Aufgabe 1

Zu zeigen:
$A \subset \Sigma^*$ ist lexikographisch rekursiv aufzählbar $\Longleftrightarrow$ A ist entscheidbar.

Wir zeigen zuerst $\Longrightarrow$:

Sei A also lexikographisch rekursiv aufzählbar. Dann gilt entweder A ist endlich, womit A sofort auch entscheidbar ist, oder es gibt eine überall (Turing-)berechenbare Funktion $G : \mathbb{N} \longrightarrow \Sigma^*$ die A in lexikographischer Ordnung aufzählt, sprich: $\forall n \in \mathbb{N} : G(n)$ ist der n-kleinste String in A.

Sei nun M eine Turingmaschine (TM), die A lexikographisch rekursiv aufzählt. Wir suchen nun eine TM M_A, die A entscheidet, sprich: M_A hält nach endlich vielen Schritten und akzeptiert eine Eingabe $w \in \Sigma^*$ genau dann, wenn $w \in A$. M_A verfährt wie folgt: M_A simuliert M solange, bis M entweder w aufzählt, oder ein Wort aufzählt, das lexikographisch größer ist als w. Im ersten Fall akzeptiert M_A den Input w im zweiten Fall weißt sie w zurück. Damit entscheided M_A die Sprache A und daher ist A entscheidbar.

Wir zeigen $\Longleftarrow$:

Seit A also entscheidbar. Das heißt A ist entweder endlich, und damit auch lexikographisch rekursiv aufzählbar, womit nichts zu zeigen bleibt, oder es exisitiert eine TM M_A, die A entscheided.

Für den zweiten Fall konstruieren wir aus dieser TM M_A eine TM M, die alle Worte $w \in A$ nacheinander in lexikographischer Reihenfolge konstruiert wie folgt: M konstruiert alle $w \in \Sigma^*$ nacheinander in lexikographischer Reihenfolge (z.B: $\varepsilon, a, b, aa, ab, ba, \ldots$) und nutzt M_A um zu beurteilen ob das jeweilig generierte Wort in A ist. Wenn dem so ist geht sie in einen dafür vorgesehenen Zustand $IstInA$. In jedem Fall jedoch setzt M danach die Berechnung des lexikographisch-nächstgroßen Wortes fort. Die Existenz von M zeigt die Existenz einer überall berechenbaren Funktion G wie gefordert. Also ist A lexikographisch rekursiv aufzählbar.

Aufgabe 2

Es sei $A \subset \Sigma^*$, $B \subset \Gamma^*$ und $C \subset \Delta^*$, sowie $A \preceq B$ und $B \preceq C$, dann gibt es nach Definition $f : \Sigma^* \to \Gamma^*$ und $g : \Gamma^* \to \Delta^*$, so dass $x \in A \Leftrightarrow f(x) \in B$ und $y \in B \Leftrightarrow g(y) \in C$ für alle $x \in \Sigma^*, y \in \Gamma^*$.
Dann folgt $x \in A \Leftrightarrow f(x) \in B \Leftrightarrow g(f(x)) \in C$. Mit $g \circ f$ haben wir eine Reduktion von A auf C gefunden. Die $\preceq$-Relation ist also transitiv.

Aufgabe 3

Gegeben sind eine Partitionierung von Q in zwei disjunkte, nichtleere Teilmengen Q_1 und Q_2 und eine TM

$$M = (\Sigma, \Gamma, \#, Q, s, F, \Delta)$$

mit Codierungen für M, Q_1 und Q_2. Wir betrachten die Sprachen

$$B = \{cod(M)\$cod(Q_1)\$cod(Q_2) \mid Q_1, Q_2 \text{ sind sequentiell in M}\}$$

$$A = \{cod(M) \mid L(M) = \emptyset\}$$

B formalisiert das gegebene Problem. A ist eine nicht entscheidbare Sprache (vgl. Satz von Rice). Um zu zeigen, dass B nicht entscheidbar ist, wird $A \preccurlyeq B$ gezeigt, d.h. wir finden ein F, sodass $x \in A \Leftrightarrow F(x) \in B$.

Sei $x = cod(M)$ gegeben, mit $M = (\Sigma, \Gamma, \#, Q, s, F, \Delta)$. Wir konstruieren eine DTM $M' = (\Sigma', \Gamma', \#', Q', s', F', \Delta')$

wie folgt:

$$\begin{aligned}
\Sigma' &= \Sigma \\
\Gamma' &= \Gamma \\
\#' &= \# \\
Q' &= Q \cup \{f'\}, \quad f' \notin Q \\
s' &= s \\
F' &= \{f'\} \\
\Delta' &= \Delta \cup \{(f, y, f', y, B) \mid f \in F,\ y \in \Gamma\ \}
\end{aligned}$$

F gibt $cod(M')\$cod(Q_1)\$cod(Q_2)$ zurück mit $Q_1 = Q$, $Q_2 = \{f'\}$. Diese Konstruktion erfüllt die Bedingungen für unsere Reduktion:

- $x \in A \Rightarrow F(x) \in B$: Sei $L(M_x) = \emptyset$. Dann erreicht M nie einen Endzustand. M' kann somit nie f' erreichen und bleibt immer in Zuständen $q \in Q_1$, also gilt $F(x) \in B$.

- $x \notin A \Rightarrow F(x) \notin B$: Sei $L(M_x) \neq \emptyset$. Dann erreicht M für mindestens eine Eingabe $x \in \Sigma^*$ einen Endzustand. Damit erreicht M' für Eingabe xy den Zustand f' und macht einen Übergang von Q_1 nach Q_2, d.h. $F(x) \notin B$.

Aufgabe 4

Es sei $\Sigma = \{0, 1, \$\}$. Wir betrachten

$$L = \{u\$v \mid u \in SAM \text{ und } v \in \overline{SAM}\}.$$

Weder L noch $\overline{L}$ sind akzeptierbar.

a) Wir zeigen $\overline{SAM} \leq L$. Dazu konstruieren wir eine TM-berechenbare Funktion $f : \Sigma^* \to \Sigma^*$ mit $x \in \overline{SAM} \iff f(x) \in L$.

Es sei M_1 eine Turing-Maschine mit $M_1 \in SAM$. Wir setzen $f(x) = \langle M_1 \rangle \$x$. Dann gilt:

$$\begin{aligned}
&x \in \overline{SAM} \Longrightarrow f(x) = \langle M_1 \rangle \$x \in L \\
&x \notin \overline{SAM} \Longrightarrow x \in SAM \Longrightarrow f(x) = \langle M_1 \rangle \$x \notin L.
\end{aligned}$$

Aus der Vorlesung wissen wir, daß $\overline{SAM}$ nicht akzeptierbar ist. Mit Behauptung f aus Aufgabe 1 ($L_1 \leq L_2$ und L_1 nicht akzeptierbar $\Longrightarrow L_2$ nicht akzeptierbar) sehen wir, daß L nicht akzeptierbar ist.

b) Wir zeigen $\overline{SAM} \leq \overline{L}$. Dazu sei M_2 eine Turing-Maschine mit $M_2 \notin SAM$. Wir setzen $f(x) = x\$\langle M_2 \rangle$. Dann gilt:

$$\begin{aligned}
&x \in \overline{SAM} \Longrightarrow f(x) = x\$\langle M_2 \rangle \notin L \Longrightarrow f(x) \in \overline{L} \\
&x \notin \overline{SAM} \Longrightarrow x \in SAM \Longrightarrow f(x) = x\$\langle M_2 \rangle \in L \Longrightarrow f(x) \notin \overline{L}.
\end{aligned}$$

Wie oben folgt, daß $\overline{L}$ nicht akzeptierbar ist.