Wrocław University of Technology, WPPT 27.10.05

CRYPTOGRAPHY, 2004 Assignments, list # 2

1. Write a pseudocode for finding inverse of a number a modulo n. Implement such an algorithm so that no division is used – this is necessary for reducing the execution time. (Such an algorithm exists and is called *binary gcd*.)
2. Consider the tasks of computing $m^e \bmod n$ for a fixed parameter e. Assume that the binary representation of e contains only a few ones. Which exponentiation algorithm is best suited for this case?
3. Let n be a RSA number. Is it true that every $a < n$ has 4 square roots?
4. Let n be a RSA number. Let $e < \phi(n)$ be an arbitrary number coprime with $\phi(n)$. Given $a < n$. How many roots of degree e of a exist? Discuss all cases.
5. Estimate the number of $k < n$ such that k has at least one divisor less than B. Based on this estimation discuss how much influence on the runtime of primality test has preliminary trial divisions by small prime numbers.
6. Estimate the probability that Miller-Rabin primality test presented during the last lecture finds an appropriate witness for an RSA number n (i.e. a witness that proves n to be composite).

/-/ Mirosław Kutyłowski