University of Freiburg
Dept. of Computer Science
Prof. Dr. F. Kuhn
P. Bamberger, P. Schneider

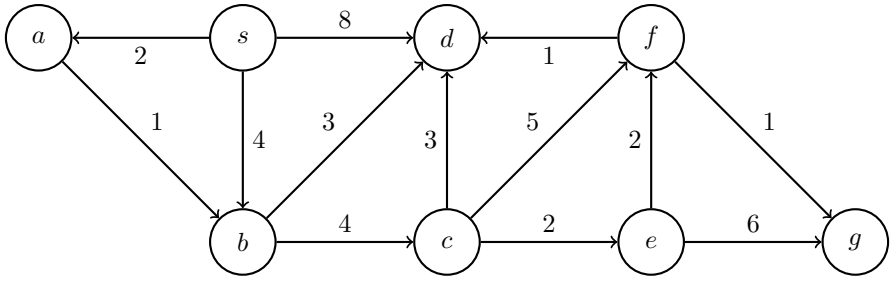
Algorithms and Data Structures
Summer Term 2019
Exercise Sheet 8

Exercise 1: Check for Cycles

Let $G = (V, E)$ be an undirected graph represented by an adjacency list. Describe an algorithm that tests in $\mathcal{O}(|V|)$ steps whether G has a cycle.

Exercise 2: Shortest Paths

Execute **Dijkstras' Algorithm** on the following weighted, directed graph, starting at node s. Write the distances that the algorithm stores in the priority queue after each iteration in the following table.



Initialisation	s	a	b	c	d	e	f	g
$\delta(s, \cdot) =$	0	∞	∞	∞	∞	∞	∞	∞
1. Step $(u = s)$	s	a	b	c	d	e	f	g
$\boldsymbol{\delta(s, \cdot) =}$								
2. Step $(\boldsymbol{u =} \quad)$	s	a	b	c	d	e	f	g
$\boldsymbol{\delta(s, \cdot) =}$								
3. Step $(\boldsymbol{u =} \quad)$	s	a	b	c	d	e	f	g
$\boldsymbol{\delta(s, \cdot) =}$								
4. Step $(\boldsymbol{u =} \quad)$	s	a	b	c	d	e	f	g
$\boldsymbol{\delta(s, \cdot) =}$								
5. Step $(\boldsymbol{u =} \quad)$	s	a	b	c	d	e	f	g
$\boldsymbol{\delta(s, \cdot) =}$								
6. Step $(\boldsymbol{u =} \quad)$	s	a	b	c	d	e	f	g
$\boldsymbol{\delta(s, \cdot) =}$								
7. Step $(\boldsymbol{u =} \quad)$	s	a	b	c	d	e	f	g
$\boldsymbol{\delta(s, \cdot) =}$								
8. Step $(\boldsymbol{u =} \quad)$	s	a	b	c	d	e	f	g
$\boldsymbol{\delta(s, \cdot) =}$								