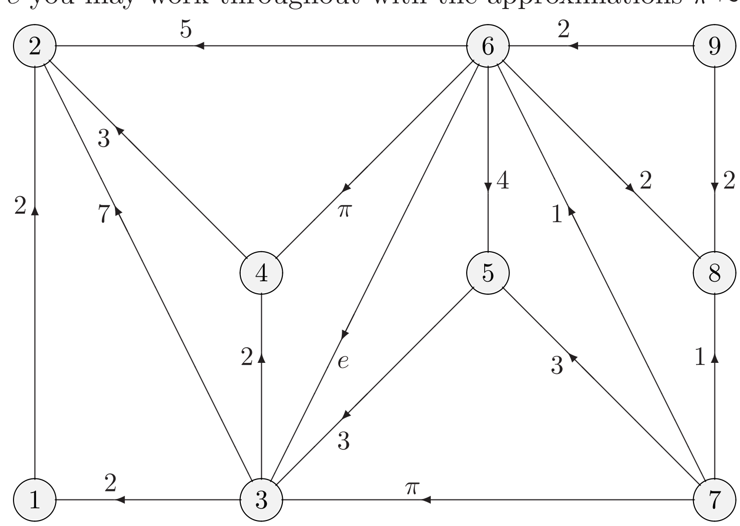
Tentamen Optimalisering in Netwerken (2WO12) Donderdag, 14 April 2011

This tentamen consists of 5 problems on three pages. The tentamen has to be done with pencil and paper alone: **You are not allowed to use your books or notes.** You may formulate your answers in Dutch or in English.

Altogether, you can reach 100 points. Your grade is your total number of points divided by 9.99, and then rounded to the nearest positive integer.

Problem 1. Consider the following arc-weighted, directed graph with $n = 9$ vertices. For the arc-weights π and e you may work throughout with the approximations $\pi \approx 3.1$ and $e \approx 2.7$.



(a) [10 points] Is this directed graph acyclic? If yes, then exhibit a topological ordering. If no, then exhibit a directed circuit. **Report** every step of your computations.

For the following two questions (b) and (c), you will **ignore the orientations** of the arcs and consider the underlying undirected graph (with the old arc-weights as new edge-weights).

(b) [10 points] Use the **Dijkstra** algorithm to compute shortest paths from vertex 1 to all other vertices. **Report** every step of your computation.

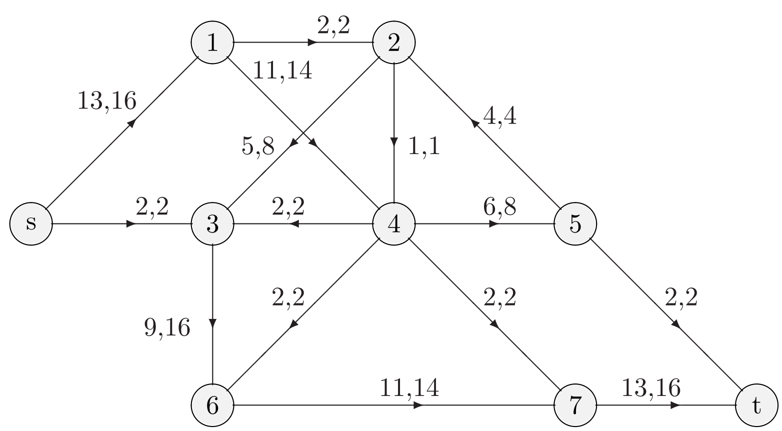
(c) [10 points] Use the **Prim-Dijkstra** algorithm to compute a minimum spanning tree for this undirected graph. Use vertex 7 as your starting vertex. **Report** every step of your computation.

Problem 2. Consider the following network with 9 nodes s, t, and $1, 2, 3, 4, 5, 6, 7$. For every arc the first number denotes the current flow on this arc, and the second number denotes the capacity of this arc. For instance, the arc $s \to 1$ has a current flow of 13 and a capacity of 16.

(a) [5 points] Determine the corresponding auxiliary graph D_f for the given current flow!

(b) [10 points] Starting from the given current flow, compute a maximum flow from s to t with the algorithm discussed in the course. **Report** every step of your computation! **Report** the value of the maximum flow!

(c) [5 points] Determine a minimum s-t-cut for this network! **Compute** the capacity of this cut, and justify why your cut is a **minimum** s-t-cut!



Problem 3. Indicate for each of the following 10 statements whether it is true (T) or false (F). There is **no need to justify** your answers; just write T or F. For each good/wrong/empty answer you obtain 2/-2/0 points. In the following every graph is an undirected, simple graph (without parallel edges and without loops).

(a) Every graph contains an even number of vertices with odd degree.

(b) Every tree on 100 vertices contains 40 vertices that have the same degree.

(c) There exists a non-connected graph with 10 vertices and 37 edges.

(d) There exists a non-connected planar graph with 10 vertices and 22 edges.

(e) If a graph is non-connected, then also its line-graph is non-connected.

(f) If a graph is regular, then also its line-graph is regular.

(g) Every $K_{n,n}$ with $n \geq 10$ has a Hamiltonian cycle.

(h) The edge-chromatic number of every planar graph is at most 8.

(i) Every graph with chromatic number 5 contains a triangle.

(j) The vertex-chromatic number of every graph $K_{n,m}$ is at most 3.

Problem 4. Consider a simple, undirected, unweighted, bipartite graph $G = (V, E)$ with bipartition $V = L \cup R$. The lefthand vertex set L in the bipartition contains $2n$ vertices. The righthand vertex set R in the bipartition contains $4n$ vertices that are partitioned into n disjoint groups $R_1, \ldots, R_n$ that each contain four vertices. All edges in E are between vertices in L and vertices in R. A subset $M \subseteq E$ of the edges is called an *Eindhoven-set*, if

(i) every vertex in L is incident to exactly one edge in M,

(ii) every vertex in R is incident to at most one edge in M, and

(iii) in every group R_i with $1 \leq i \leq n$ at most three out of its four vertices are incident to an edge in M.

[20 points] **Design** a polynomial time algorithm for deciding whether such a bipartite graph contains an Eindhoven-set. **Prove** the correctness of your algorithm, and **analyze** its time complexity. (Hint: Model the problem in terms of one of the optimization problems that you have met in the course.)

Problem 5. In a graph $G = (V, E)$ a subset $X \subseteq V$ is said to *induce a matching*, if in the subgraph of G induced by X every vertex has degree one. Consider the following decision problem in undirected graphs:

PROBLEM: Large Induced Matching

INSTANCE: An undirected graph $G = (V, E)$; a bound k.

QUESTION: Does V contain a subset X with $|X| \geq k$ that induces a matching?

[10 points] **Prove** that the *Large Induced Matching* problem is NP-hard.

Summary of points:

Problem 1: (a) 10; (b) 10; (c) 10
Problem 2: (a) 5; (b) 10; (c) 5
Problem 3: 20
Problem 4: 20
Problem 5: 10