Aufgabe 1.

a) The runtime of the Algorithm 1 for the shortest path problem equals $\mathcal{O}(n \cdot m)$, where n equals the size of vertices and m equals the size of edges.

Beweis. The initialization of the lengths runs in $\mathcal{O}(n)$. For the worst case scenario, for every length update a new shorter path will be revealed, until every vertex is finally evaluated $()\mathcal{O}(n))$. To find the newly revealed shortest path, it takes $\mathcal{O}(m)$ comparisons. Hence, the algorithm will run in $\mathcal{O}(n \cdot m)$ in the worst case and in $\mathcal{O}(\max\{n, m\})$ in best case. □

b) $A <_{\text{aymp}} B <_{\text{aymp}} E <_{\text{aymp}} C <_{\text{aymp}} D$

Beweis.

- $A <_{\text{aymp}} B$

$$2^{\sqrt{\log n}} = 2^{\log n^{\frac{1}{2}}} = 2^{\frac{1}{2}^{\log n}} = \sqrt{2}^{\log n}$$

Also, the following holds. Let f be a monotone increasing function and $c > 1$. Then:

$$\lim_{\infty} \frac{c^f}{f} > 1$$

- $B <_{\text{aymp}} E$

$$2^{\sqrt{\log n}} < 2^{\sqrt{\log n \cdot \log n}} = n < 28n$$

- $E <_{\text{aymp}} C$

$$\log n^{\log n} = (2^{\log \log n})^{\log n} = n^{\log \log n}$$

Therefore, every linear function out of $\mathcal{O}(n)$ will be outgrown by C.

- $C <_{\text{aymp}} D$

$$\mathcal{O}(2^{\log \log n}) \subset \mathcal{O}(2^n)$$

□

Aufgabe 2.