Basics:forrF the contests are ended Math Concept=>ispass=>Petya=>290 Java Advanced=>somePass=>Kaloyan=>400 Part Two Interview=>successPass=>Drago=>120 Java Advanced=>somePass=>Petyr=>90 Java Web Basics=>forrF=>Kaloyan=>280 Part Two Interview=>successPass=>Petyr=>0 Math Concept=>asdasd=>Drago=>250 Part Two Interview=>successPass=>Kaloyan=>200 the submissions are ended	Candidate number one is Kaloyan with total 880 points. Ranking: Drago # Math Concept -> 250 # Part Two Interview -> 120 Kaloyan # Java Advanced -> 400 # Java Web Basics -> 280 # Part Two Interview -> 200 Petyr # Java Advanced -> 90 # Part Two Interview -> 0