Problems Solved:

41	42	43	44	45

Name:

Matrikel-Nr.:

Problem 41. Prove or disprove the following:

1. $O(g(n))^2 = O(g(n)^2)$
2. $2^{O(g(n))} = O(2^{g(n)})$

Hint: First transform above equations into a form that does not involve the O-notation on the left hand side, then prove the correctness of the resulting formulas.

Problem 42.

1. Consider the probability space $\Omega = \{0,1\}^n$ of all strings over $\{0,1\}$ of length n where each string occurs with the same probability 2^{-n}. Let $X : \Omega \to \mathbb{N}$ be a random variable that gives the position of the first occurrence of the symbol 1 in a string, if 1 occurs at all. For completeness, we also define that $X(0^n) = 0$. Positions are numbered from 1 to n. Give a (not necessarily closed form) term for the expected value $E(X)$ of the random variable X and justify your answer.

2. Evaluate the sum

$$S = \sum_{k=1}^{n} \frac{1}{2^k} k$$

in *closed form*, i.e., find a formula for the sum which does not involve a summation sign.

Hint: Using your high-school knowledge about geometric series, compute a closed form of the function

$$F(z) := \sum_{k=0}^{n} \left(\frac{z}{2}\right)^k.$$

and compute its first derivative $F'(z)$. The desired closed form is then $S = F'(1)$. Why?

Note that the index for the geometric series starts at $k = 0$.

Problem 43. Let $M = (Q, \Gamma, \sqcup, \Sigma, \delta, q_0, F)$ be a Turing machine with $Q = \{q_0, q_1\}$, $\Sigma = \{0,1\}$, $\Gamma = \{0, 1, \sqcup\}$, $F = \{q_1\}$ and the following transition function δ:

δ	0	1	$\sqcup$
q_0	$q_0 0 R$	$q_1 1 R$	–
q_1	–	–	–

1. Determine the (worst-case) time complexity $T(n)$ and the (worst-case) space complexity $S(n)$ of M.

2. Determine the average-case time complexity $\bar{T}(n)$ and the average-case space complexity $\bar{S}(n)$ of M. (Assume that all 2^n input words of length n occur with the same probability, i.e., $1/2^n$.)

3. Bonus: Using results from Problem 42, express all answers in closed form, i.e., without the use of the summation symbol.

Problem 44. Let $T(n)$ be the number of multiplications carried out by the following Java program.

```
int a, b, i, product, max;
max = 1;
a = 0;
while ( a < n ) {
  b = a;
  while (b <= n) {
    product = 1;
    i = a;
    while (i < b) {
      product = product * factors[i];
      i = i + 1; }
    if (product > max) { max = product; }
    b = b + 1; }
  a = a + 1; }
```

1. Determine $T(n)$ exactly as a nested sum.

2. Determine $T(n)$ asymptotically using Θ-Notation. In the derivation, you may use the asymptotic equation

$$\sum_{k=0}^{n} k^m = \Theta(n^{m+1}) \text{ for } n \to \infty$$

for fixed $m \geq 0$ which follows from approximating the sum by an integral:

$$\sum_{k=0}^{n} k^m \simeq \int_0^n x^m \, dx = \frac{1}{m+1} n^{m+1} = \Theta(n^{m+1})$$

Problem 45. Let $T(n)$ be given by the recurrence relation

$$T(n) = 3T(\lfloor n/2 \rfloor).$$

and the initial value $T(1) = 1$. Show that $T(n) = O(n^\alpha)$ with $\alpha = \log_2(3)$.
Hint: Define $P(n) :\iff T(n) \leq n^\alpha$. Show that $P(n)$ holds for all $n \geq 1$ by induction on n. It is not necessary to restrict your attention to powers of two.