Week 2 Preparation

Instructions to the students: The preparation problems are not assessed, but we strongly recommend that you try to solve them before your applied class this week. These preparation problems test your basic knowledge of the contents taught in the seminar of the previous week. The problems in the applied class assume that you have this basic knowledge and will build on top of it. You might find it helpful to try these problems before doing the quiz that is due this week.

Problem 1. Given the following pseudocode, derive the recurrence relation that represents its time complexity. Define the base case and recurrence step.

```
1: function FACTORIAL(n)
2:     if n = 0 then return 1
3:     else return n × FACTORIAL(n − 1)
```

Problem 2. Given the following pseudocode, derive the recurrence relation that represents its time complexity. Define the base case and recurrence step.

```
1: function IS_POWER_OF_TWO(n)
2:     if n = 1 then return true
3:     else if n < 1 then return false
4:     else return IS_POWER_OF_TWO(n/2)
```

Problem 3. Given the following pseudocode, derive the recurrence relation that represents its time complexity. Define the base case and recurrence step.

```
1: function FIND_MIN(arr[1..n])
2:     if n = 1 then return arr[1]
3:     else
4:         potentialMin = FIND_MIN(arr[2..n])
5:         if arr[1] > potentialMin then
6:             return potentialMin
7:         else return arr[1]
```

Problem 4. Find a closed form solution for the following recurrence relation:

$$T(n) = \begin{cases} T(n-1)+c, & \text{if } n > 0, \\ b, & \text{if } n = 0. \end{cases}$$

[Hint: Use telescoping in order to express $T(n)$ in terms of $T(n-2)$ instead of $T(n-1)$, then in terms of $T(n-3)$ instead of $T(n-2)$, and so on until you can figure out the general pattern. Then use the base case to obtain a formula for $T(n)$ that only depends on n, b and c, but not on $T(\cdot)$.]

Problem 5. Find a closed form for the following recurrence relation:

$$T(n) = \begin{cases} 3T(n-1), & \text{if } n > 0, \\ c, & \text{if } n = 0. \end{cases}$$