# Problem 2. Entertrain

You must be good at entertaining people, now how about “**entertraining**” people. The process of entertraining people is releasing wagons when your locomotive doesn’t have enough power to carry them. You’ll have to write a program that simulates the process.

You will be given **N** – an **integer**, indicating the **locomotive’s power**. After that you will begin receiving integers, each on a **new line** – the **wagons** and their **weight**. You must keep the **wagons** in a **sequence**.

When the **total** **sum** of the **integers** in the sequencebecomes **GREATER** than the **locomotive’s power**, you must **calculate** the **average** of **the sequence**. Then you must take the **element** which is **closest** **BY VALUE** to the **average** and **REMOVE** it from the **sequence**. This is what **entertraining** means.

When you get the command “**All ofboard!**” you should **end** the **input sequence**. When that happens you should print the elements from the **sequence** of **wagons** in **reversed order** of **input**, with the **locomotive’s power** as the **last element**.

### Input

* On the **first line** you will get **N** – the **locomotive’s power**.
* On the **next** **several lines** you will be getting **integers** – the **wagons** and their **weight**.
* The **input ends** when you get the command “**All ofboard!**”.

### Output

* As output you must print the **sequence** of **wagons** in **reversed order** of **input**, with the **locomotive’s power** as **last element**.
* The elements must be **separated** by a **single space**.

### Constrains

* All **integers** in the **input** will be in **range [0, 1000]**.
* Note that you are **working** with **integers**, all **DIVISION** is **between integers**.
* There will never be **2 wagons** with the **same value**, **present** at the **same time**.
* Allowed time / memory: 100ms / 16 MB.

### Examples

|  |  |  |
| --- | --- | --- |
| **Input** | **Output** | **Comment** |
| 30  11  16  4  All ofboard! | 4 16 30 | The **locomotive’s power** is **30**. We start adding **wagons**.  **11**, **16** (**total** **weight** = **27**)  When we **add** **4,** the **total weight** = **31**, which is **>** **30**.  The **average** is **31 / 3** = **10**. The **closest** **wagon** by **value** to **10** is **11**, so we **remove** it.  We receive the final command, and we print the train in **reversed order** of **input** with the **locomotive** as **last**.  The **first entered** is **11**, but we **removed it**. The next is **16** and then **4**.  So, 4, 16, 30 = “**4 16 30**” |
| 50  11  20  25  All ofboard! | 25 11 50 |  |