

Name:

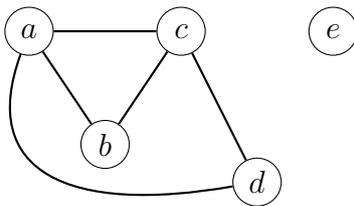
Dauer:

Umfang: Graphen und Tiefensuche, Travelling Salesman Problem, Algorithmen, GGT, Laufzeitkomplexität

Hilfsmittel: keine

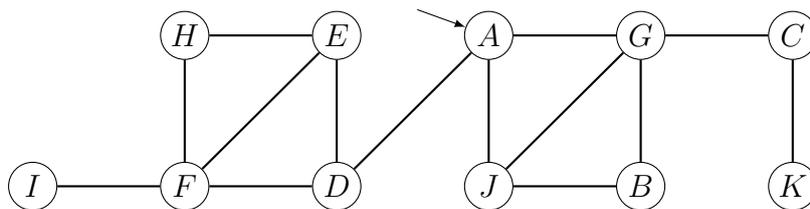
Aufgabe 1

Stelle den Graphen (a) als Adjazenzmatrix (b) als Adjazenzliste dar.



Aufgabe 2

Führe auf dem unten abgebildeten Graphen eine Tiefensuche mit dem Startknoten A durch. Beachte dass die Knoten in einer alphabetisch aufsteigend sortierten Adjazenzlistenstruktur gespeichert sind und die Nachbarknoten somit in alphabetischer Reihenfolge besucht werden.



Aufgabe 3

Gegeben ist die Distanzmatrix eines TSP.

	<i>A</i>	<i>B</i>	<i>C</i>	<i>D</i>
<i>A</i>	0	5	20	3
<i>B</i>	5	0	2	1
<i>C</i>	20	2	0	4
<i>D</i>	3	1	4	0

(a) Berechne die Länge der Rundreise mit der Nearest-Neighbor-Heuristik und dem Startknoten *A*.

(b) Bestimme irgendwie eine Rundreise mit minimaler Länge.

(c) Bestimme irgendwie eine Rundreise mit maximaler Länge.

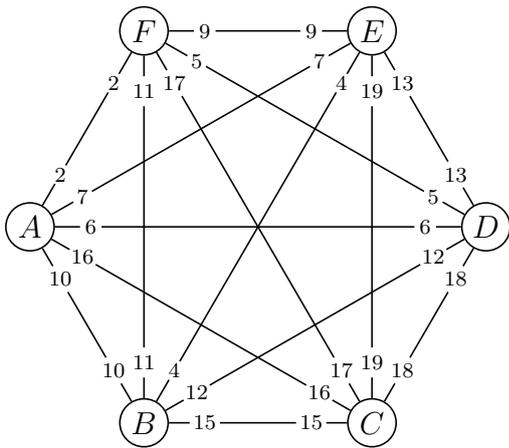
Aufgabe 4

Zeige, dass die folgende Distanzmatrix nicht metrisch ist.

	<i>A</i>	<i>B</i>	<i>C</i>	<i>D</i>
<i>A</i>	0	7	5	2
<i>B</i>	7	0	9	4
<i>C</i>	5	9	0	6
<i>D</i>	2	4	6	0

Aufgabe 5

Löse das TSP mit der MST-Heuristik und dem Startknoten A .



Aufgabe 6

- Beschreibe einen Vorteil der MST-Heuristik gegenüber der Nearest-Neighbor-Heuristik.
- Beschreibe einen Vorteil der Nearest-Neighbor-Heuristik gegenüber der Brute Force-Methode.

Aufgabe 7

- (a) Beschreibe kurz und präzise, was ein *Algorithmus* ist.
- (b) Ist ein Kochrezept ein Algorithmus? Begründe die Antwort.

Aufgabe 8

Zeige schrittweise, wie der klassische euklidische Algorithmus $\text{ggT}(35, 14)$ berechnet.

Aufgabe 9

Zeige schrittweise, wie der moderne euklidische $\text{ggT}(35, 14)$ berechnet.

Aufgabe 10

Bestimme die minimale Laufzeitklasse $O(f(n))$, in welcher der Algorithmus mit der Anzahl elementarer Rechenschritte $T(n)$ liegt, wobei n für die Problemgröße steht.

- (a) $T(n) = 4n^2 + 2n + 5n^3 + 1$
- (b) $T(n) = 5 \cdot 3^{n-1}$
- (c) $T(n) = 27$
- (d) $T(n) = (4n^2 + 3)(5n - 4)(7n^3 - 6)$
- (e) $T(n) = \log_2(4n^2)$