Problem 3 – Greedy Times

Finally, you have unlocked the safe and reached the treasure! Inside there are all kinds of gems, cash in different currencies and gold bullions. Next to you there is a bag which unfortunately has a limited space. You don't have much time so you need take as much wealth as possible! But in order to get bigger amount of the most valuable items you need to keep the following rules:

- The **gold amount** in your bag should **always be more** than **or equal** the **gem amount** at **any** time
- The **gem amount** should **always be more** than **or equal** the **cash amount** at **any** time

If you read an **item** which **breaks this rule** you **should not put** it in the **bag**. You should **always** be careful **not** to **exceed** the overall **bag's capacity** as it will tear down you will lose everything! You will receive the **content of** the **safe** on a **single line** in the **format** of **item - quantity** pairs separated by **whitespace**. You need to gather **only three types** of items:

- Cash - All **three letter** items
- Gem - All **items** which **end** on "**Gem**" (at least 4 symbols)
- Gold - this type has **only one item** with the name - "**Gold**"

Each **item** which **does not** fall **in** one of the **above categories** is **useless** and you should **skip it**. Reading item's **names** should be **CASE-INSENSITIVE**. You should **aggregate item's quantities** which have the **same name**.

If you kept the rules you should have escaped successfully with a bag full of wealth. Now it's time to review what you have managed to get out of the safe. **Print all** the **types** ordered by **total amount** in **descending order**. Inside a type **order** the **items** first **alphabetically** in **descending** order and **then by** their **amount** in **ascending** order. Use the following format for each type:

"<{type}> ${total amount}"

"##{item} - {amount}" - each item from this type on new line

Input

- On the **first line**, you will receive a **number** which represents the **capacity** of the **bag**
- On the **second line**, you will receive a **sequence** of **item - quantity** pairs

Output

Print **only** the **types** from which you **have items in the bag** ordered by **`Total Amount`** descending. Inside a type order the **items first alphabetically** in **descending** order and **then** by **amount** in **ascending order**. Use the following format for each type:

"<{type}> ${total amount}"

"##{item} - {amount}" - each item on new line

Constraints

- Bag's **max capacity** will **always** be in the range [0 … 90000000000]
- All **quantities** will be **positive integer** in the range [0 … 2100000000]
- Each item of type **gem** will have a **name** - **at least 4** symbols
- Time limit: 0.1 sec. Memory limit: 16 MB

Examples

Input	Output
150 Gold 28 Rubygem 16 USD 9 GBP 8	<Gold> $28 ##Gold - 28 <Gem> $16 ##Rubygem - 16 <Cash> $9 ##USD - 9
24000010 USD 1030 Gold 300000 EmeraldGem 900000 Topazgem 290000 CHF 280000 Gold 10000000 JPN 10000 Rubygem 10000000 KLM 3120010	<Gold> $10300000 ##Gold - 10300000 <Gem> $10290000 ##Topazgem - 290000 ##Rubygem - 10000000 <Cash> $3410010 ##KLM - 3120010 ##JPN - 10000 ##CHF - 280000
80345 RubyGem 70000 JAV 10960 Bau 60000 Gold 80000	<Gold> $80000 ##Gold - 80000