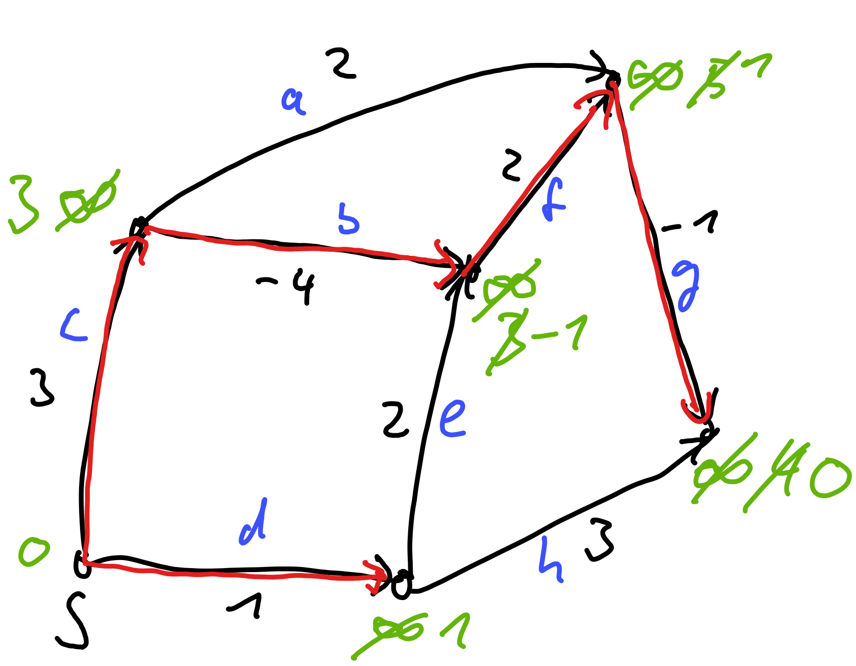
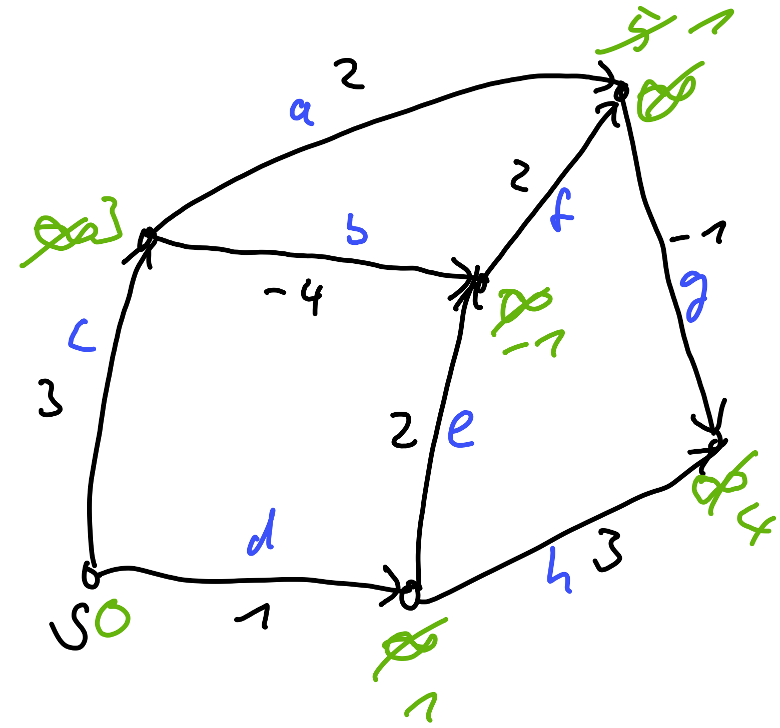
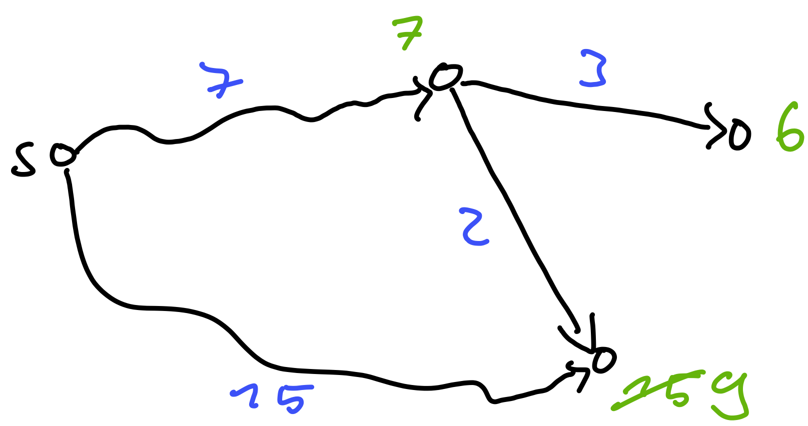
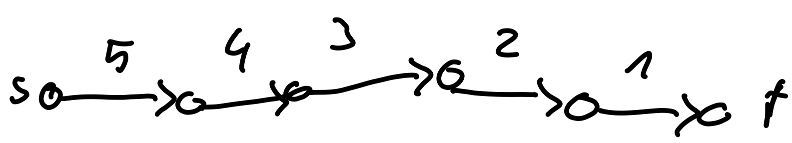








Zeigen: Pfad der Länge n von s nach t: $s = v_1, v_2, v_3, \dots, v_n = t$

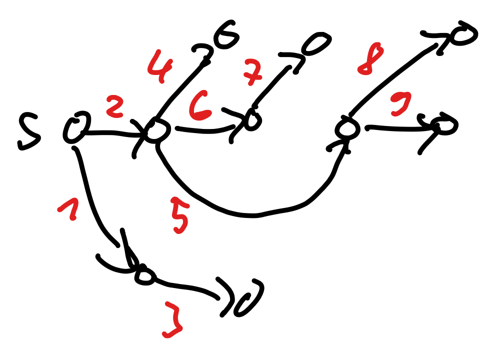
Kantenreihenfolge: von hinten nach vorne. $(v_{n-1}, v_n), \dots, (v_1, v_2)$

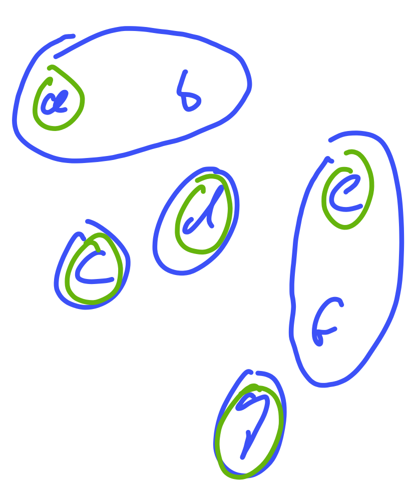
Beh.: Das benötigt $n-1$ Iterationen.

Bew: Nach Iteration i haben die Knoten $v_1, \dots, v_{i+1}$ den korrekten Wert, alle nachfolgenden haben den Wert ∞. ⇝ Induktion

Zeigen: Wir wissen aus der VL, dass nach k Schritten der kürzeste Pfad zu jedem Knoten berechnet wurde unter allen Pfaden, die aus höchstens k Kanten bestehen.

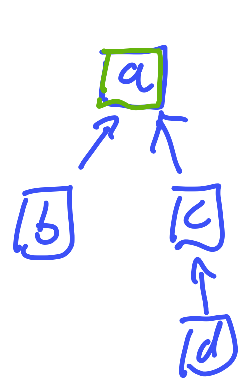
Kürzeste-Wege-Baum hat Höhe h ⇒ zu jedem Knoten gibt es kürzesten Weg mit nur höchstens h Kanten ⇒ nach h Iterationen gefunden.



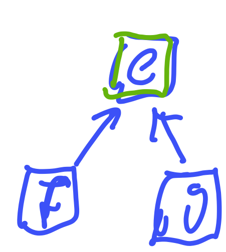


union(a, b)
union(e, f)
find(b) ⇝ a
find(g) ⇝ g

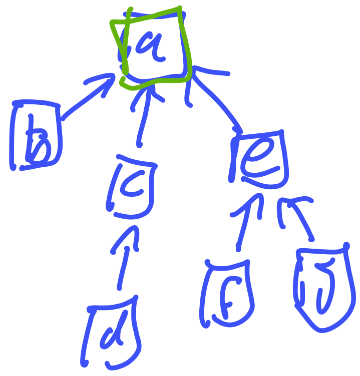
Menge a, b, c, d

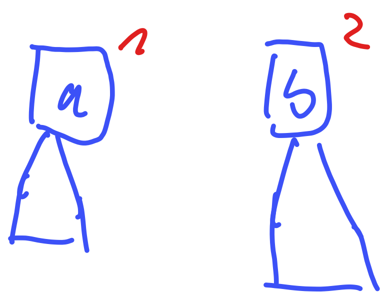


Menge e, f, g

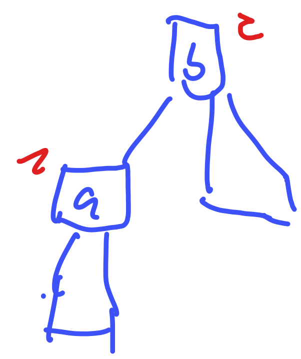


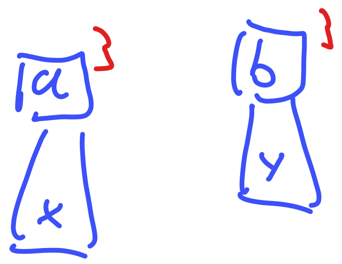
Union a, e





Union(a, b)

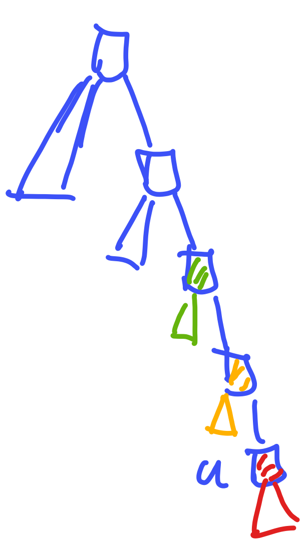




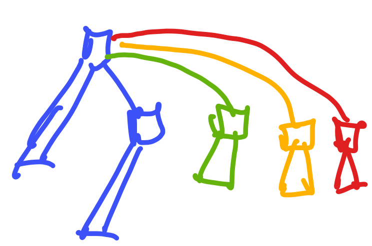
Union(a, b)

union(x, y)

Baum unter Knoten mit Rang i hat $\geq 2^i$



find(u) $\rightsquigarrow$



1 2 3 4 5 6 7 8 9

5, 2, 1, 3, 1, 4, 0, 3, 4

1 1 2 2 2 3

Wofür soll $T[i]$ stehen?

($T[i]$: Länge der längsten Teilfolge von $\langle a_1, \dots, a_i \rangle$

Wie bekommen wir $T[i]$ aus $T[0], \dots T[i-1]$

$T[1], \dots, T[4], T[5] \to T[6]$)

$T[i]$: Länge der längsten Teilfolge die auf a_i endet.

$$J_i = \{ j < i \mid a_i = a_j \pm 1 \}$$

$$T[i] = \begin{cases} \max\limits_{j \in J_i} \{ T[j] + 1 \} & \text{wenn } J_i \neq \emptyset \\ 1 & \text{, sonst} \end{cases}$$