



# 3869 - The Polyathlon

North America - Southeast - 2007/2008

A decathlon is a sporting event with ten different events, and their results are combined to find an overall winner. In a polyathlon, the contestants participate in a variable number of events. You will write code to determine the order of finish of a group of participants in such a multievent competition.

There are two types of events: timed events and measured events.

- Timed events will have the contestant scores recorded in one of three forms:  $hh : mm : ss$  or  $h : mm : ss$  or  $mm : ss$ , where  $h$  is a 1-digit integer, and  $hh$ ,  $mm$ , and  $ss$  are 2-digit integers (representing hours, minutes, and seconds), with  $0 \leq mm, ss \leq 59$ . Example times are 12:01:01, 3:24:05 and 02:27.

Example timed events are a marathon, or time to find a hidden treasure. In a timed event, a lower time is better.

- Measured events will have the contestant scores recorded as integers. Example measured events are the distance a discus is thrown, or the number of hot dogs eaten in 5 minutes. In a measured event, a higher score is better.

After each event, the contestants are placed in order of their finish. The one with the best score is in position 1, the second best score in position 2, etc. In the case of ties, contestants with the same score are all in the same position. But if  $n$  contestants are in position  $p$ , the next contestant will be in position  $p + n$ . For example, consider the contestant results and positions in a measured event below:

Since two participants have the second highest score, 83, they are both in position 2. The third highest score is then in position 4. Participants with equal positions are ordered by alphabetically increasing order of their name.

Once all events are completed, the positions of each contestant in each event are summed. The final results will put the contestants in order from lowest to highest based on the sum of their positions. The lowest sum is in position 1. Ties in final results are handled in the same way as in events. If a contestant does not have a score in all events, they are omitted from the final results.

## Input

The input to your program will be 1 or more data sets. Each data set will begin with two lines. The first line of the data set will be a line of one or more characters in the set  $\{A-Z, 0-9, \_ \}$ , giving the name of the competition. The next line will be a positive integer,  $n$ , being the number of events in the competition. There will then be  $n$  sets of data about each event.

The data about each event will begin with 3 lines describing the event. The first line of the description will be one or more characters in the set  $\{A-Z, 0-9, \_ \}$ , giving the name of the event. The second line will be either "TIMED" or "MEASURED", telling whether the event is timed or measured. The third line will be a positive integer,  $p$ , giving the number of participant scores in the event. There will then be  $p$  lines, each giving a participant name and score.

Each of the participant lines will have a participant name consisting of one or more uppercase characters, a space, and the participant's score. The score will be in the format described in the problem description.

End of input is indicated by end of file.

## Output

For each input set, have a line with the name of the competition, a blank line, and then the output for each of the events in the competition, in the order they appear in the competition followed by the final results.

For each event, the output should be the name of the event on a single line, followed by a single line for each competitor giving their position, their score (output in exactly the same format they were input), and their name, in order of their position. If two competitors finish in the same position, they should be listed alphabetically. There should be a blank line after the output for each event.

After all the output for events, there should be a line saying ``OVERALL\_RANKINGS'', followed by a single line for each competitor giving their position, the sum of their other positions, and their name. Again, these should be in position order, with ties listed alphabetically. You may assume there will be at least one competitor who participates in all events. There should be two blank lines after the output for each competition.

## Sample Input

```
ACM_CONTEST_PREP
2
PROBLEMS_SOLVED
MEASURED
3
DIJKSTRA 3
WIRTH 1
HOARE 3
TIME_TO_SOLVE
TIMED
4
WIRTH 02:12
BOHR 12:01:29
KNUTH 02:12
DIJKSTRA 1:49:05
GREAT_BAKING_CONTEST
1
PIES_BAKED
MEASURED
5
CHERRY 3
APPLE 3
LEMON 3
BLUEBERRY 2
PUMPKIN 4
```

## Sample Output

```
ACM_CONTEST_PREP

PROBLEMS_SOLVED
1 3 DIJKSTRA
1 3 HOARE
3 1 WIRTH
```

TIME\_TO\_SOLVE  
1 02:12 KNUTH  
1 02:12 WIRTH  
3 1:49:05 DIJKSTRA  
4 12:01:29 BOHR

OVERALL\_RANKINGS  
1 4 DIJKSTRA  
1 4 WIRTH

GREAT\_BAKING\_CONTEST

PIES\_BAKED  
1 4 PUMPKIN  
2 3 APPLE  
2 3 CHERRY  
2 3 LEMON  
5 2 BLUEBERRY

OVERALL\_RANKINGS  
1 1 PUMPKIN  
2 2 APPLE  
2 2 CHERRY  
2 2 LEMON  
5 5 BLUEBERRY

---

Southeast 2007-2008