$\star$Se tiene una grilla con $m \times n$ posiciones, cada una de las cuales tiene un número entero en $[0, k)$, para un $k \in \mathbb{N}$ dado. Dado un valor objetivo $w \in \mathbb{N}$ y una posición inicial (x_1, y_1), que tiene un valor inicial v_1, queremos determinar la mínima cantidad de movimientos horizontales y verticales que transformen v_1 en w, teniendo en cuenta que el i-ésimo movimiento transforma a v_i por $v_{i+1} = (v_i + z) \bmod k$, donde z es el valor que se encuentra en la casilla de destino del movimiento. Por ejemplo, para la siguiente grilla y $k = 10$, se puede transformar $v_1 = 1$ en $w = 0$ con tres movimientos $1 \to 6 \to 4 \to 9$, aunque la solución óptima es vía el camino $1 \to 3 \to 6$.

1	3	6
6	7	4
4	9	3

Modelar este problema como un problema de grafos que se resuelva usando BFS en $O(kmn)$ tiempo.

Los nodos serán (x_i, y_i, v_i) tal que x, y es la posición en la grilla, $v_i \in [0, k), k \in \mathbb{N}$ es el valor actual.

Suponiendo que $z_{i,j}$ es el valor asociado a su respectiva posición (x_i, y_j) en la grilla.

Las aristas son no pesadas, con nodos (x_i, y_i, v_i) hacia $(x_{i+1}, y_i, (v_i + z) \bmod k)$ y $(x_i, y_{i+1}, (v_i + z) \bmod k)$ con $x \leq n, y \leq m$ (si se va de rango no hay arista)

Para la complejidad, es la misma que en el primer taller, tenemos $2mn - m - n$ chances de ir vertical u horizontal, y nm casillas en total, por lo que esto es $O(nm)$. Nosotros le agregamos además un parámetro k que en el peor caso nos suma k nodos por cada x, y, por lo que es $O(knm)$.

El problema se resuelve haciendo BFS desde x_1, y_1, devolviendo un vector de padres. Luego se itera sobre el vector de padres buscando uno con valor congruente a 0 módulo k, y desde ese reconstruimos hasta v usando el vector de padres.