

3. Diseño de la solución

Decisiones de implementación

- El tablero se representará mediante una matriz
- Cada cuadro almacenará en que posición del viaje ha sido visitado
- Movimientos del caballo

Movimientos:

- Dada una posición inicial (u, v)
- Las posibles nuevas posiciones se obtienen:
$$x = u + \text{Movimientos}[i].X$$
$$y = v + \text{Movimientos}[i].Y$$
 - Si x ó y no están en el intervalo [1, n], el movimiento no es válido
 - Además es necesario que el cuadro (x, y) este a 0 ($Ta[x, y] = 0$)
 - No están disponibles los 8 movimientos posibles en todos los cuadros del tablero

Componentes BackTracking:

- Ensayar(int i, int x, int y, bool &exit, int tablero[MAX][MAX]) es la función donde se va a implementar el BackTracking, y que será usada recursivamente hasta obtener la solución.
- Dentro de la función tenemos:
 1. Un bool Q, que será la que nos indique si hemos resuelto el problema, y una variable k, que proporcionará el nº de movimientos intentados en la misma posición.
 2. Luego, está el bucle que no parará hasta que este la solución o se hayan recorrido 8 movimientos en la misma posición (k).
 3. Una vez dentro, aumentamos k, y comprobamos si puede realizarse un movimiento. Si se puede realizar, volvemos a llamar a la función recursiva, pero si no se encuentra la solución, se pone la posición del tablero a 0. Y así hasta encontrar una solución o que se agoten todas las opciones del tablero (8^8 en este caso).