Technische Universität München
Fakultät für Informatik
Lehrstuhl für Theoretische Informatik
Prof. Dr. Susanne Albers
Dr. Arindam Khan
Maximilian Janke

Winter Semester 2018/19
Problem Set 4
November 13, 2018

Advanced Algorithms

Due November 27, 2018 at 10:00

Note: You are welcome to submit in groups of two. If you wish to submit individually Exercises 2 and 3 are to be solved.

Exercise 1 (RSA – 10 points)

For an RSA encryption choose $p = 31$, $q = 17$. Moreover, let $e = 131$.

1. Compute the number d and specify the outputs of the algorithm EXTENDED-EUCLID. Furthermore, give the public and private key.

2. Generate a digital signature for the message $M = 72$. What does a recipient of the message have to check in order to verify the signature?
   Hint: For generating the signature, use the fast exponentiation algorithm POWER *but omit the check for square roots of 1 (modulo n).*

Exercise 2 (Primality Test – 10 points)

Consider the RANDOMIZED PRIMALITY TEST algorithm taught in the class. Let n be a composite number. We pick $a \in \{2, 3, \ldots, n-1\}$ uniformly at random. If $a^{n-1} \neq 1 \pmod{n}$ then we call a to be a witness of n. Unfortunately, there are composite numbers, known as *Carmichael numbers (CN)*, that have no witnesses. However, in this exercise we will show that CN are the only bad inputs for the algorithm. We use the notation $\mathbb{Z}_n^*$ to be the multiplicative group of integers coprime to n (i.e., with $gcd(a, n) = 1$).

1. Let $\mathcal{S}_n := \{a \in \mathbb{Z}_n^* : a^{n-1} = 1 \bmod n\}$, i.e., $\mathcal{S}_n$ is the set of all $a \in \mathbb{Z}_n^*$ that are not witnesses. Show that $\mathcal{S}_n$ is a proper subgroup of $\mathbb{Z}_n^*$ if n is not a Carmichael number.

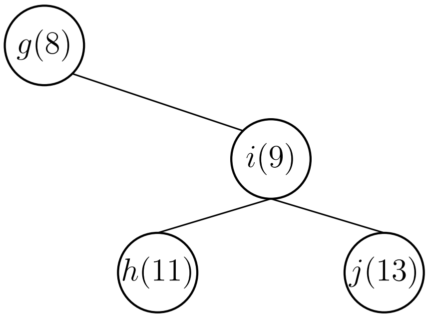
2. Now show that if n is composite and not a Carmichael number, then $Pr[a \text{ is not a witness of } n] \leq 1/2$.
   Hint: Show that if $a \in \mathcal{S}_n$ and $c \in \mathbb{Z}_n^* \setminus \mathcal{S}_n$ then $a \cdot c \in \mathbb{Z}_n^* \setminus \mathcal{S}_n$

Exercise 3 (Operations on Treap – 10 points)

1. Sequentially insert the keys c, d, e, b, a with respective priorities $7, 3, 10, 1, 4$ into an initially empty treap. For all the intermediate stages, e.g. after performing a rotation, illustrate the state of the treap and specify the operation that leads to this state.

2. Delete the root of the treap resulting from part 1. Again illustrate treap prior to and after each rotation.

3. Merge the treap resulting from part 2 and the treap shown below. Illustrate all intermediate stages.



Exercise 4 (Binary Search Tree – 10 points)

Let T be a binary search tree, in which all keys are distinct. Consider a leaf x of T and let y be its parent. Show that $key(y)$ is either the smallest key in T larger than $key(x)$ or the largest key in T smaller than $key(x)$.