

Problema 7

SOTILLO SOTILLO, CLARA (TAIS96)

ID envío	Usuario/a	Hora envío	Veredicto
55652	TAIS96	2021-09-21 10:14	AC
55610	TAIS96	2021-09-21 10:00	AC
55561	TAIS96	2021-09-21 09:38	WA

Fichero ejercicio7.cpp

```

*
* Indicad el nombre completo y usuario del juez de quienes habéis hecho esta solución:
* Estudiante 1: Carlos Gómez López TAIS45
* Estudiante 2: Clara Sotillo Sotillo TAIS96
*
*
*

```

El coste de la función en el peor de los casos es :
 $O(\log n * (n + c))$ = al máximo entre $O(n \log n)$ y $O(c \log n)$
donde n es el número de cajas y c el número de clientes

*¿Por qué? Operaciones
de coste logarítmico de las
colas de tamaño n*

Inicializamos al comienzo del programa una `priority_queue` con todas las cajas vacías (tiempo 0)
procesamos los clientes añadiéndole al tiempo del nuevo cliente el tiempo del cliente anterior y asignándole la caja de mayor prioridad
volviendo a meter esta en la cola y quitando la del cliente anterior
finalmente en la salida sumamos uno a la caja que queda libre para que salga el verdadero número de caja

```

struct caja {
    int caja, tiempo;
};

bool operator<(caja const& a, caja const& b) {
    return b.tiempo < a.tiempo || (b.tiempo == a.tiempo && b.caja < a.caja);
}

bool resuelveCaso() {

    // leemos la entrada
    int N, C, aux, cj, t;
    caja a;
    priority_queue<caja> colaC;

    cin >> N >> C;
    if (N == 0)
        return false;

```

```
for (int i = 0; i < N; i++) { $O(n \log n)$ 
    a.caja = i;
    a.tiempo = 0;
    colaC.push(a);
}

for (int i = 0; i < C; i++) { $O(c \log n)$ 

    cin >> aux;
    cj = colaC.top().caja;
    t = colaC.top().tiempo;
    colaC.pop(); $\log n$ 
    a.caja = cj;
    a.tiempo = aux + t;
    colaC.push(a); $\log n$ 
}

cout << colaC.top().caja + 1 << "\n";

return true;
}
```

