Problems Solved:

46	47	48	49	50

Name:

Matrikel-Nr.:

Problem 46. Let $L = \left\{ 1^n 0 1^m \,\middle|\, m, n \in \mathbb{N} \wedge n > 0 \wedge m = n^2 \right\}$.

- Describe (informally) a Turing machine M with $L(M) = L$.
- Analyse the time and space complexity of M.

Problem 47. An n-bit binary counter counts in $2^n - 1$ steps from $(00\ldots0)_2 = 0$ to $(11\ldots1)_2 = 2^n - 1$ and in one more step back to $(00\ldots0)_2 = 0$. The cost of a step is the number of bits changed at that step. (For instance, the cost of increasing a 4-bit counter from `1011` to `1100` is 3 since 3 bits are modified.)

1. Consider how often bit position i changes in the 2^n cycles and compute the sum of the number of changes of all positions. The amortized cost is this sum divided by the number of cycles.
2. Compute the amortized cost by applying the potential method.
   Use as the potential $\Phi(s_i)$ (where s_i is the counter after the i-th application of the increment operation) $\Phi(s_i) = b(s_i)$ where $b(s_i)$ is the number of 1s in the binary representation of the counter.
   For the computation of an upper bound of the amortized cost $\hat{c}_i$ derive inequalities $b(s_i) \leq \ldots$ and $c_i \leq \ldots$ using the notion $t(s_i)$ for the number of bits reset from 1 to 0 by the i-th increment operation.

Problem 48. Take the following recursive program.

```
1 f(n,b) ==
2     if n < 1 then return 0
3     d := floor(n/3)
4     return b + f(d,1) + 2*f(d,2)
```

Let $C(n)$ be the number of comparisons executed in line 2 while running $f(n, 0)$ for some positive integer n.

1. Write down a recurrence for C and determine enough initial values.
2. Solve that recurrence for the given initial values and arguments n of the form $n = 3^m$.
3. Prove by induction that your solution is correct.

Problem 49. Does there exist for every finite language $L \subseteq \{0, 1\}^*$ a Turing machine D such that D

1. takes as input the code $\langle M \rangle$ of a Turing machine M that stops on every input and
2. decides whether $L \subseteq L(M)$ holds?

Justify your answer.

Problem 50. Consider a RAM program that evaluates the value of $n! = \prod_{i=1}^{n} i$ in the naive way (by iteration). Analyze the worst-case asymptotic time and space complexity of this algorithm on a RAM assuming the existence of operations operation `ADD r` and `MUL r` for the addition and multiplication of the accumulator with the content of register r.

1. Determine a Θ-expression for the number $S(n)$ of registers used in the program with input n (space complexity).
2. Determine a Θ-expression for the number $T(n)$ of instructions executed for input n (time complexity in constant cost model),
3. Determine an O-expression for the asymptotic time complexity $C(n)$ of the algorithm for input n assuming the logarithmic cost model for a RAM (with cost $al \cdot rl$ for operation `MUL r` where al is the digit length of the content of the accumulator and rl is the digit length of the content of register r).

   Hint: approximate in the asymptotic analysis summation $\sum_{i=a}^{b} T$ by integration $\int_a^b T \mathrm{d}i$ and solve this integral; you may use a computer algebra system or WolframAlpha for this purpose.