Matrikelnummer:

1. O-**Notation** (5 Punkte).

/5

Welche der folgenden Aussagen sind richtig? Falsche Aussagen sollten korrigiert werden, indem die richtige Beziehung zwischen den Funktionen angegeben wird.

(a) $n^2 = O(n \log^2 n)$

(b) $3^n = 2^{O(n)}$

(c) $\sqrt{n} = O(n/\log n)$

(d) $2^{\log^2 n} = O(n^4)$

(e) $(3 \log n)/n = O(1)$

Matrikelnummer:

2. **Kürzeste Wege** (8 Punkte).

/8

Gegeben sei ein gerichteter azyklischer Graph $G = (V, E)$ zusammen mit einer Gewichtsfunktion $w : E \to \mathbb{R}$. Entwerfen und begründen Sie einen Linearzeitalgorithmus (linear in der Anzahl der Knoten plus Kanten), der für einen Startknoten s die Längen von kürzesten (bzgl. der Funktion w) Wegen von s zu allen anderen auf gerichteten Wegen erreichbaren Knoten berechnet. Bauen Sie gleichzeitig in Linearzeit eine Datenstruktur, mit der (implizit) kürzeste Wege zu den Knoten abgespeichert werden.

3. **Lokales Alignment** (8 Punkte).

/8

Seien S_1 und S_2 Strings über einem endlichen Alphabet Σ. Sie wissen, dass das Lokale–Alignment–Problem darin besteht, Teilstrings der beiden Strings mit maximaler Ähnlichkeit v^* zu finden und dass man/frau dies mit Dynamischen Programmieren elegant lösen kann. Dazu sei s eine scoring-Tabelle und $v(i,j)$ der Wert einer maximalen Ähnlichkeit zwischen einem Präfix des i–ten Suffix von S_1 und einem Präfix des j–ten Suffix von S_2.

(a) Die Rekursionsvorschrift zur Berechnung der $v(i,j)$ lautet

$$v(i,j) = max\{0, v(i-1,j-1) + s(S_1(i), S_2(j)),$$

$$v(i-1,j) + s(S_1(i), -), v(i,j-1) + s(-, S_2(j))\}$$

Die Initialisierung ist, wie Sie auch wissen, $v(i,0) = v(0,j) = 0$ für alle i,j.
Begründen Sie die Richtigkeit dieser Vorschrift.(2/8)

(b) Berechnen Sie den Wert v^* für die Strings $S_1 = a\ b\ c\ x\ d\ e\ x$ und $S_2 = x\ x\ x\ c\ d\ e$.
Die Scoring–Funktion bewertet Matches mit +2 und Mismatches bzw. Spaces mit -1.(3/8)

(c) Welche Teilstrings im obigen Beispiel realisieren durch welches konkrete Alignment diesen Wert v^*? (3/8)

Matrikelnummer:

4. **Separierte Palindrome** (9 Punkte).

/9

Ein Teilstring $S[i \dots j]$ eines Strings S heißt *l–separiertes Palindrom*, $l \geq 0$, wenn er die Form ucu^{rev} hat mit $u \in \Sigma^+$ und $c \in \Sigma^*$, $|c| \leq l$. In diesem Fall ist $S[i \dots j]$ ein *maximales* l-separiertes Palindrom, falls $S[i-1 \dots j+1]$ kein l–separiertes Palindrom ist.

(a) Beschreiben und begründen Sie einen Linearzeitalgorithmus, der für festes l und beliebigen String S alle l–separierten Palindrome berechnet. (7/9)

(b) Beachten Sie insbesondere, dass die Ausgabe kompakt dargestellt ist und kein maximales l–separiertes Palindrom $S[i \dots j]$ doppelt aufgeführt wird. (2/9)