Prof. Friedrich Eisenbrand
Nicolai Hähnle

Discussions from: April 5, 2011

Computer Algebra

Spring 2011

Assignment Sheet 4

Exercises marked with a ⋆ can be handed in for bonus points. Due date is April 19.

Exercise 1
Prove that

$$\operatorname{ord}_p(n!) = \frac{n - S_p(n)}{p-1},$$

where $S_p(n)$ is the sum of the digits of n written in base p.

Exercise 2 (⋆)
Develop an algorithm that given an odd-degree polynomial $f \in \mathbb{Z}[x]$ and $\varepsilon > 0$ computes an interval of length at most ε enclosing a root of f using binary search in polynomial time in the encoding length of f and ε. Prove the correctness of your algorithm.

Exercise 3
Let $A \in \mathbb{Q}^{n\times n}$. Denote the columns of A by $a_1, \ldots, a_n$. Let B be an upper bound on the absolute values of entries in A.

1. Show the Hadamard bound $|\det(A)| \le \prod_{j=1}^n |a_j|_2$, where $|\cdot|_2$ is the Euclidean norm.

   Hint: Equality holds when the a_j are pairwise orthogonal.

2. Derive from this that $|\det(A)| \le n^{n/2} B^n$. How does this compare to the bound derived from Leibniz' formula?

Exercise 4 (⋆)
Show that, using Gauss elimination, one can compute a solution to the system $Ax = b$, $A \in \mathbb{Q}^{m\times n}$, $b \in \mathbb{Q}^m$, or assert that none exists, in polynomial time in the encoding length of A and b.

Note: You will have to extend results on the row echelon form to show that the encoding length of the numbers involved remains polynomial.

Exercise 5
Let

$$A = \begin{pmatrix} 1 & 0 & -2 \\ 2 & -1 & 1 \\ 0 & 2 & 2 \end{pmatrix}$$

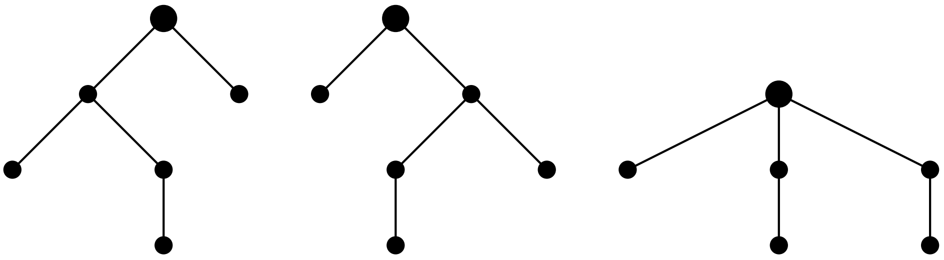
1. Use Gauss elimination modulo p to compute the determinant of A modulo p, for $p = 3, 5, 7$.

2. Use the Leibniz bound to show that $2|\det(A)|+1 \leq 105$. Conclude that you can directly obtain $\det(A)$ from the previous results.

3. Develop this approach into an algorithm that computes the determinant of a matrix $A \in \mathbb{Z}^{n\times n}$ while using only arithmetic with small integers (except for a combination step that computes the final result). Show that your algorithm runs in polynomial time in the input length.

Exercise 6
Two rooted trees T_1 and T_2 are said to be isomorphic if there exists a bijection f from the vertices of T_1 to those of T_2 satisfying the following conditions: the root of T_1 is mapped to the root of T_2, and for each vertex v of T_1 with children $v_1,\ldots,v_k$, the vertex $f(v)$ has as children exactly the vertices $f(v_1),\ldots,f(v_k)$. No ordering is assumed on the children of any internal vertex.

Of the following three examples, the first two are isomorphic, while the last example is isomorphic to neither of the first two.



Associate to each vertex v a polynomial f_v recursively: for a leaf vertex, set $f_v = x_0$. For an internal vertex v of height h with children $v_1,\ldots,v_k$, set

$$f_v = (x_h - f_{v_1})(x_h - f_{v_2})\cdots(x_h - f_{v_k}) \in \mathbb{Z}[x_0,\ldots,x_h]$$

1. Show that two rooted trees are isomorphic if and only if the polynomials associated to their roots are equal.

2. Develop an efficient randomized algorithm for testing whether two rooted trees are isomorphic and analyze its running time and probability of success.