Grades

Mr. Panda is in charge of processing the grades of CS2040 students at the end of the semester. One of the components of the grades is the Visualgo Quiz, of which there are a **total of 4 quizzes**. The final score for each student is the sum of the scores of their **top 3** quizzes.

Mr. Panda has a list of **N** student names with the scores for each of the 4 Visualgo Quizzes. He wants to sort the students in decreasing order of their final score, breaking ties by sorting in increasing lexicographical order of their names. Can you help him do so?

Instead of sorting the list of students directly, Mr. Panda advises that computing each student's final score before sorting them would be easier.

Input
The first line of input contains an integer **N**. The next **N** lines each contain the name of a student followed by 4 integers representing their score for the 4 Visualgo Quizzes.

Output
Output the students in decreasing order of their final score, breaking ties by sorting in increasing lexicographical order of their names. For each student, print the name of the student followed by their final score.

Limits

- $1 \leq \mathbf{N} \leq 50{,}000$
- The names of students will only contain lowercase and uppercase English alphabets. It will also not be longer than 20 characters. However, there might be more than one student with the same name.
- All the scores will range from 0 to 10^8 inclusive.

Sample Input (**grades1.in**)	Sample Output (**grades1.out**)
4 Rar 7 8 7 8 Shark 8 10 3 6 Zip 7 9 0 9 Panda 6 7 8 9	Zip 25 Panda 24 Shark 24 Rar 23

Explanation of Sample Testcase 1
Rar's total score = 8 + 8 + 7 = 23.
Shark's total score = 10 + 8 + 6 = 24.
Zip's total score = 9 + 9 + 7 = 25.
Panda's total score = 9 + 8 + 7 = 24.

Notes:

1. You should develop your program in the subdirectory **ex2** and use the skeleton java file provided. You should not create a new file or rename the file provided.
2. You are free to define your own helper methods and classes (or remove existing ones).
3. Please be reminded that the marking scheme is:
   a. Public Test Cases (1%) - 1% for passing **all** test cases, 0% otherwise
   b. Hidden Test Cases (1%) - Partial scoring depending on test cases passed
   c. Manual Grading (1%)
      i. Overall Correctness (correctness of algorithm, severity of bugs)
      ii. Coding Style (meaningful comments, modularity, proper indentation, meaningful method and variable names)
4. Your program will be tested with a time limit of not less than **2 sec** on Codecrunch.

Skeleton File – Grades.java

You are given the below skeleton file `Grades.java`. You should see a non-empty file when you open the skeleton file. Otherwise, you might be in the wrong working directory.

```
/**
 * Name         :
 * Matric. No   :
 * PLab Acct.   :
 */

import java.util.*;

public class Grades {
    private void run() {
        //implement your "main" method here
    }

    public static void main(String[] args) {
        Grades newGrades= new Grades();
        newGrades.run();
    }
}
```