CS2810 OOAIA: Mid-Sem Exam

Code Deadline: March 2 at 16:45 on Hackerrank

Hackerrank Link: www.hackerrank.com/midsem-1

Problem Statement

Create a class in C++ to represent Students(similar to the previous assignment).
The class should have the following attributes:

- Marks of 5 subjects (No two students will have the same set of scores in all five subjects)
- Skill level(Two students can have the same skill level)

The condition for comparing arbitrary students A and B is:

- Rank(A) < Rank(B) if $M_1(A)>M_1(B)$
- Rank(A) < Rank(B) if $M_i(A)==M_i(B)$ and $M_{i+1}(A)>M_{i+1}(B)$ for i=1 to 4 where $M_i(A)$ denotes marks of student A in subject i

Rank is defined as the position of a student in the sorted order using the above condition. It is indexed from 1.

Let the skill levels of student A and student B ($A \neq B$) be S_A and S_B, and let their ranks be R_A and R_B. (A, B) is a special pair if A's skill level is higher than B's skill level, but A has a greater(worse) rank than B, i.e. (A, B) is a special pair if $S_A > S_B$ and $R_A > R_B$

Count the total number of special pairs.

(Hint: Merge-Sort)

Input Format

The first line contains the total number of students n.
Each of the following n lines contains 6 space-separated integers: marks of 5 subjects and skill level.

n // Number of total students
n lines in the following format:
<Marks 1><space><Marks 2><space><Marks 3><space><Marks 4><space><Marks 5><space><Skill level>

Output Format

A single integer denoting the number of special pairs.

Constraints

$1 \leq$ Number of students $\leq 10^5$
$1 \leq$ Marks in any subject $\leq 10^3$
$1 \leq$ Skill level $\leq 10^9$

Subtasks

10 marks:
$1 \leq$ Number of students ≤ 5000
$1 \leq$ Marks in any subject $\leq 10^3$
$1 \leq$ Skill level $\leq 10^9$

5 marks:
$1 \leq$ Number of students $\leq 10^5$
$1 \leq$ Marks in any subject $\leq 10^3$
$1 \leq$ Skill level $\leq 10^2$

10 marks:
$1 \leq$ Number of students $\leq 10^5$
$1 \leq$ Marks in any subject $\leq 10^3$
$1 \leq$ Skill level $\leq 10^9$

Sample Testcase

Input:
5
1 2 3 4 5 5
2 3 4 5 1 4
3 4 5 1 2 3
4 5 1 2 3 2
5 1 2 3 4 1
Output:
10