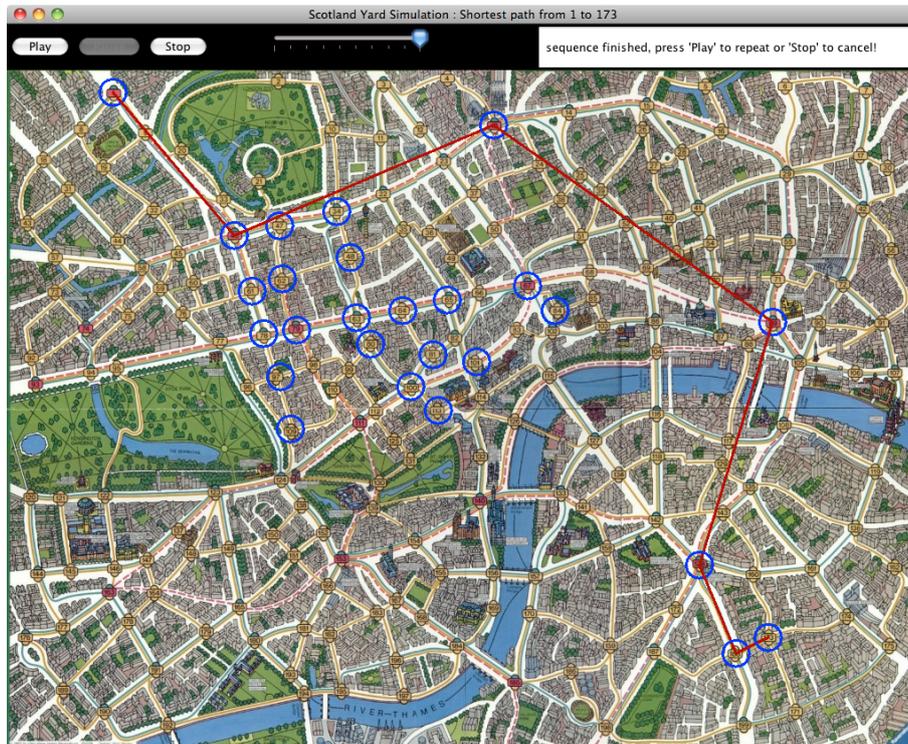


## Aufgabenblatt 3

### Kürzeste Wege in London mit dem Dijkstra- und dem A\*-Verfahren



Der Spielplan des Spiels „Scotland Yard“ (Ravensburger; Spiel des Jahres 1983) besteht aus einer Menge von 199 Knoten (nummeriert von 1 bis 199), die durch Taxi, Bus oder U-Bahn verbunden sind. Auf der Web-Seite finden Sie in einer zip-Datei einen Spielplan als jpg-Datei, eine Textdatei mit allen Kantenverbindungen, eine Textdatei mit x-y-Koordinaten (Pixelkoordinaten) der Knoten und eine Klasse zur Animation des Spielplans mit der Möglichkeit Knoten und Verbindungen farblich zu markieren.

Lösen Sie folgende Teilaufgaben:

1. Implementieren Sie die Klasse **ShortestPath** wie in der Javadoc beschrieben. Achten Sie darauf, dass die Suche abgebrochen wird, sobald der Zielknoten erreicht ist. Testen Sie Ihre Klasse mit dem kleinen Beispielgraphen, der in der Klasse **ExampleGraph** definiert ist.
2. Testen Sie die Klasse **ShortestPath** nun mit dem Scotland-Yard-Spielplan. Dazu ist eine Klasse **ScotlandYard** (siehe Javadoc) zu erstellen. Eine rudimentäre Klasse ist bereits vorhanden. In der main-Methode sind einige Testfälle vorgegeben.
3. Animieren Sie die Suche `searchShortestPath(s,g)` graphisch mit der gegebenen Animationsklasse. Zeichnen Sie die vom Algorithmus besuchten Kandidatenknoten in blau und den gefundenen kürzesten Weg in rot ein (siehe Abb. oben). Beobachten Sie den Unterschied zwischen A\* und dem Dijkstra-Verfahren.
4. Die U-Bahn streikt. Blockieren Sie daher alle U-Bahn-Verbindungen, indem das Gewicht auf einen großen Wert gesetzt wird. Welche kürzesten Wege werden jetzt gefunden?