Communications and Transport Systems
Department of Science and Technology **Fall 2019**
Linköping University

Christiane Schmidt

Design and Analysis of Algorithms Part 1 - Mathematical tools and Network problems homework 2, 14.11.2019

Problem 1 (Trees):

(a) Prove Theorem 1.61 from the lecture.

(b) Prove Theorem 1.62 from the lecture.

(c) Prove Corollary 1.64 from the lecture.

Problem 2 (Directed cycles and directed cuts):
Show:
In a digraph G, each edge belongs either to a (directed) cycle or to a directed cut. Moreover, the following statements are equivalent:

(a) G ist strongly connected.

(b) G contains no directed cut.

(c) G is connected and each edge of G belongs to a cycle.

(Hint: Take a look at the statements you proved in Problem 2.)

Problem 3 (Best-case running time for quicksort):
Proof Lemma 3.12 from the lecture, that is, the best-case running time for quicksort

Problem 4 (Heap Sort):
Prepare a 10 minute presentation of heapsort: the algorithm, its correctness and runnning time.

Problem 5 (Merge sort):

Sort the sequence $(33, 14, 7, 9, 2, 11, 45, 21)$ using merge sort. Give the intermediate steps in appropriate form.

Problem 6 (Mastertheorem):

a) Determine the asymptotic growth of the following recursion using the master theorem

   $U(n) = 4 \cdot U(\frac{n}{3}) + 17 \cdot n^2 + 20 \cdot U(\frac{n}{6})$.

   Determine the value of all parameters used in the master theorem.

b) Determine the asymptotic growth of the following recursion using the master theorem

   $V(n) = 14 \cdot V(\frac{n}{36}) + 23n + 12 \cdot V(\frac{n}{24}) + V(\frac{n}{10})$.

   Determine the value of all parameters used in the master theorem.

c) Determine the asymptotic growth of the following recursion using the master theorem

   $T(n) = 49 \cdot T(\frac{n}{7}) + 42n$.

   Determine the value of all parameters used in the master theorem.

Problem 7 (Quicksort):

Sort the numbers in the following array using the algorithm quicksort presented in the lecture.

$$A[1] = 14 \qquad A[2] = 3 \qquad A[3] = 7 \qquad A[4] = 1 \qquad A[5] = 2$$

The reference element should be chosen as in the lecture (that is, $A[r]$). Give the array after **each** swap operation. Give the intermediate steps from Quicksort- and Partition calls.

Problem 8 (The Kevin Bacon oracle):
The *Kevn Bacon oracle* is based on the actor graph G: actors are given as vertices. Two actor vertices are connected by an edge if they appeared in a movie together. The vertex of Kevin Bacon has value 0; the *Kevin-Bacon number* (KBN) of another actor is the length of a shortest path in G. (Tom Hanks played with Kevin Bacon in Apollo 13, thus, he has Kevin-Bacon number 1.)
The oracle is available here: `http://oracleofbacon.org/`. The movie data it is based on is taken from the *Internet Movie Database*: `http://www.imdb.com`.
Our questions:

(a) Describe a strategy to definitely find an actor with a KBN as high as possible in G, even if you've never heard of Hollywood. On which graph algorithm is this strategy based?

(b) Find a vertex with KBN at least 4.

Problem 9 (Eulerian Path):

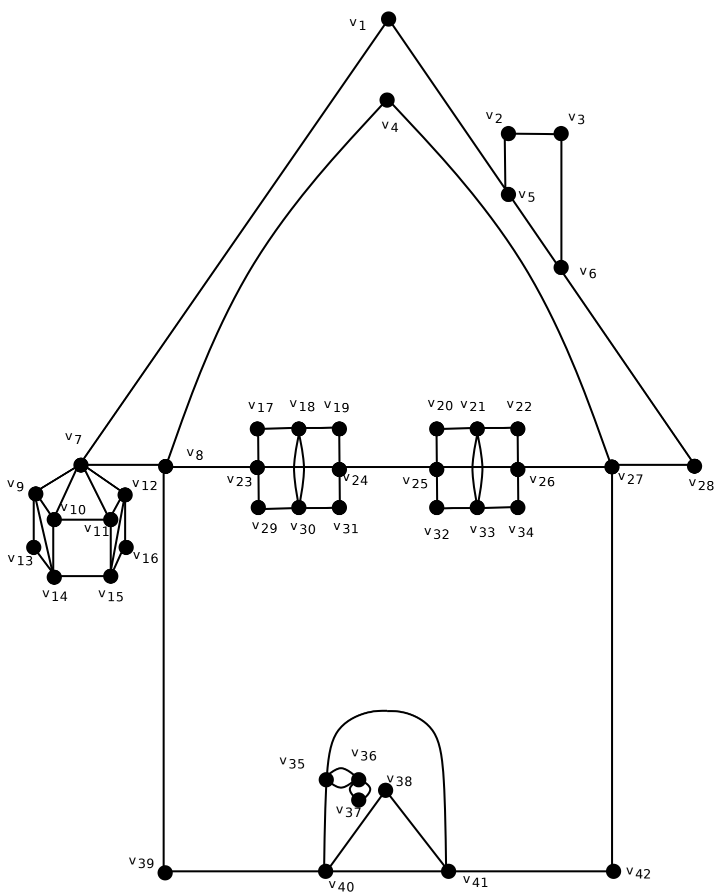


Abbildung 1: Euler on his way home!

Find a Eulerian path in the graph from Figure 1 or show that none exists.

Problem 10 (BFS and DFS):

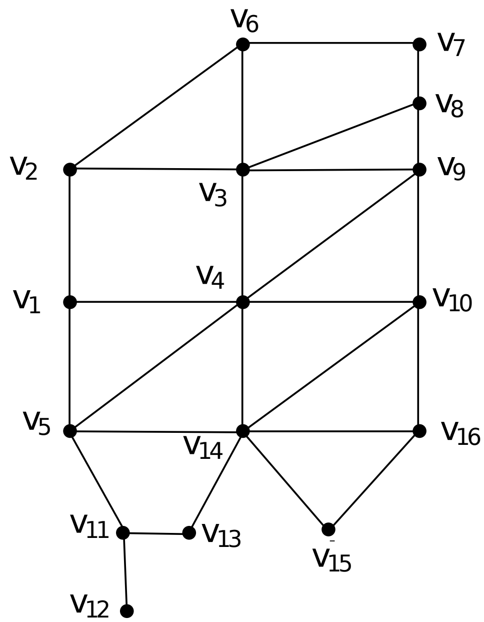


Abbildung 2: The graph G.

a) Apply BFS with start vertex v_1 to graph G from Figure 2.

b) Apply DFS with start vertex v_1 to graph G from Figure 2.

c) Give the adjacency list for G.

(Ad a) and b): If at any time there is more than one vertex to choose from, use the one with the smallest index.)

Problem 11 (BFS and DFS in trees):
Construct an algorithm that determines whether an arbitrary given graph G=(V,E) is a tree based on

(a) DFS

(b) BFS

Problem 12 (Trees and Leaves):

Show that (also during winter) each (undirected) tree has a leaf. (Hint: In an undirected tree a leaf is defined as a vertex of degree 1.)

(10 points)

Problem 13 (BFS):

Let $G = (V, E)$ be a graph and $s \in V$ a vertex; for an arbitrary vertex $x \in V$ let $d(s, x)$ denote the length of a shortest path from s to x. Let $e = \{u, v\} \in E$ be an edge.

a) Prove: $d(s, v) \leq d(s, u) + 1$.

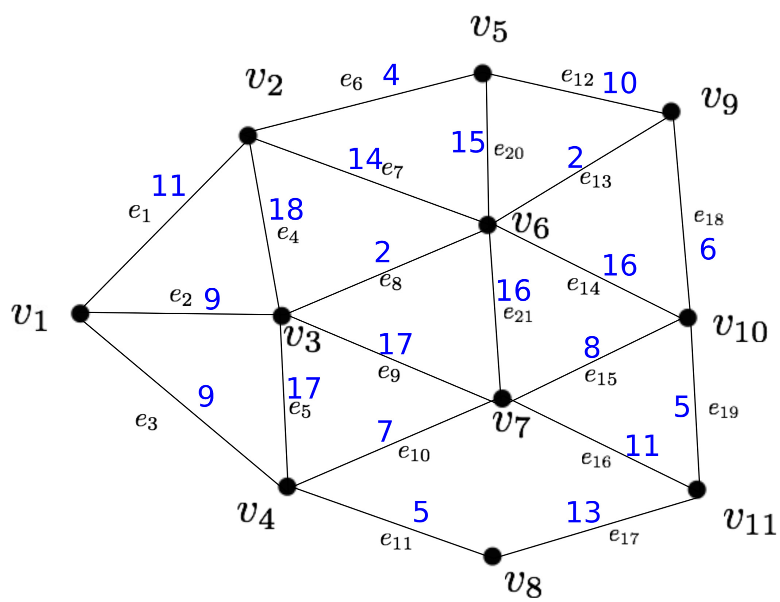
b) Prove or disprove: $d(s, u) \leq d(s, v) + 1$.

c) Does $d(s, v) = d(s, u) + 1$ oder $d(s, u) = d(s, v) + 1$ always hold?

Problem 14 (Forests and Connected Components):
Show: Given a forest with n vertices, m edges and p connected components, then $n = m + p$ holds.

(8 points)

Problem 15 (Kruskal):



Determine an MST using Kruskal's algorithm. Give the edges in the order in which they are included to the tree, and draw the resulting solution to the problem. Tie breaking: if in any step several edges could be chosen, choose the one with the smallest edge index.

We assume that we write edges as $e = (v_i, v_j)$ with $i < j$.

IMPORTANT: To obtain a runtime of $O(m \log n)$, the data structure presented in the seminar can be used. Give the state of the data structure after each edge insertion.

(Note: if there is more than one possibility to add an edge, choose the edge that runs from the vertex with lower index to the vertex with higher index.)