Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Institute of Theoretical Computer Science
Mohsen Ghaffari, Angelika Steger, David Steurer, Emo Welzl, Peter Widmayer

Algorithms, Probability, and Computing — Final Exam — HS18

Candidate

First name: ....

Last name: ....

Student ID (Legi) Nr.: ....

I attest with my signature that I was able to take the exam under regular conditions and that I have read and understood the general remarks below.

Signature: ....

General remarks and instructions

1. Check your exam documents for completeness (10 one-sided pages with 8 exercises).
2. You have **3 hours** to solve the exercises.
3. You can solve the exercises in any order. You should not be worried if you cannot solve all exercises! Not all points are required to get the best grade.
4. Immediately inform an assistant in case you are not able to take the exam under regular conditions. Later complaints will not be accepted.
5. **Pencils** are not allowed. Pencil-written solutions will not be reviewed.
6. No auxiliary material allowed. All electronic devices must be turned off and are not allowed to be on your desk. We will write the current time on the blackboard every 15 minutes.
7. Attempts to cheat/defraud lead to immediate exclusion from the exam and can have judicial consequences.
8. Provide only one solution to each exercise. Cancel invalid solutions clearly.
9. **All solutions must be understandable and well-founded. Write down the important thoughts in clear sentences and keywords. Unless stated otherwise, no points will be awarded for unfounded or incomprehensible solutions.**
10. You may use anything that has been introduced and proved in the lecture notes without reproving it, except when you are asked explicitly to do so.
11. Write your student-ID (**Legi-number**) on **all** sheets (and your name only on this cover sheet).

Good luck!

	achieved points (maximum)	reviewer's signature
1	(15)	
2	(15)	
3	(15)	
4	(15)	
5	(15)	
6	(15)	
7	(15)	
8	(15)	
Σ	(120)	

Exercise 1: MST and Minimum Cut (15 points)

(a) **(10 points)** Describe a *deterministic linear time* algorithm for finding a minimum spanning tree in a planar graph $G = (V, E)$ with an injective weight function $w : E \to \mathbf{R}$. Justify your answer.

Hint: You may use the fact that by contracting an edge in a planar graph the graph remains planar, without proving it.

(b) **(5 points)** You know that a graph G on n vertices cannot have more than $\binom{n}{2}$ minimum cuts. Prove that this bound is tight, i.e., for any $n \geq 3$ there is an n-vertex graph which has $\binom{n}{2}$ minimum cuts.

Exercise 2: Random Search Trees (15 points)

In a random search tree for n keys, we denote by $W_n^{(i)}$ the random variable for the number of descendants of the key of rank i. Compute $E[W_n^{(n-1)}]$ exactly.

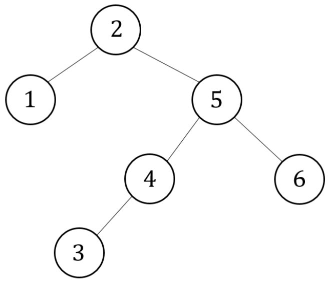

Figure 1: In this example, key 5 has 4 descendants, namely $3, 4, 5$ and 6.

Exercise 3: Point Location (15 points)

Let S be a set of n disjoint segments in the plane. We define P to be the set of $2n$ endpoints of the segments in S. Assume that the set $\{\text{distance}(p, q) : p, q \in P, p \neq q\}$ has exactly $\binom{2n}{2}$ elements.

Let $s_1, s_2, \cdots, s_n$ be a permutation of the segments in S, drawn uniformly at random from the set of all permutations of S. Imagine that the segments are inserted into the plane in this order, one by one. Let P_i be the set of the endpoints of segments $s_1, \cdots, s_i$. The *diameter* of P_i is the maximum distance between two elements of P_i. If the diameter of P_i is different from P_{i-1}, then we call it a diameter change. Here, we count the insertion of s_1 as a diameter change. Let random variable X be the number of diameter changes. Give a function $f(n)$ such that $f(n) \leq E[X] \leq 2f(n)$.

Exercise 4: Linear Programming (15 points)

(a) **(5 points)** Prove that a system $A\mathbf{x} = \mathbf{b}$ of linear equations is unsolvable **if** there is $\mathbf{y}$ with $A^{\mathsf{T}}\mathbf{y} = \mathbf{0}$ and $\mathbf{b}^{\mathsf{T}}\mathbf{y} = 1$.

(b) **(5 points)** A linear program in *standard form* is a linear program of the form

$$\text{maximize } \tilde{\mathbf{c}}^{\mathsf{T}}\tilde{\mathbf{x}} \quad \text{subject to} \quad \tilde{A}\tilde{\mathbf{x}} \leq \tilde{\mathbf{b}} \text{ and } \tilde{\mathbf{x}} \geq 0.$$

Recall that its dual is

$$\text{minimize } \tilde{\mathbf{b}}^{\mathsf{T}}\tilde{\mathbf{y}} \quad \text{subject to} \quad \tilde{A}^{\mathsf{T}}\tilde{\mathbf{y}} \geq \tilde{\mathbf{c}} \text{ and } \tilde{\mathbf{y}} \geq 0.$$

By applying the definition of duality, prove that the dual of

$$\begin{array}{lrcl} \text{maximize} & \mathbf{c}^{\mathsf{T}}\mathbf{x} & & \\ \text{subject to} & A\mathbf{x} & \leq & \mathbf{b}, \end{array} \tag{1}$$

is

$$\begin{array}{lrcl} \text{minimize} & \mathbf{b}^{\mathsf{T}}\mathbf{y} & & \\ \text{subject to} & A^{\mathsf{T}}\mathbf{y} & = & \mathbf{c} \\ & \mathbf{y} & \geq & \mathbf{0}. \end{array} \tag{2}$$

(c) **(5 points)** Let $\mathbf{c} \in \mathbb{R}^n$ be some fixed vector. Furthermore, assume that $\mathbf{x}$ is a vector of n real valued variables. Describe explicitly the set of all basic feasible solutions and find an optimal basic feasible solution to the linear program maximize $\mathbf{c}^{\mathsf{T}}\mathbf{x}$ subject to $\mathbf{1}^{\mathsf{T}}\mathbf{x} = 1$ and $\mathbf{x} \geq \mathbf{0}$.

Exercise 5: Linear Programming and Vertex Cover (15 points)

Assume that you are given a graph $G = (V, E)$, where each vertex $v \in V$ has a weight $w_v > 0$. A set $S \subseteq V$ is a *vertex cover* of G if for each edge in E at least one of its endpoints is in S. We want to approximate the minimum weight of a vertex cover in G, where the weight of a vertex cover is equal to the sum of the weights of its vertices.

(a) **(6 points)** For each vertex $v \in V$ consider a zero-one variable y_v. Prove that the optimal value of the following integer linear program is equal to the minimum weight of a vertex cover in G.

$$\begin{array}{lll} \text{minimize} & \sum_{v \in V} w_v y_v & \\ \text{subject to} & y_v + y_u \geq 1 & \forall \{v, u\} \in E \\ & y_v \in \{0, 1\} & \forall\, v \in V. \end{array} \tag{3}$$

(b) **(9 points)** Relax the integer linear program from (a) and assume that $\mathbf{y}^*$ is an optimal solution to this linear programming relaxation. We define the set S to be $S := \{v : y_v^* \geq \alpha\}$. Select α such that S is a vertex cover and the weight of S is at most $2 \cdot \mathrm{OPT}$, where OPT is the minimum weight of a vertex cover in G. Justify your answer.

Exercise 6: Randomized Algebraic Algorithms (15 points)

(a) **(7 points)** We say that an orientation $\vec{G}$ of a graph G is *Pfaffian* if every nice cycle is oddly oriented. Prove that the complete bipartite graph $K_{3,3}$ does not have a Pfaffian orientation. Furthermore, show that by removing an edge from $K_{3,3}$, the resulting graph has a Pfaffian orientation.

Hint: You may use the fact that the complete bipartite graph $K_{2,3}$ cannot be oriented so that every even cycle is oddly oriented, without proving it.

(b) **(8 points)** Let $G = (V, E)$ be a planar graph with n vertices and $2n$ edges. Prove that G has at most $\sqrt{2}^n$ perfect matchings.

Hint: Recall that for a matrix $A = (a_{ij})_{i,j=1}^n \in \mathbb{R}^{n\times n}$,

$$|\det(A)| \leq \sqrt{\frac{1}{n}\sum_{ij} a_{ij}^2}^{\,n} .$$

Exercise 7: Deterministic Parallel Algorithms (15 points)

Devise a deterministic parallel algorithm (in CRCW PRAM), with $O(\log n)$ depth and $O(n)$ total work, for the following input and output:

Input: An arbitrary tree $T = (V, E)$ with $|V| = n$, provided as the adjacency lists of its vertices. That is, for each vertex $v \in V$, the vertices adjacent to v are given in a linked list $L[v] =< u_0, u_2, \ldots, u_{d-1} >$, where d is the degree of the vertex v. For each vertex $v \in V$, we are given a value $b(v) \in \{1, \cdots, n\}$. We are also given a root vertex $r \in V$.

Desired Output: For each vertex $v \in V$, we should output $score(v) = \sum_{u \in T_v} b(u)$. Here, T_v denotes all descendants of v (including v itself), with respect to root r, that is, all vertices u for which the shortest path connecting u to r includes v.

Exercise 8: Randomized Parallel Algorithms (15 points)

You are given as input an array $A[1..n]$ of n elements, which are pairwise comparable, i.e., for each $i, j \in \{1, ..., n\}$ where $i \neq j$, we have either $A[i] < A[j]$ or $A[j] < A[i]$. Devise a randomized parallel algorithm (in CRCW PRAM) that computes (an arbitrary) one of the $O(n^{2/3})$ smallest items, with probability $1 - O(1/n^5)$. Your algorithm should have $O(1)$ depth and use $O(n)$ total work.

A solution with $O(\log n)$ depth and $O(n)$ work receives 5 points.