Green Tech Sustainability

The Reply Code Masters Team

12 March 2025

1. Problem statement

Since 2021, Reply has adheared to the UN Global Compact committing to contribute to the achievement of **Sustainable Development Goals** described in the **UN Agenda 2030**.

The validity of Reply's Corporate Social Responsibility management model is recognised by rating agencies and **ESG indices**, which guarantee transparency in communication to investors and comparability with peers, as well as increasing the Group's visibility on financial market.

Why not increase Reply score by installing **photovoltaic panels** on the roofs of our beloved Headquarters?! So let's start with Milan K1 and Turin N250 in order to build two photovoltaic parks with equal total productivity by using all the available panels. Please, keep in mind that each panel has a specific KWh performance.

You will be provided with a catalogue of P photovoltaic panels, each with a specific performance in KWh. You will be requested to create two parks with equal total productivity. Please, use all the available panels only once.

2. Input format

The input file is a regular ASCII text file. Each line of the input file ends with a single $\backslash n$ character (UNIX-style). If a line contains multiple data, each value is separated by a single space character.

The first row of the input file will be composed of an integer number C, representing the number of test cases to be solved. The following rows represent for each test case C_i in ascending order:

- The integer number N_i, indicating the number of all panels to be installed
- N integer numbers P_1, P_2, ..., P_N separated by a single space. Each integer represents the power performance value generated by each photovoltaic panel

3. Output format

The output file must be a regular ASCII text file. Each line of the output file must end with a single $\backslash n$ character (UNIX-style). The rows represent for each test case, in ascending order:

- a string indicating the test case number, in the format $Case\ \#C_i$:
- a list of integer numbers: the first integer indicating the number of panels elegible for 1° Headquarter, the other integers indicating the power performance values of each panel, separated by a single space
- a list of integer numbers: the first integer indicating the number of panels elegible for 2° Headquarter, the other integers indicating the power performance values of each panel, separated by a single space

4. Constraints

- Each available panel must be installed.
- Each available panel must be allocated to only one headquarter.
- Each headquarter must have the same total photovoltaic productivity.
- Number of cases for each level: $1 \leq C_i \leq 10$
- Number of panels for each case: $1 \leq P_i \leq 500$
- Power of each panel: $1 \leq \text{KWh}_{P_i} \leq 500000$

5. Example

5.1. Input file example

```
2
4
5 1 6 2
6
21 14 16 8 7 20
```

In this example, players have 2 test cases to be solved. For the first case: 4 panels with power respectively equal to 5, 1, 6, 2 KWh. For the second case: 6 panels with power respectively equal to 21, 14, 16, 8, 7, 20 KWh.

5.2. Output file example

```
Case #1:
2 1 6
2 5 2
Case #2:
3 21 14 8
3 16 7 20
```

For the case C_1, the solution is composed by:

- two panels with power of 1, 6 KWh installed on the 1° Headquarter
- two panels with power of 5, 2 KWh installed on the 2° Headquarter

For the case C_2, the solution is composed by:

- three panels with power of 21, 14, 8 KWh installed on the 1° Headquarter
- three panels with power of 16, 7, 20 KWh installed on the 2° Headquarter

6. CTF recall

A well-optimized solar panel distribution requires precise calculations, but what if financial approval hinges on a locked system? The Solar Panels Office might hold crucial data. Maybe that CTF challenge about bypassing restricted access could provide insight. Dig into it, securing the best panels might take more than just math.