Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Institute of Theoretical Computer Science
Thomas Holenstein, Ueli Maurer, Angelika Steger, Emo Welzl, Peter Widmayer

Algorithms, Probability, and Computing — Midterm Exam — HS13

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. You can solve the 4 tasks in any order. Begin by reading all tasks carefully. They are not ordered by difficulty or in any other meaningful way.
2. Check your exam documents for completeness (2 cover pages and 2 pages with 4 exercises).
3. Immediately inform an assistant in case you are not able to take the exam under regular conditions. Later complaints are not accepted.
4. Pencils are not allowed. Pencil-written solutions will not be reviewed.
5. 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.
6. Attempts to cheat/defraud lead to immediate exclusion from the exam and can have judicial consequences.
7. Provide only one solution to each exercise. Cancel invalid solutions clearly.
8. **All solutions must be understandable and well-founded. Write down the important thoughts in clear sentences and keywords. No points will be awarded for unfounded or incomprehensible solutions (except in the multiple-choice parts). You can write your solution in English or German.**
9. You may use anything that has been introduced and proved in the lecture without re-proving it. However, if you need something *different* than what we have in the notes, you must write a new proof or at least list all necessary changes.
10. 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	(20)	
2	(30)	
3	(30)	
4	(30)	
Σ	(110)	

Exercise 1: Multiple Choice (20 points)

Consider the following 4 claims and mark the corresponding boxes. Grading: 2 points for a correct marking without a correct justification, 5 points for a correct marking with a correct short justification, and -2 points for a wrongly marked box (you will receive non-negative total points for the exercise in any case).

(a) Let T_n be the unique tree in $\mathcal{B}_{[n]}$ on $n = 2^k - 1$ vertices with height $k-1$ for $k > 0$. Then $\Pr[T_n] \geq \frac{(\frac{n+1}{2})!}{n!}$, where $\Pr[\cdot]$ denotes the probability distribution defined on $\mathcal{B}_{[n]}$ from the lecture.

[ ] False [ ] True

Justification: ....

....

(b) Let $G = (V, E)$ be a multigraph with n vertices and at least two disjoint minimum cuts. Let $e \in_{\text{u.a.r.}} E$ and let $G' = (V', E') := G/e$. Then the probability of $\mu(G) = \mu(G'/e')$ for $e' \in_{\text{u.a.r.}} E'$ is at least $1 - \frac{2}{n-1}$.

[ ] True [ ] False

Justification: ....

....

(c) Consider the polynomial $p(x, y, z) = 3xyz + 5x^2y + 4xz + 2xy^2 + z^2$ in $GF(7)[x, y, z]$. Suppose that x, y, z are chosen independently and uniformly at random from $GF(7)$. Then the probability of $p(x, y, z) = 0$ is less than $\frac{1}{2}$.

[ ] False [ ] True

Justification: ....

....

(d) Consider the ClSP $F = \{\{x \neq 0, y \neq 0, z \neq 0\}, \{x \neq 0, y \neq 1\}, \{x \neq 2, z \neq 1\}\}$ on $V = \{x, y, z\}$ with domains $L_x = \{0, 1, 2\}$ and $L_y = \{0, 1\}$ and $L_z = \{0, 1, 2, 3\}$. Then the expected number of satisfied clauses of F for a uniformly random assignment of the variables is $31/12$.

[ ] True [ ] False

Justification: ....

....

Exercise 2: Expectations in Binary Search Trees (30 points)

Let $X_n^{(k)}$ be the number of nodes in the subtree rooted at the node with rank k in a random binary search tree on n nodes.

Let $Y_n^{(k)}$ be the number of nodes with depth k in a random binary search tree on n nodes.

(a) Compute $\mathbf{E}\left[X_n^{(k)}\right]$ for $n \geq 1$ and $1 \leq k \leq n$.

(b) Compute $\mathbf{E}\left[Y_n^{(1)}\right]$ for $n \geq 1$.

(c) Compute $\mathbf{E}\left[Y_n^{(2)}\right]$ for $n \geq 1$.

Exercise 3: Perfect Matchings (30 points)

Let $G = (V, E)$ be a planar graph on n vertices (n even). Let $E^+ \subseteq \binom{V}{2} \setminus E$ be a set of c edges not present in E for some constant $c \geq 0$. Now let $G^+ = (V, E \cup E^+)$. Note that G^+ might not be planar.

(a) Does G always admit a Pfaffian orientation? How about G^+?

(b) For the sake of simplicity, let $c = 2$ (in fact, you can do this for any c). Give an $O(n^3)$ algorithm to compute the number of perfect matchings of G^+.

For an orientation $\vec{G}$ with its skew symmetric matrix $A_s(\vec{G}) = (a_{ij})_{i,j \in [n]}$ (which we used to count perfect matchings in planar graphs), recall that

$$\det(A_s(\vec{G})) = \sum_{\pi \in \mathcal{S}_n} \operatorname{sign}(\pi) a_{1\pi(1)} a_{2\pi(2)} \ldots a_{n,\pi(n)} .$$

(c) Characterize the permutations π (dependent on G) for which

$$\operatorname{sign}(\pi) a_{1\pi(1)} a_{2\pi(2)} \ldots a_{n,\pi(n)} = 1$$

holds independently of the orientation chosen in $\vec{G}$.

Exercise 4: Line Segments (30 points)

(a) Let L be a set of n non-crossing line segments in general position. Let $\langle \ell \rangle := (\ell_1, \ldots, \ell_n)$ be a uniformly random permutation of L. Given $\langle \ell \rangle$, show how to preprocess L in $O(n \log n)$ expected time (over the choice of $\langle \ell \rangle$) such that you can answer queries of the following type in $O(\log n)$ expected time (over the choice of $\langle \ell \rangle$):

- Given a pair (k, q) where $1 \leq k \leq n$ and $q \in \mathbb{R}^2$, return the line segment from $\{\ell_1, \ldots, \ell_k\}$ that is intersected first by a vertical upward ray from q (if one exists).

You may assume that q is not on a vertical line through an endpoint of any segment of L.

(b) Design an efficient data structure that stores a set of n line segments in the plane, so that the number of segments intersected by a query line can be computed in $O(\log n)$ time.