Technische Universität München
Fakultät für Informatik
Lehrstuhl für Theoretische Informatik
Prof. Dr. Harald Räcke
Richard Stotz, Dennis Kraft

Winter term 2016 / 2017
Assignment 3
November 7, 2016

# Efficient Algorithms and Data Structures I

*Deadline: November 14, 10:15 am in the* ***Efficient Algorithms*** *mailbox.*

## Homework 1 (4 Points)

Solve the following recurrence relation:

$$a_n = -a_{n-1} + 9a_{n-2} - 11a_{n-3} + 4a_{n-4} \text{ with } a_0 = -7, a_1 = 4, a_2 = 48 \text{ and } a_3 = 0.$$

## Homework 2 (5 Points)

Solve the following recurrence relation without using generating functions

$$a_n = 4(2a_{n-1} - 5a_{n-2} + 4a_{n-3}) \quad \text{with} \quad a_0 = 6,\ a_1 = 28, a_2 = 108\ .$$

## Homework 3 (4 Points)

Give tight asymptotic bounds for the following recurrence relation:

$$T(n) = T(\sqrt{n}) + 1$$

## Homework 4 (7 Points)

This exercise will provide an alternative method for analyzing homogeneous linear recurrences. Consider the following recurrence:

$$a_n = 6a_{n-1} - 11a_{n-2} + 6a_{n-3} \qquad \text{for } n \geq 3\ .$$

with $a_0 = 3$, $a_1 = 6$, and $a_2 = 14$.
Let $b_n = a_{n-1}$ for $n \geq 1$ and $c_n = b_{n-1}$ for $n \geq 2$. Finally, let $\vec{x}_n = (a_{n+2}, b_{n+2}, c_{n+2})^T$ for $n \geq 0$.

1. Determine a matrix $M \in \mathbb{R}^{3\times 3}$ such that $\vec{x}_n = M \cdot \vec{x}_{n-1}$ for $n \geq 1$.
2. Let $\lambda$ be an eigenvalue of $M$. Show that $\vec{x}_0$ can be chosen such that $\lambda^n \cdot \vec{x}_0$ is a solution to the recurrence relation derived in the first part.
3. Determine the eigenvalues of $M$ and, for each eigenvalue, determine a corresponding eigenvector.
4. Clearly $x_0 = (14, 6, 3)^T$. Use the eigenvectors and eigenvalues of $M$ to solve the recurrence relation derived in the first part.
5. Determine a closed form for $a_n$.

## Bonus Homework 1 (10 Bonus Points)

Give tight asymptotic bounds for the following recurrence relation:

$$T(n) = T\left(\frac{n}{\log n}\right) + 1$$

**Hint:** How often do you have to apply the iteration $n \longmapsto \frac{n}{\log n}$ until the problem size drops to $\sqrt{n}$? How often do you have to apply it to bring it down from $\sqrt{n}$ to $\sqrt{\sqrt{n}}$? Also use the fact that $\sum_{i=1}^{k} \frac{2^i}{i} = O(\frac{2^k}{k})$.
**Hint 2:** Don't copy old solutions from the internet.

## Tutorial Exercise 1

In the early days of the internet, when there were only the websites `tum.de` and `lmu.de`, the random surfer Androidus aimlessly walked the web. If Androidus was on the page of TUM, he stayed there with probability 0.9 and moved to the LMU with probability 0.1. If he was on `lmu.de`, he stayed with probability 0.8 and moved to `tum.de` with probability 0.2.

1. Let $a_n$ and $b_n$ denote the probability that Androidus is after timestep $n$ on `tum.de` and `lmu.de`, respectively. Express $a_n$ and $b_n$ using a recurrence relation for $n \geq 1$. Also draw the corresponding Markov chain.

2. Brian, the creator of Androidus, asks for a closed form of $a_n$ and $b_n$, assuming that the bot starts on `tum.de`. Find these expressions without using generating functions.

3. Determine the limiting distribution of Androidus over the web. How is it linked to the eigenvectors?

```
[The PageRank Computation] is es-
sentially the determination of the
limiting distribution of a random
walk on the web graph.
      - Page, Brin, Motwani, Winograd
```