TU Berlin, March 6, 2018

Name: .... **Matr.-Nr.:** ....

Advanced Algorithmics Exam

(Bredereck/Nichterlein/Bentert/Niedermeier, Winter Term 2017/2018)

Exercise No.:	1	2	3	4	5	6	7	8	**Sum**
Points:	20	20	20	16	16	16	20	12	140
Achieved:									

Time limit: 120 Minutes

Max. number of points: 140 Points
Best grade: $\geq$ 86 Points

General hints:

- You are not allowed to use any technical aids or learning material during the exam.
- Do not use a pencil. Use a black or blue pen (non-erasable).
- Write your name and matriculation number on each sheet.
- **If not explicitly excluded in the task, then all answers have to be justified! Answers without justification receive 0 points.**

We wish you success!

Name: .... **Matr.-Nr.:** ....

Task 1: **Integer Linear Programming** (20 Points)

Provide integer linear programming (ILP) formulations for the two following problems (without justification).

To this end, define the used variables, all constraints, and the objective function (maximization or minimization). Use only a polynomial number of constraints and variables!

(a) **2-Club** (10 P)

Input: An undirected graph $G = (V, E)$.

Task: Find the largest 2-club in G, that is, find the largest vertex subset $V' \subseteq V$ such that every two distinct vertices in V' have distance at most 2 in $G[V']$.

Hint:

$G[V']$ is the subgraph induced by V', formally, $G[V'] := (V', E \cap \{\{u, v\} \mid u, v \in V'\})$.

(b) **Edge-Disjoint** s-t–**Paths** (10 P)

Input: A directed graph $D = (V, A)$ and two vertices $s, t \in V$.

Task: Find the largest number of edge disjoint s-t–paths in D.

Note: Use the following notation for the vertices: $V = \{v_1, \ldots, v_n\}$, $s = v_1$, and $t = v_n$.

Name: .... **Matr.-Nr.: ....**

Task 2: **Color Coding** (20 Points)

Consider the following problem:

DISJOINT K_4S
Input: An undirected graph $G = (V, E)$ and an integer k.
Question: Are there k vertex-disjoint K_4s in G?

Note: A K_4 is a graph with four vertices and six edges, that is, all vertices are pairwise adjacent.

Show that DISJOINT K_4S admits a randomized algorithm that, in $f(k) \cdot |V|^{\mathcal{O}(1)}$ time with some constant probability, finds k vertex-disjoint K_4s if they exist.

Hint: Recall that $(1-x)^t \leq e^{-t \cdot x}$ for $0 \leq x \leq 1$. Try to construct a $k^{O(k)}$-time algorithm.

Name: .... **Matr.-Nr.:** ....

Task 3: **Geometry** (20 Points)

Consider the following problem from computational geometry stated as decision problem:

Points in General Position
Input: A set of points $P \subseteq \mathbb{Q}^2$ and $k \in \mathbb{N}$.
Question: Is there a subset $S \subseteq P$ in general position of cardinality at least k?

Hint: A set $Q \in \mathbb{Q}^2$ of points is in general position if no three points of Q are on a straight line.

(a) Prove the correctness of the following data reduction rule, that is, prove that the input instance is a yes-instance if and only if the produced instance is a yes-instance: (12 P)

Rule. Let (P, k) be an instance of Points in General Position. If there is a line L that contains at least $\binom{k}{2} + 2$ points from P, then remove all the points on L and set $k := k - 2$.

Hint: Let L be a straight line with at least $\binom{k}{2} + 2$ points. Intuitively, the rule says that two points of L can be added to a solution S with $k-2$ points in general position where S does not contain any point on L so far.

(b) Provide a polynomial-time factor-3 approximation algorithm for the optimization variant of Points in General Position, where as few points as possible should be deleted from the input. In other words, given $P \subseteq \mathbb{Q}^2$, your algorithm should delete points from P in polynomial time such that (8 P)

- the remaining points are in general position and
- your algorithm deletes at most 3-times as many points as an optimal solution.

Name: .... **Matr.-Nr.: ....**

Task 4: **Monte Carlo vs. Las Vegas Algorithms** (16 Points)

Let P be a decision problem and let $\mathcal{A}$ be a randomized algorithm for P that always gives the correct answer (to the word problem $x \in P$) but finishes within n^2 steps only with probability 1/2. Prove that there is a randomized algorithm $\mathcal{B}$ that (always) finishes within $O(n^2)$ steps but gives the answer to $x \in P$ with probability at least 7/8.

Hint: Assume that in each application the running time of $\mathcal{A}$ is based on a fair coin flip.

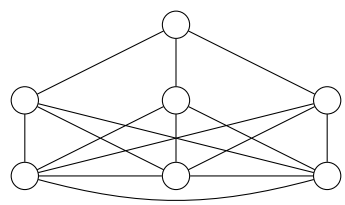
Name: .... **Matr.-Nr.:** ....

Task 5: **Constant-Factor Approximation** (16 Points)

Consider the following optimization problem:

VERTEX DELETION TO CLIQUE
Input: An undirected graph $G = (V, E)$.
Task: Compute a smallest vertex subset $V' \subseteq V$ such that deleting all vertices in V' transforms G into a single clique?

(a) Mark one optimal solution (the set V') for VERTEX DELETION TO CLIQUE in the graph below (without justification). (6 P)



(b) Design a polynomial-time 2-approximation algorithm for VERTEX DELETION TO CLIQUE. (10 P)

Name: .... **Matr.-Nr.: ....**

Task 6: **Distributed Algorithms: Determine the Twins** (16 Points)

Let $G = (V, E)$ be a synchronous ring, where $|V| = n = |E|$, that is, $V = \{v_1, v_2, \ldots, v_n\}$ and $E = \{\{v_i, v_{i+1}\} \mid 1 \leq i < n\} \cup \{\{v_n, v_1\}\}$ and all nodes V use the same clock. In each step each node $v \in V$ can perform an arbitrary amount of computations, store an arbitrary amount of information, and send one message to each neighbor. Assume that the nodes can distinguish between their neighbors (that is, if a node receives a message, then the node knows which neighbor sent it) but the nodes do not know which neighbor sits clockwise and which sits counterclockwise. In the beginning each node $v \in V$ knows:

- its identifier $\mathrm{id}(v)$ and
- that two nodes have the same identifier, but
- they do **not** know how many nodes are in the ring.

Write a synchronous distributed algorithm with a message complexity polynomial in n that determines which identifier is **not** unique.

Name: .... **Matr.-Nr.: ....**

Task 7: **Computational Social Choice: Adjusted Winner** (20 Points)

Recall the protocol from the lecture for allocating goods to two agents which can be formulated as follows:

Adjusted Winner:

1. Each agent distributes 100 points to a set of goods (only integer points).
2. Give each good to an agent that assigns the maximum number of points to it.
3. As long as one agent is more happy than the other agent, a good with the smallest ratio $\frac{\text{value of happier agent}}{\text{value of other agent}}$ is transferred to the other agent:
   (a) Transfer the full good if the happier agents remains the happier agent, or
   (b) Split the good in a ratio such that both agents become equally happy.
4. Repeat Step 3 until both agents are equally happy.

Note: We measure happiness of an agent by the sum of points it assigns to its goods.

The goods are: car (C), house (H), dog (D), music collection (M), video collection (V).

(a) Assume the points are distributed as follows: (4 P)

	agent 1	agent 2
C	23	22
H	23	22
D	10	10
M	22	23
V	22	23

What is the outcome of the protocol?

(b) Find an alternative distribution of the points such that at least two goods are transferred (fully or split). Avoid that the dog is split/shared. (4 P)

(c) What is the minimum number of points an agent may receive? Construct a worst-case distribution of the points, that is, a distribution that minimizes the happiness of the happiest agent. (6 P)

Hint: To show minimality, observe how much the total happiness may decrease when some good is transferred in Step 3.

(d) Assume that Step 2 would be "Give each good to agent 1." Is this modified protocol still anonymous, that is, does it treat the agents equally in the sense that the outcome remains the same if one exchanges the roles of the agents? (6 P)

Name: .... **Matr.-Nr.: ....**

Task 8: **Quantum Algorithms and Randomized Algorithms** (12 Points)

Describe in **at most 40 words** connections between Quantum Algorithms and Randomized Algorithms.