**Analysis of Algorithms, WS 2020**
Prof. Dr. P. Rossmanith
Dr. E. Burjons, H. Lotze, D. Mock

Date: 4 November 2020

# Exercise Sheet 02

## Problem T3

In this warmup exercise we are going to analyse $C'$—the number of comparisons done in quicksort during partitioning phases *on the left side* of the array. All comparisons are distributed among the following two lines in the program. We split the number of comparions into $C'$ for the first line and $C''$ for the second line.

**do** { $i$++; } **while**($a[i] < k$);
**do** { $j$−−; } **while**($k < a[j]$;

We would expect $C'$ should be around half the number of comparions, i.e., around $C/2$.

## Problem T4

If a flow diagram consists of $n$ nodes and $m$ edges, how many fundamental cycles do we get?

## Problem T5

Prove or disprove: In every flow diagram you can find a spanning tree such that all fundamental cycles contain only edges that are labeled with plus.

## Problem T6

Let $w \in \{a, b\}^n$ a word that has been chosen uniformly at random. How often is the body of the `while`-loop executed on average in the following algorithm? The function `is_palindrome` tests whether a word is a palindrome, i.e., the same when read backwards.

```
i = 2;
while (i <= n) {
  if (is_palindrome(w[1],...,w[i])) return true;
  i++;
}
return false;
```

## Problem H4 (15 credits)

We consider the following Algorithm. The array `a` contains a random permutation of the the numbers $1, \ldots, N$.

```
void doSomething(int *a, int N)|
{
  int i;|

  for (i=0; i<N-1; i++)   /*  1  */
    while (a[i] > a[i+1]) /*  2  */
      a[i]--;             /*  3  */
}
```

How often is line 3 executed on average?

**Problem H5** (15 credits)

In this exercise, we consider Prim's Algorithm, which computes a minimum spanning tree. The input to this algorithm is a graph $G = (V, E)$, a weight function on the edges $w\colon E \to \mathbf{R}$ and a starting node $r$.

```
for each u ∈ V do
    key[u] ← ∞
    π[u] ← NIL
key[r] ← 0
M ← V
while (M ≠ ∅) do
    u ← min-from(M)
    for each v ∈ neighbors(u) do
        if (v ∈ M) ∧ (w(u, v) < key[v]) then
            π[v] ← u
            key[v] ← w(u, v)
```

Construct the control flow graph, a spanning tree in the control flow graph, the fundamental cycles, a corresponding linear system of equations and a solution to this system.