# Cooking



*You are asked to cook baked foods for the near bakery, because your recipes are so great. You need to mix liquids with ingredients to cook the required delicacies.*

First, you will be given **a sequence of integers, representing liquids**. Afterwards, you will be given another **sequence of integers representing ingredients**.

You need to start from the **first liquid** and try to mix it with the **last ingredient.** If the **sum** of their values is **equal** to **any of the items in the table below** – **cook the food corresponding** to the **value** and **remove** **both** the **liquid** and the **ingredient**. Otherwise, **remove only the liquid** and **increase** the **value** of the **ingredient by 3**. You need to **stop** combining when you have **no more liquids** or **ingredients**.

|  |  |
| --- | --- |
| **Food** | **Value needed** |
| Bread | 25 |
| Cake | 50 |
| Pastry | 75 |
| Fruit Pie | 100 |

### Input

* On the **first line**, you will receive the integers representing the **liquids**, **separated** by a **single space**.
* On the **second line**, you will receive the integers representing the **ingredients**, **separated** by a **single space**.

### Output

* On the **first** line of output print one of the following ouputs:
  + "**Wohoo! You succeeded in cooking all the food!**" -if you have at least

**one of each** of the **foods**, after completing combining.

* + "**Ugh, what a pity! You didn't have enough materials to cook everything.**" – if you **did not** collect **one of each** of the **foods**, after completing combining.
* On the **second** line - print all liquids you have left:
  + If there are no liquids: "**Liquids left: none**"
  + If there are liquids: "**Liquids left: {liquid1}, {liquid2}, {liquid3},** **(…)**"
* On the **third** line - print all physical materials you have left:
  + If there are no items: "**Ingredients** **left: none**"
  + If there are items: "**Ingredients** **left: {ingredient}, {ingredient}, {ingredient},** **(…)"**
* Then**,** you need to print **all** AdvancedMaterials and the **amount you have of them**, ordered **alphabetically**:
  + **"Bread: {amount}"**
  + **"Cake: {amount}"**
  + **"Fruit Pie: {amount}"**
  + **"Pastry: {amount}"**

### Constraints

* All of the given numbers will be valid integers in the range **[0, 100]**.
* Advanced materials **can be** crafted more than once.

### Examples

|  |  |  |
| --- | --- | --- |
| ****Input**** | ****Output**** | ****Comment**** |
| **1 25 50 50**  **50 25 25 24** | **Wohoo! You succeeded in cooking all the food!**  **Liquids left: none**  **Ingredients left: none**  **Bread: 1**  **Cake: 1**  **Fruit pie: 1**  **Pastry: 1** | The first pair is the **first liquid** with value of 1 and the **last ingredient** of value 24, their **sum** is 25, so we **cook** Bread. Then we have **sum** of 50, we **cook** Cake. After that we have **sum** of 75, we **cook** Pastry. Next we have **sum** of 100, so we **craft** Fruit Pie. We have **no left liquids and/or ingredients**, so we **stop** trying to cook foods, but we **have enough** of them to **give them to the bakery.** |
| **10 20 30 40 50**  **50 40 30 30 15** | **Ugh, what a pity! You didn't have enough materials to cook everything.**  **Liquids left: none**  **Ingredients left: 39, 40, 50**  **Bread: 1**  **Cake: 1**  **Fruit pie: 0**  **Pastry: 0** | **First, we take **the first given liquid** and **the last ingredient**, their **sum** is 25 and we **cook** Bread, **removing** **both** of them from the collections. Then, we take the **next pair** and their **sum** is 50, **cooking** Cake and again – **removing both** the liquid and the ingredient. Next, we take the **next pair** and their **sum** is 60, so we **remove the liquid** and **increase** the **ingredient's** value by 3. The next 2 pairs follow **the same scenario**, so we end up with **not enough** materials for all the food, **no liquids left** and **some** **ingredients**, one of which is **39** (**originally 30**, **increased** its value **three** times).** |