

Randomized algorithms – Problem 1-2

Problem 1

Define the decision problems

$$\begin{aligned} \text{FACTORIZATION} &= \{(n, k) \mid \text{integer } n \text{ has factor } f \text{ with } 1 < f < k\} \\ \text{PRIMALITY} &= \{n \mid \text{integer } n \text{ is a prime number}\} \end{aligned}$$

In the following you may use that $\text{PRIMALITY} \in \text{P}$.

- a. Show $\text{FACTORIZATION} \in \text{NP} \cap \text{co-NP}$.
- b. Show that if $\text{FACTORIZATION} \in \text{RP} \cup \text{co-RP}$ then $\text{FACTORIZATION} \in \text{RP} \cap \text{co-RP}$.

Problem 2

If $n = \prod_{i=1}^{\omega} p_i^{m_i}$, is the complete prime factorization of n , then $\phi(n) = \prod_{i=1}^{\omega} (p_i - 1)p_i^{m_i - 1}$, where ϕ denotes Eulers function, and $\phi(n)$ is the order of the multiplicative group $|\mathbf{Z}_n^*| = \phi(n)$.

- a. Design and analyse a randomised polynomial time algorithm that given n and $\phi(n)$ constructs the complete factorisation of n .

The discrete logarithm problem is the following. Given g, n, a , find some i such that $g^i \equiv a \pmod{n}$ (or report that no such i exists).

An oracle for a problem returns the correct answer immediately without using any resources.

- b. Design and analyse a randomised polynomial time algorithm that given n and an oracle for the discrete logarithm problem factors n .