

7

Problema 34
OTEGUI MARÍN, JOSÉ (TAIS73)

ID envío	Usuario/a	Hora envío	Veredicto
61337	TAIS73	2022-11-02 10:02	AC
61327	TAIS73	2022-11-02 09:56	AC
61302	TAIS73	2022-11-02 09:39	AC

Fichero prob-34.cpp

*
 * Indicad el nombre completo y usuario del juez de quienes habéis hecho esta solución:
 * Estudiante 1: Alberto Chaves TAIS20
 * Estudiante 2: Jose Otegui TAIS73
 *

Para optimizar la diferencia de los resultados hemos ordenado mediante una PriorityQueue de menor a mayor los resultados de los contrarios y de mayor a menor los resultados de los Broncos, de esta forma, compara la mayor puntuación de los Broncos y la menor de los contrarios haciendo que la diferencia de estos resultados sea la máxima posible en caso de ser mayor el resultado de los broncos que de los contrarios ($b > c$).

Para la lectura de datos habrá que crear las estructuras ordenadas, cuya creación tiene un coste de $O(\log n)$ por cada inserción al insertar cada elemento, que se repite n veces, por lo que el coste de creación de las estructuras es $O(n \log n)$

20 30 40 80
 50 40 30 30
 30 10 0 0 = 40

Falta demostración de la corrección de la estrategia voraz

```
bool resuelveCaso() {
    // leer los datos de la entrada
    int nPartidos;
    cin >> nPartidos;
    if (nPartidos == 0)
        return false;

    PriorityQueue<int> contrarios;
    int aux;
    for (int i = 0; i < nPartidos; i++) {
        cin >> aux; contrarios.push(aux);
    }
}
```

```
PriorityQueue<int, greater<int>> broncos;
for (int i = 0; i < nPartidos; i++) {
    cin >> aux; broncos.push(aux);
}

int sol = 0;
int c, b;
for (int i = 0; i < nPartidos; i++) { // O(N)
    if (broncos.top() > contrarios.top())
        sol += broncos.top() - contrarios.top();

    contrarios.pop(); // O(1)
    broncos.pop(); // O(1)
}
// escribir la solución
cout << sol << "\n";
return true;
}
```

