**ETH**
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Institute of Theoretical Computer Science
Thomas Holenstein, Emo Welzl, Peter Widmayer

## Algorithms, Probability, and Computing — Midterm Exam — HS14

### 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 (2 cover pages and 7 pages with 6 exercises).
2. You can solve the six 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.
3. Immediately inform an assistant in case you are not able to take the exam under regular conditions. Later complaints will not be 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. Unless stated otherwise, no points will be awarded for unfounded or incomprehensible solutions. You can write your solutions in English or German.**
9. You may use anything that has been introduced and proved in the lecture or in the exercise sessions without reproving it. However, if you need something *different* than what we have shown, 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 | (25) | |
| 2 | (15) | |
| 3 | (15) | |
| 4 | (15) | |
| 5 | (15) | |
| 6 | (15) | |
| $\Sigma$ | (100) | |

## Exercise 1: Multiple Choice (25 points)

Consider the following five claims/questions 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 a non-negative total number of points for the whole exercise in any case.

(a) Recall that, in the worst case, both randomized quicksort ($qs_R$) and deterministic quicksort ($qs_D$) require $\binom{n}{2}$ comparisons when given a permutation $S$ of the numbers $1, \ldots, n$ as input. We call a permutation $S$ a *bad input for* $qs_R$ (resp., *for* $qs_D$) if, when given $S$ as input, it is possible that $qs_R$ (resp., $qs_D$) does exactly $\binom{n}{2}$ comparisons. Is the number of bad inputs for $qs_R$ at most the number of bad inputs for $qs_D$?

[ ] False [ ] True

Justification: ..........................................................................

..........................................................................

(b) Let $G$ be a catalog graph with vertices $1, \ldots, m$ and maximum degree 2, and let $n_1, \ldots, n_m$ be the sizes of the corresponding (non-empty) catalogs. Then, there exists a data structure of size $O(\sum_i n_i)$ that allows any multiple look-up query $(x, \pi)$ to be performed on $G$ in time $O(k + \log n_j)$ where $k$ is the size of the output and $j$ is the first vertex in $\pi$.

[ ] False [ ] True

Justification: ..........................................................................

..........................................................................

(c) There exists a non-zero polynomial $p \in \mathbb{R}[x_1, x_2, x_3]$ which satisfies $p(x) = 0$ for all vectors $x \in \mathbb{R}^3$ with $\|x\| \leqslant 1$.

[ ] False [ ] True

Justification: ..........................................................................

..........................................................................

(d) In a permutation $\pi$ chosen uniformly at random from the set $S_n$, what is the expected number of inversions (i.e., pairs $(i, j)$ with $i < j$ and $\pi(j) < \pi(i)$)?

[ ] $\frac{n^2}{2} - \frac{n}{2}$ [ ] $\frac{n^2}{2} - \frac{n}{4}$ [ ] $\frac{n^2}{4} - \frac{n}{2}$ [ ] $\frac{n^2}{4} - \frac{n}{4}$

Justification: ..........................................................................

..........................................................................

(e) Let $X_n$ denote the number of descendants of the key with smallest rank in a random search tree with $n$ keys. In an exercise we showed that $\mathsf{E}[X_n] = H_n$ holds for all $n$. Is it also true that $\Pr\left[X_n \geqslant \sqrt{n}\right] = O(\frac{\log n}{\sqrt{n}})$?

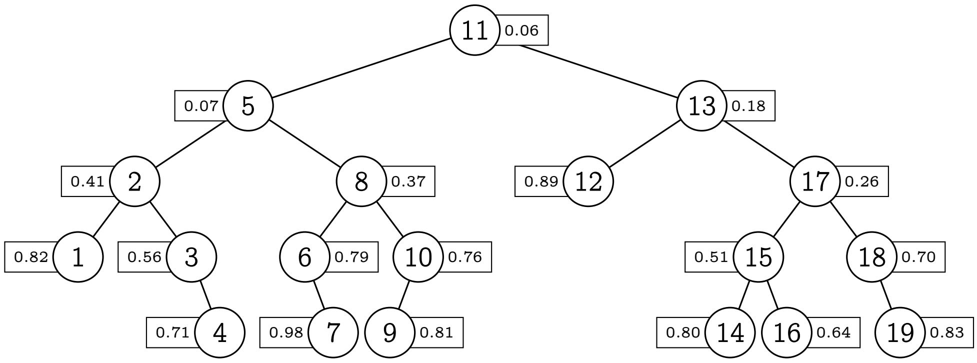
[ ] False [ ] True

Justification: ..........................................................................

..........................................................................

**Exercise 2: Random(ized) Search Trees (5 + 10 points)**

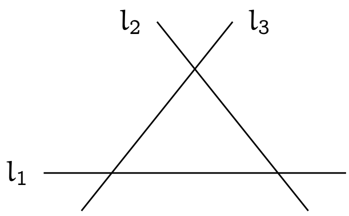
(a) Consider the following treap which was obtained by inserting the keys $1, \ldots, 19$ in some order into an initially empty treap. Let $x$ be the key inserted last, and suppose that during the insertion of $x$ exactly 3 rotations were performed. What are the possible values of $x$?

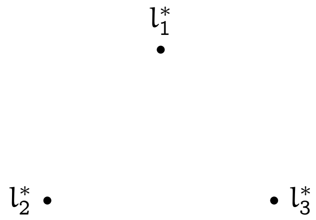



(b) A node in a binary tree is called *cute* if its left subtree has exactly one node and, at the same time, its right subtree has exactly one node (in the treap above, 15 is the only cute node). What is the expected number of cute nodes in a random search tree on $n$ nodes?

## Exercise 3: Point Location (5 + 10 points)

(a) Depicted on the left there is a planar line arrangement with lines $l_1, l_2, l_3$. Depicted on the right there are the dual points $l_1^*, l_2^*, l_3^*$ of these lines. In the figure on the right, draw the set of dual points of all lines which pass through exactly one of the three vertices of the line arrangement on the left. Do not forget to give a justification!

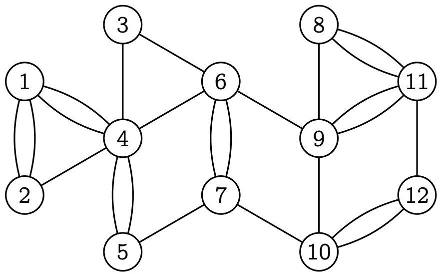



(b) Let $L$ be a set of $n$ lines in $\mathbb{R}^2$. We assume that no two lines are parallel, that no three lines intersect in a common point, and that the distances between the pairwise intersection points of these lines are distinct. Moreover, let $l_1, \dots, l_n$ be a permutation of $L$ chosen uniformly at random, and suppose that $l_1, \dots, l_n$ are inserted one by one into an initially empty line arrangement. Compute (exactly) the expected number of insertions which change the length of the shortest edge in the current line arrangement (where we count the insertions of the first two lines $l_1$ and $l_2$ as changes).

## Exercise 4: Minimum Cut (5 + 10 points)

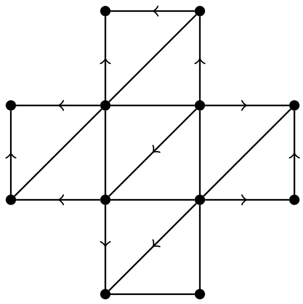
(a) Let $G$ be the multigraph with vertices $1, \dots, 12$ depicted below, and let $e$ be an edge of $G$ chosen uniformly at random. Calculate the value of $\Pr[\mu(G) = \mu(G/e)]$.




(b) Let $\mathcal{A}$ be a randomized algorithm which, when given any multigraph $G$ on $n$ vertices as input, outputs a certain (random) value $A$, where $A = \mu(G)$ with probability $1/n$ and $A < \mu(G)$ in all other cases. Let $A_1, \dots, A_k$ be the values returned by $k$ independent invocations of $\mathcal{A}$ with input $G$. Choose $k$ as small as possible and describe another algorithm $\mathcal{B}$ which takes $A_1, \dots, A_k$ as input and which outputs a value $B$ such that $B = \mu(G)$ with probability at least $1 - n^{-42}$. You do not have to prove that your choice of $k$ is optimal.

## Exercise 5: Randomized Algebraic Algorithms (5 + 10 points)

(a) Consider the graph depicted below, which is already partially oriented. Orient the remaining edges (without changing any of the already oriented edges) in order to obtain a Pfaffian orientation. Do not forget to give a justification!

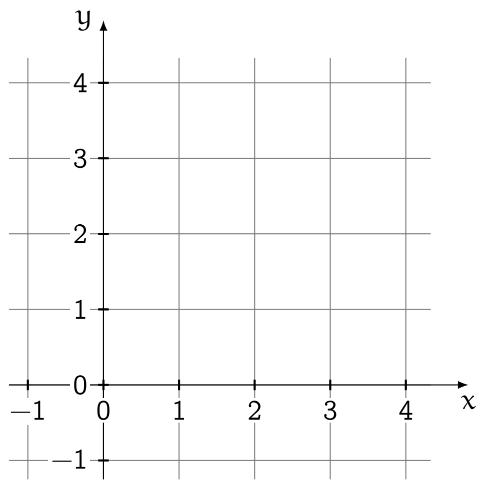


(b) Let $n \geqslant 1$ be any natural number. Furthermore, let $A \in \mathbb{R}^{n\times n}$ be a matrix which has at least one non-zero entry, and let $x$ and $y$ be two vectors chosen independently and uniformly at random from the set $\{-1, 0, 1\}^n$. Prove that $\Pr\left[x^\mathsf{T} A y \neq 0\right] \geqslant \frac{4}{9}$.

**Exercise 6: Linear Programming (5 + 10 points)**

(a) Consider the linear program LP in two real variables $x$ and $y$, which is defined in the figure below. In the coordinate system on the right, draw (i) the set of all feasible solutions of LP and (ii) the optimal solution of LP. Argue briefly why you have indeed found the optimum.

$$\text{LP:} \quad \begin{aligned} \text{maximize} \quad & x + 2y \\ \text{subject to} \quad & x \geqslant 0 \\ & y \geqslant 0 \\ & 2x + y \leqslant 8 \\ & x + y \leqslant 5 \\ & 2y - x \leqslant 4 \end{aligned}$$




(b) In order to celebrate the end of the midterm exam, you would like to make exactly one kilogram of delicious cake, for which you are going to use a mixture of the ingredients *milk*, *flour*, *egg* and *sugar*. Of each ingredient there is an unlimited amount at your disposal. Of course, every ingredient adds a different amount of the qualities *toughness*, *moisture* and *flavor* to the cake. We assume that these qualities can be expressed as real numbers. The corresponding ratings (how much of each quality is added to the cake per kilogram of the respective ingredient) can be seen in the table below. Your cake will be *delicious* if it has a toughness of 5, a moisture of 4, and a flavor of 7. Sadly, it is impossible to make delicious cake with only your four basic ingredients. What you want to make instead is one kilogram of cake whose qualities are as close as possible to being delicious. Each quality is equally important to you, and you consider having too much of a certain quality equally bad as having too little of that quality.

| | Milk | Flour | Egg | Sugar |
|---|---|---|---|---|
| Toughness | −5 | 20 | 5 | −5 |
| Moisture | 30 | −5 | 10 | 0 |
| Flavor | 0 | 0 | 0 | 50 |

Model the described problem as a linear program. You do not have to find the optimal solution, but if you want you can bring it to the lecture tomorrow and share it with your colleagues.