

# Einführung in die Künstliche Intelligenz

SS17- Prof. Dr. J. Fürnkranz

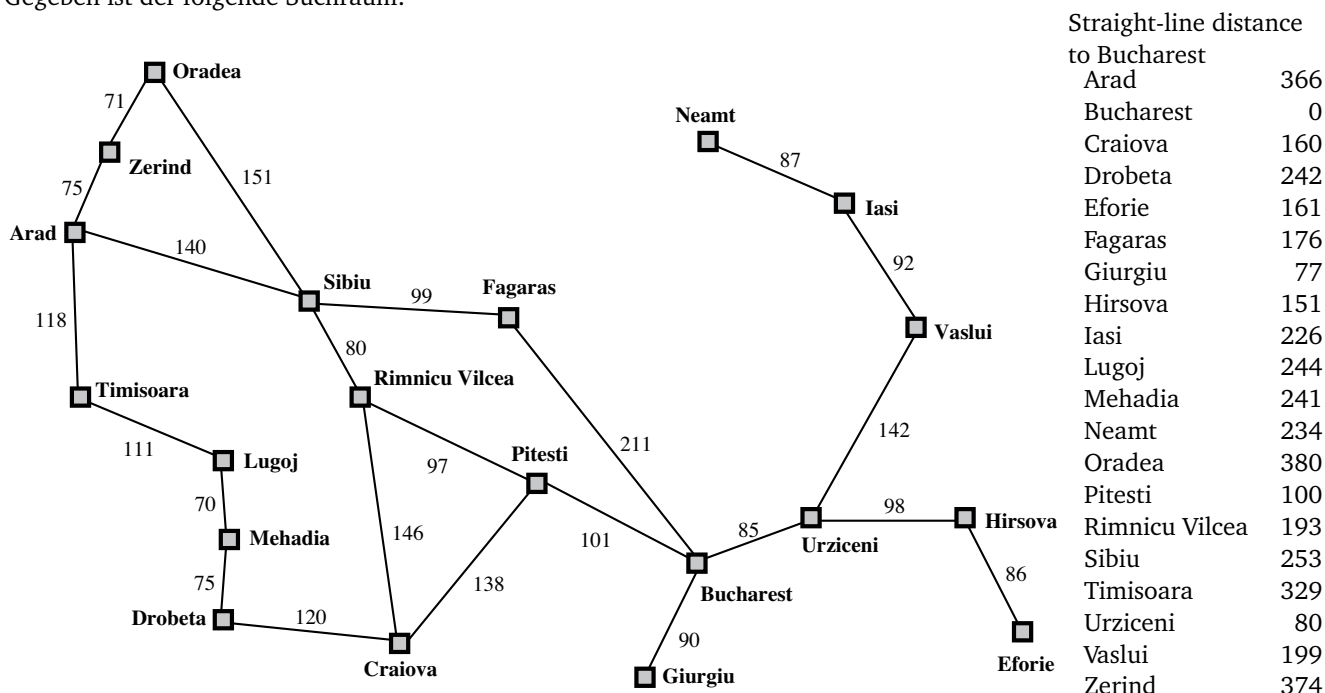


TECHNISCHE  
UNIVERSITÄT  
DARMSTADT

## 2. Übungsblatt

### Aufgabe 1 Informierte Suchalgorithmen

Gegeben ist der folgende Suchraum:



- Gesucht ist die kürzeste Verbindung von Lugoj nach Bucharest. Geben Sie den Suchbaum einer A\*-Suche an.
- Existiert eine Stadt, von der aus Greedy Best-First Search keine Verbindung nach Bucharest findet?  
*Hinweis: Es ist nicht nötig, den Algorithmus auf alle Ausgangsstädte anzuwenden. Viele Städte lassen sich anhand der Grafik ausschließen.*
- Geben Sie die Ausgangsstädte an, für die Greedy Best-First Search keine optimale (hier: kürzeste) Verbindung nach Bucharest findet.  
*Hinweis: Auch hier lassen sich viele Städte ohne (schriftliche) Anwendung des Algorithmus ausschließen.*

### Aufgabe 2 Heuristiken

- Zeigen Sie, dass die Heuristiken  $h_{MIS}$  und  $h_{MAN}$  aus der Vorlesung für das 8-Puzzle (Heuristische Suche, Folie 27) *konsistent* sind (die Kosten für jeden Zug betragen 1).
- Zeigen Sie, dass die Kombination zweier *zulässiger* (admissible) Heuristiken mittels *max* wieder *admissible* ist.

---

### Aufgabe 3 Lokale Suchalgorithmen

---

Geben Sie die Namen der Algorithmen an, die folgende Spezialfälle darstellen:

- a) Beam Search mit einer Beam-Width von  $k = 1$ .
- b) Simulated Annealing mit  $\forall t : T(t) = \epsilon$  für ein beliebig kleines  $\epsilon > 0$ .
- c) Simulated Annealing mit  $\forall t : T(t) = \infty$ .

---

### Aufgabe 4 Constraint Satisfaction Problem

---

Gegeben sei das folgende Cryptarithmic-Problem:

$$\begin{array}{r} \phantom{+} A \\ + \phantom{A} B \\ \hline B \phantom{A} C \end{array}$$

- a) Geben Sie eine Formalisierung des Problems als Constraint Satisfaction Problem an, so dass die Addition für dezimal Zahlen korrekt ist. Das heißt, geben Sie noch die Wertebereiche für die vorkommenden Variablen  $A, B, C, U$  an und spezifizieren sie die Constraints.
- b) Wenden Sie den *Backtracking-Search* Algorithmus auf das Problem an. Nehmen Sie an, dass die Methode *Select-Unassigned-Variable* die Variablen in folgender Reihenfolge zurückgibt:  $C, B, A, U$ . Darüber hinaus liefert die Methode *Order-Domain-Values* die möglichen Werte in aufsteigender Reihenfolge.
- c) Wenden Sie nun zusätzlich *Forward-Checking* an und machen Sie sich klar, an welchen Stellen allgemeines Constraint-Propagation den Suchraum weiter reduziert hätte.