

## Backtracking: MazeSolver

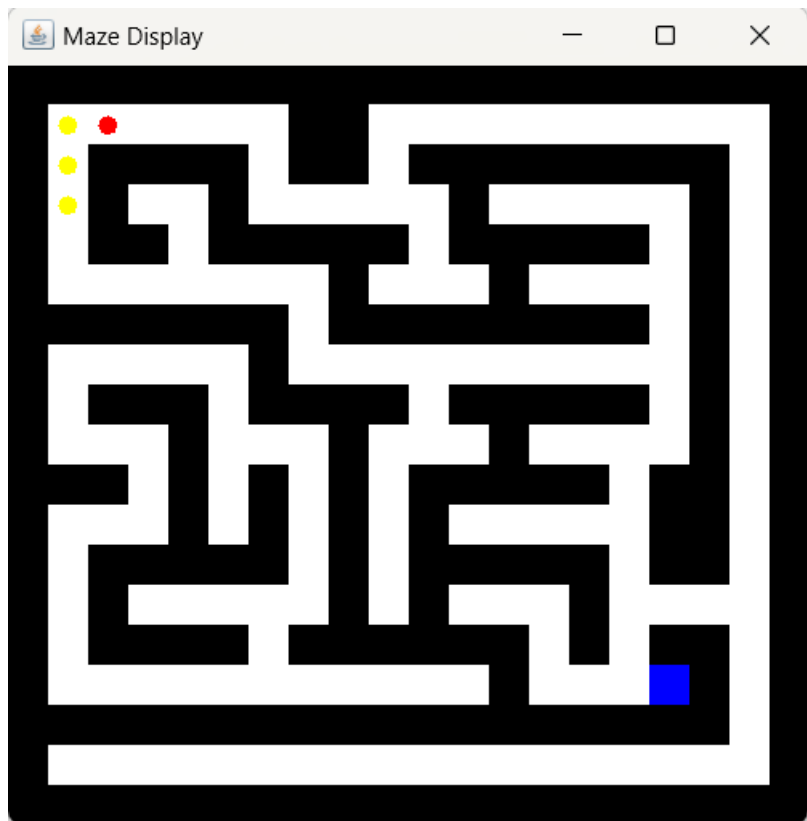
Der rote Punkt ist ein Spieler, der den Weg durchs Labyrinth zum Ziel (=blaues Quadrat) finden soll.

Besuchte Felder werden gelb markiert.

Zu implementieren ist die Methode `findGoal()` der Klasse `MazeSolver.java`

In der Klasse `MazeSolver` gibt es ein Attribut `maze` vom Typ `Maze`.

Im Konstruktor von `MazeSolver` kann man die Startwerte festlegen - und die Wartezeit zwischen den Zügen



Für die Klasse `Maze` gilt das Klassendiagramm unten.

Im Klassendiagramm sind nur die für die Implementierung wesentlichen Methoden aufgeführt.

Maze	
// Attribute spielen keine Rolle	
+ Maze(field: int[][])	
+ isPlayerOnGoal(): boolean	// prüft, ob der Spieler auf dem Ziel steht.
+ isPlayerOnVisited(): boolean	// prüft, ob der Spieler auf einem schon // besuchten Feld steht.
+ isWallInDirection(direction: int): boolean	// prüft, ob in der Richtung eine Wand ist.
+ markPlayerFieldAsVisited(mark: boolean)	// markiert das Feld, auf dem der Spieler ist, als besucht // oder nimmt die Markierung weg.
+ move(direction: int): boolean	// bewegt den Spieler in direction: // 0: oben, 1: rechts; 2: unten; 3: links // wenn die Bewegung auf eine Wand führen würde, // dann passiert nichts und es wird false zurückgegeben. // gibt zurück, ob der Spieler bewegt wurde.
+ moveBack(direction: int): boolean	// bewegt den Spieler ZURÜCK // d.h. für direction: 0 wird der Spieler nach <u>unten</u> bewegt. // gibt zurück, ob der Spieler bewegt wurde.