Aufgabe 31. Seien M_1, M_2 zwei Turingmaschinen und seien $L_1 = L(M_1)$ und $L_2 = L(M_2)$ die zugehörgien akzeptierten Sprachen. Man könnte versuchen, eine Prozedur (im intuitiven Sinn) zum Akzeptieren von $L_1 \cup L_2$ wie folgt zu beschreiben:

> Man gebe ein gegebenes Inputwort x zunächst als Input für M_1; falls x von M_1 akzeptiert wird, wird x durch die Prozedure akzeptiert. Anderenfalls gebe man x als Input für M_2 und akzeptiere x falls M_2 das Wort x akzeptiert.

1. Was stimmt an diesem Argument nicht?
2. Man gebe eine verfeinerte Prozedur an, welche $L_1 \cup L_2$ tatsächlich akzeptiert.
3. Man zeige, daß auch $L_1 \cap L_2$ von einer geeigneten Prozedure (im intuitiven Sinn) akzeptiert wird.

Anmerkung: Die Prozeduren aus den letzten beiden Punkten lassen sich ohne weiteres auf Turingmaschinen implementieren; $L_1 \cup L_2$ und $L_1 \cap L_2$ werden also ebenfalls von geeigneten Turingmaschinen akzeptiert.

Aufgabe 32. Sei eine Funktion f definiert als

$$f(n) = \begin{cases} 3n+1 & \text{falls } n \text{ ungerade} \\ \frac{n}{2} & \text{falls } n \text{ gerade} \end{cases}$$

Man kann sich nun folgende Prozedur vorstellen. Man wähle eine positive natürliche Zahl und wende f darauf an. Ist das Ergebnis gleich 1, so stoppe man, anderenfalls, wende man wieder f auf das Ergebnis an und führe das Verfahren solange fort, bis man eine 1 als Ergebnis erhält.
Sei ν eine Funktion, die als Input positive natürliche Zahlen akzeptiert und als Ausgabe die Anzahl der Iterationen ausgibt, die im obigen Verfahren notwendig sind, um als Ergebnis für f eine 1 zu erhalten.
Zeigen Sie dass, ν eine rekursive Funktion ist. Sie können dabei alle Sätze aus dem Skriptum verwenden.
Sehr schwierige Zusatzaufgabe: Ist ν sogar primitiv rekursiv?

Aufgabe 33. Welche der folgenden Aussagen sind richtig? Begründen Sie.

1. $\log(n^{100})$ ist $O(n)$
2. $\exp(n^{100})$ ist $O(n)$
3. $2n$ ist $O(n)$
4. Für alle $\varepsilon > 0$ gilt: $\sqrt{e^n}$ ist $O(e^{\varepsilon n})$.
5. Für alle $\varepsilon > 0$ und für alle $k \in \mathbb{N} \setminus \{0\}$ gilt $\mathrm{e}^{\varepsilon n}$ ist $O(n^k)$
6. Für alle $\varepsilon > 0$ und für alle $k \in \mathbb{N} \setminus \{0\}$ gilt $\mathrm{e}^{\varepsilon n}$ ist $O(k^n)$
7. 2^n ist $O(8^n)$
8. 8^n ist $O(2^n)$

Aufgabe 34. Beweisen Sie: Seien $f(n)$ und $g(n)$ von der Ordnung $O(n)$. Dann gilt:

1. $(f+g)(n)$ ist von der Ordnung $O(n)$;
2. $(f \cdot g)(n)$ ist von der Ordnung $O(n^2)$.

Zeigen oder widerlegen Sie: Falls $g(n)$ von der Ordnung $O(f(n))$ ist, ist auch $2^{g(n)}$ von der Ordnung $O(2^{f(n)})$.

Aufgabe 35. Bestimmen Sie die Anzahl der Multiplikationen, die benötigt werden, um die Polynome

$$a_0x^3 + a_1x^2 + a_2x + a_3 \text{ und } b_1x^2 + b_2x + b_3$$

mit der klassischen und mit der Karatsuba Methode zu multiplizieren.