

	UNIVERSIDAD AUTONOMA DE OCCIDENTE			
	FACULTAD DE INGENIERIA NUCLEO MIDIA		ESTRUCTURA DE DATOS Y ALGORITMOS 1	
	CODIGOS:		NOMBRES:	
TAREA FINAL TABLERO DE RESULTADOS				FECHA: 6 de abril de 2024

Profesor: Orlando Arboleda Molina, Msc.

1. Objetivos

El presente proyecto tiene por objeto enfrentar a los estudiantes del curso:

1. A la construcción de aplicativos webs para resolver problemas.
2. Al uso de los algoritmos vistos en clase.
3. Al diseño e implementación de soluciones correctas y eficientes a problemas computacionales, usando los conceptos vistos en clase.

2. Definición de la Tarea

Tomando como referencia el ejercicio que hemos denominado **Tablero de resultados** (mostrado en la Figura 1)¹, construir una solución al ejercicio en el cual se aplicarán las siguientes modificaciones:

- Solo se procesará un único caso.
- La entrada del caso estará compuesta por m submissions. Su formato será el siguiente:
submission₁; submission₂;; submission_m
- Teniendo en cuenta que hay n contest ($1 \leq n \leq 100$), p problems ($1 \leq p \leq 9$), la entrada de cada submission mantiene el formato:

contest problem time L

Donde *contest* es una cadena iniciada con Team y seguida del número correspondiente.

Ejemplo: si la entrada es:

'Team1 1 5 C;Team2 5 10 C;Team3 2 15 I;Team1 3 20 R;Team2 1 25 R;Team2 4 30 C;Team1 3 35 I;Team1 3 40 C;Team3 2 45 I;Team3 2 50 C'

- El scoreboard se calcula aplicando los siguientes criterios:
 - Ordenamiento descendentemente por el número de problemas resueltos (*problemsSolved*) - criterio1.
 - Ordenamiento **ascendentemente** por el tiempo de penalización (*penaltyTime*) - criterio2.
 - Ordenamiento ascendentemente por el *contest* (criterio3).

Por lo cual cada línea del scoreboard debe mantener el siguiente formato:

contest problemsSolved penaltyTime

¹ El nombre real del ejercicio será indicado después de la revisión de las propuestas dadas.

- La solución obligatoriamente debe tener una función en JavaScript denominada ***calcularScoreBoard(caso)*** (escrita en el archivo *util.js*) que recibe el caso y produce una sola cadena con el formato indicado para la salida.

Ejemplo: Si la variable ***caso*** se corresponde a la cadena del ejemplo anterior, la cadena retornada por ***calcularScoreBoard(caso)*** será:

'Team2 2 40

Team1 2 65

Team3 1 90'

Observación: La nota obtenida depende de la correctitud de su algoritmo y eficiencia, con relación a las soluciones de los demás grupos y una solución generada por el docente.

3. Entrega

La entrega consiste en un archivo comprimido en formato zip o rar, que contenga:

- 1- Un documento **Word versión 2013 (o inferior) o pdf**, de máximo 5 páginas, que incluya:
 - Una hoja de portada en la que se indique claramente: los códigos y nombres de los alumnos que hacen la entrega, el grupo de EDyA1 en el que están matriculados.
 - Una hoja con un párrafo de máximo 10 líneas en que se resuma el enunciado dado, el código/pseudocódigo de la función ***calcularScoreBoard(caso)*** y las complejidades individuales y totales de los fragmentos que la constituyen (en la cual deben aparecer los términos *m* para submissions, *n* para contests y *p* para problems).
- 2- Un aplicativo web realizado en JavaScript, sin errores de compilación, con datos iniciales de prueba.

Nota: La implementación **no será evaluada** si **no se suministra el documento** o la función ***calcularScoreBoard(caso)*** no contiene las complejidades individuales y totales solicitadas.

Fecha de Entrega: en la plataforma UAOVirtual, a más tardar a las 8pm del **día 19 de mayo de 2024**.

Sustentación: En la semana 16, en el horario escogido por el grupo, en los mismos horarios de las sesiones de clase. Debido a las limitaciones de tiempo, el horario puede corresponder a un día festivo.

4. Grupos de trabajo

Grupos de 3 personas. Solo se permite una cantidad diferente, si es consultada al docente y este da su aval.

5. Bibliografía Mínima

Los textos guías del curso y enlaces indicados al describir el presente proyecto.

Want to compete in the ACM ICPC? Then you had better know how to keep score! Contestants are ranked first by the number of problems solved (the more the better), then by decreasing amounts of penalty time. If two or more contestants are tied in both problems solved and penalty time, they are displayed in order of increasing team numbers.

A problem is considered solved by a contestant if any of the submissions for that problem was judged correct. Penalty time is computed as the number of minutes it took until the first correct submission for a problem was received, plus 20 minutes for each incorrect submission prior to the correct solution. Unsolved problems incur no time penalties.

Input

The input begins with a single positive integer on a line by itself indicating the number of cases, each described as below. This line is followed by a blank line. There is also a blank line between two consecutive inputs.

The input consists of a snapshot of the judging queue, containing entries from some or all of contestants 1 through 100 solving problems 1 through 9. Each line of input consists of three numbers and a letter in the format *contestant problem time L*, where *L* can be C, I, R, U, or E. These stand for Correct, Incorrect, clarification Request, Unjudged, and Erroneous submission. The last three cases do not affect scoring.

The lines of input appear in the order in which the submissions were received.

Output

The output for each test case will consist of a scoreboard, sorted by the criteria described above. Each line of output will contain a contestant number, the number of problems solved by the contestant and the total time penalty accumulated by the contestant. Since not all contestants are actually participating, only display those contestants who have made a submission.

The output of two consecutive cases will be separated by a blank line.

Sample Input

```
1
1 2 10 I
3 1 11 C
1 2 19 R
1 2 21 C
1 1 25 C
```

Sample Output

```
1 2 66
3 1 11
```

Figura 1. Enunciado original del Tablero de resultados