Institute of Theoretical Computer Science
Emo Welzl, Angelika Steger, Peter Widmayer

11/10/2008

Algorithms, Probability, and Computing *Fall 08*
Mid-Term Exam

Candidate:

First name: ....

Last name: ....

Registration Number: ....

I attest with my signature that I could have taken 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 5 exercises in any order. **You should not be worried if you cannot solve all the exercises** (best grade with at least 48 points, i.e. 4 exercises).
2. Check your exam documents for completeness (1 cover sheet and 3 sheets containing 5 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 additives allowed.
6. Attempts to defraud yield 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 articulative sentences and keywords. Unfounded or incomprehensible solutions will not be awarded. You can write your solution in English or German.**
9. Make sure to write your name on all the sheets.

Good luck!

	achieved points (maximum)	reviewer's signature
1	(12)	
2	(12)	
3	(12)	
4	(12)	
5	(12)	
Σ	(60)	

Exercise 1

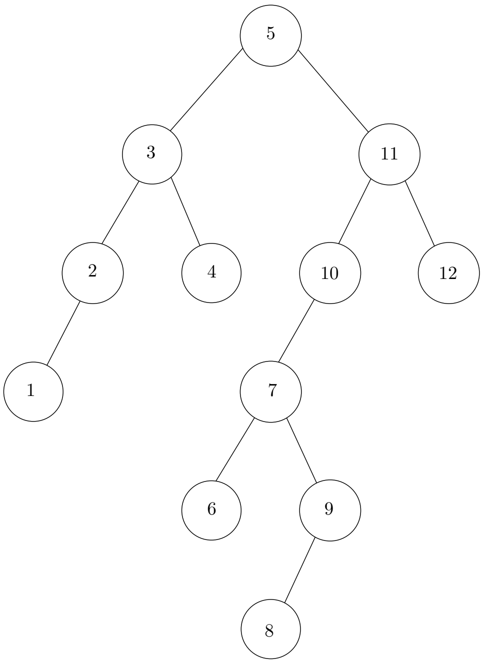
In this exercise we consider random search trees for n keys.

(a) What is the probability that the element with rank i is a common ancestor of the element with rank 1 and the element with rank 4? (You will have to distinguish between several cases)

(b) What is the probability that the element with rank i is on the path from the root to the smallest element?

Exercise 2

Let T be a search tree for $\{1, 2, \ldots, n\}$. The picture below shows an illustration



(a) By a *fancy triple* we denote a triple $\{i, i+1, i+2\}$ such that i is an ancestor of $i+2$ and $i+2$ is an ancestor of $i+1$. (In the above picture we have two fancy triples, i.e., $\{5, 6, 7\}$ and $\{7, 8, 9\}$) What is the expected number of fancy triples in a random search tree for $\{1, 2, \ldots, n\}$?

(b) Let $k \leq n$ be an integer. We say that a tree contains a k*-tail* if i is the child of $i+1$ for every i with $1 \leq i \leq k-1$. E.g. the tree in the above illustration contains a 3-tail (and therefore also a 2-tail and a 1-tail). What is the probability that a random search tree for $\{1, 2, \ldots, n\}$ contains a k-tail?

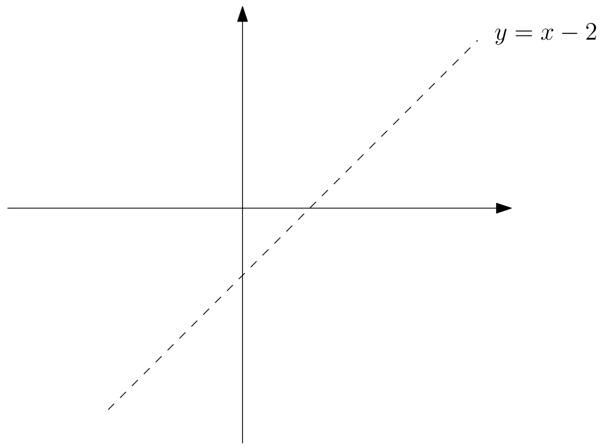
Exercise 3

Consider the line $l : y = 2$. Let S be the set of non-vertical lines $l' \neq l$ such that

(i) l' intersects l and

(ii) the x-coordinate of the intersection point is at least 1.

Draw the set of points whose dual is in S!



Exercise 4

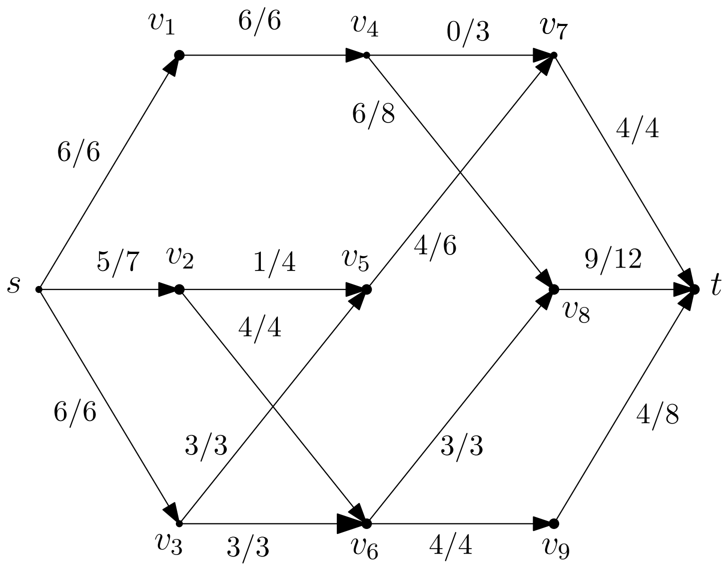
In this exercise we want to obtain a suitable preprocessing/query algorithm for efficiently deciding whether a point is inside a convex polygon, including giving a short certificate. Let P be a convex polygon with n vertices. We are interested in the following query. The input is a point q in the plane. The output is

$$\begin{cases} \text{3 vertices of } P \text{ such that the triangle spanned by these 3 vertices contains } q, & \text{if } q \text{ is in } P \\ \text{no,} & \text{otherwise} \end{cases}$$

Describe a data structure that stores P in $O(n)$ space so that the query can be processed in time $O(\log n)$.
Hint: If a point q is in P then lowest vertex $p_\perp$ of P is part of a triangle containing q.

Exercise 5

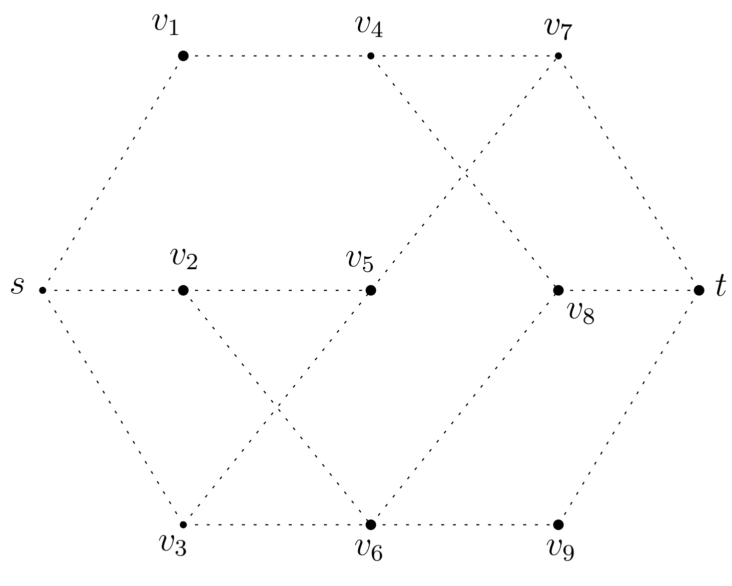
Consider the below network (G, s, t, c) with flow f (the edge labels are flow/capacity).



(a) What is the value of f?

(b) Draw the residual network. You can use the below figure (a dotted edge just indicates that in the original graph there is a directed edge at this place).

(c) Prove that f is a maximum flow by determining a cut $S \subseteq V(G)$ with $\mathsf{cap}(S) = \mathsf{val}(f)$ (you might want to recall the proof of the maxflow-mincut theorem for that); provide S by listing its vertices below.



$S = \{ \qquad\qquad\qquad\qquad \}$