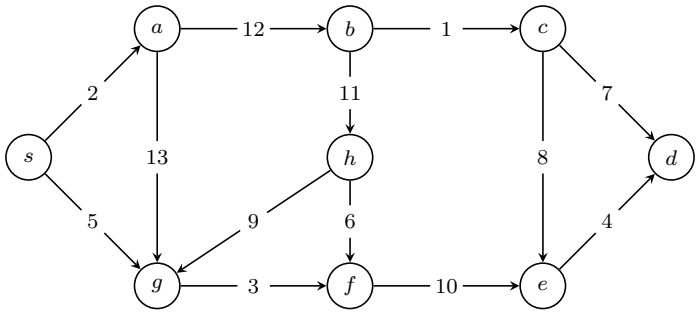
Universität Siegen
Lehrstuhl Theoretische Informatik
Markus Lohrey

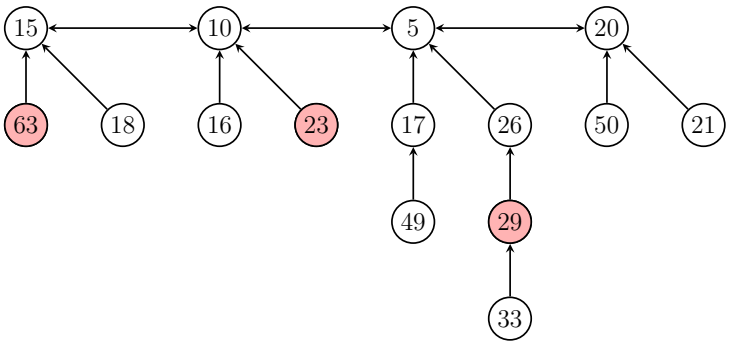
Exercise 7

Task 1
Use Dijkstra's algorithm (slide 130) to compute all shortest paths starting at s. Give the values of the program variables B, R, U, p, D after each iteration of the main **while**-loop.



Task 2
Perform the following sequence of operations on the depicted Fibonacci heap.

- **delete-min**;
- **decrease-key**('49', 3);
- **insert**(14);
- **delete-min**;
- **decrease-key**('23', 9);
- **decrease-key**('33', 25).



Task 3
Prove or disprove the statement: every Fibonacci heap with n nodes has heigth $\mathcal{O}(\log n)$.

Task 4
Find the optimal order to compute the product of matrices of the following dimensions.

$$(3 \times 6) \cdot (6 \times 3) \cdot (3 \times 1) \cdot (1 \times 10) \cdot (10 \times 10)$$

Task 5
Given a natural number n, we want to compute the least number of steps required to obtain $k = n$ from $k = 0$ where, in each step, we can either increment k (i.e., $k \leftarrow k + 1$) or double k (i.e., $k \leftarrow k \cdot 2$). For example, $n = 6$ can be obtained in 4 steps. Design an algorithm which solves this problem in time polynomial in n using dynamic programming.