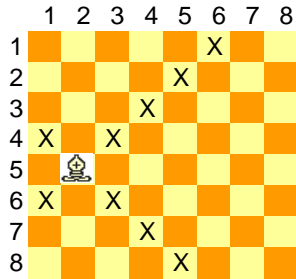


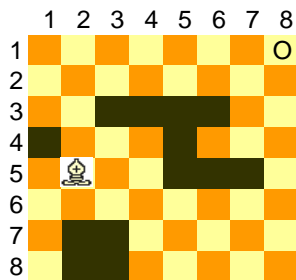
Movimientos de un Alfil

Descripción del problema

Un alfil es una pieza de ajedrez que puede mover desde su casilla actual a cualquier otra casilla ubicada en forma diagonal. Cada movimiento consiste en recorrer una o más casillas. En el dibujo se muestran los 9 movimientos posibles de un alfil en una posición dada.



En nuestro caso, el alfil se mueve en un tablero donde ciertas casillas tienen obstáculos, que impiden su paso; el alfil debe llegar desde su posición inicial hasta una casilla destino (marcada en el gráfico con O).



Se debe escribir un programa **ALFIL** en C, C++ o Pascal que determine si es posible llegar desde la casilla inicial hasta la casilla destino, y en dicho caso, la menor cantidad de movimientos necesarios.

Datos de entrada

Se recibe un archivo **ALFIL.IN** con el siguiente formato:

- Primera línea: el número **n** de cantidad de filas y columnas del tablero ($1 \leq n \leq 1000$)
- **n** líneas, cada una con una fila del tablero, consistente en **n** ceros o unos, separados por blancos. El cero indica que en la casilla no hay obstáculos, el 1 indica que hay un obstáculo.
- Una línea con la fila **x**, columna **y** de la posición inicial del alfil ($1 \leq x, y \leq n$).

- Una línea con la fila **u**, columna **v** de la casilla destino del alfil ($1 \leq u, v \leq n$).
- Las filas y columnas comienzan en el extremo superior izquierdo del tablero.

Datos de salida

El programa debe generar un archivo **ALFIL.OUT** con

- **1** línea con la palabra "SI" si es posible llegar con el alfil hasta la posición destino y "NO" si no es posible
- **1** línea con la cantidad de movidas (si la primera línea es "NO" esta línea debe decir 0)

Ejemplo

En el caso de que la entrada fuera:

ALFIL.IN

```

8
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 1 1 1 1 0 0
1 0 0 0 1 0 0 0
0 0 0 0 1 1 1 0
0 0 0 0 0 0 0 0
0 1 1 0 0 0 0 0
0 1 1 0 0 0 0 0
5 2
1 8
    
```

La salida debería ser:

ALFIL.OUT

```

SI
6
    
```