Roy is trying to develop a widget that shows Trending Topics (similar to Facebook) on the home page of [HackerEarth Academy](http://learn.hackerearth.com/).   
He has gathered a list of **N** Topics (their **ID**s) and their popularity score (say **z-score**) from the database. Now z-score change everyday according to the following rules:

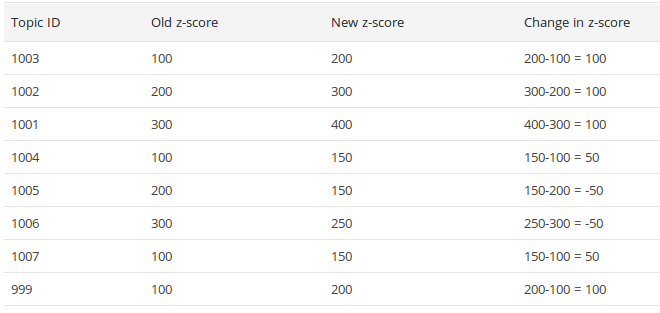
1. When a topic is mentioned in a '**Post**', its z-score is increased by **50**.
2. A '**Like**' on such a Post, increases the z-score by **5**.
3. A '**Comment**' increases z-score by **10**.
4. A '**Share**' causes an increment of **20**.

Now the Trending Topics are decided according to the change in z-score. One with the highest increment comes on top and list follows.  
Roy seeks your help to write an algorithm to find the top **5** Trending Topics.  
If change in z-score for any two topics is same, then rank them according to their ID (one with higher ID gets priority). It is guaranteed that IDs will be unique.

**Input format:**  
First line contains integer **N**  
**N** lines follow  
Each contains **6** space separated numbers representing Topic **ID**, current z-score - **Z**, Posts - **P**, Likes - **L**, Comments - **C**, Shares - **S**

**Output format:**  
Print top **5** Topics each in a new line.  
Each line should contain two space separated integers, Topic **ID** and **new z-score** of the topic.

**Constraints:**  
1 ≤ **N** ≤ 106  
1 ≤ **ID** ≤ 109  
0 ≤ **Z, P, L, C, S** ≤ 109

**Sample Test Explanation:**  


Now sort them according to the change in z-score. Change in z-score for IDs 999, 1001, 1002, 1003 is 100, so sort them according to their ID (one with the higher ID gets priority). Similarly proceed further.

After you get the top 5 list of Topic IDs according to the above criteria, find the new z-score from the table for each ID.

**SAMPLE INPUT**

8

1003 100 4 0 0 0

1002 200 6 0 0 0

1001 300 8 0 0 0

1004 100 3 0 0 0

1005 200 3 0 0 0

1006 300 5 0 0 0

1007 100 3 0 0 0

999 100 4 0 0 0

**SAMPLE OUTPUT**

1003 200

1002 300

1001 400

999 200

1007 150

**Time Limit:**3.0 sec(s) for each input file.