Exam Preparation 2

Exercise 1 *Hashing*

- Describe the different hashing mechanisms given in the lecture and compare them.
- Consider an example where hashing has to deal with collisions. How are collisions resolved by the different hashing approaches?
- Prove the upper bounds on the expected execution times (assuming random hash functions) for the different hashing approaches discussed in the lecture.
- What does it mean that a class of hash functions is called c-universal?
- Define a class of 1-universal hash functions.
- Choose a random hash function from a class of 1-universal hash functions efficiently.

Exercise 2 *Graph Algorithms*

- Give the algorithms DFS and BFS and show how they work on an example graph. What are the running times of these algorithms.
- Give an algorithm that computes the strongly connected components for a given directed graph. What is the running time?

Exercise 3 *Shortest Paths*

- Give an algorithm that solves the single-source-shortest path problem for a given weighted graph where the edge weights are positive.
- Show the execution of this algorithm on a example graph.
- Give the Bellman-Ford algorithm for solving the all-pairs-shortest-path problem with positive edge weights.

Exercise 4 *Minimum Spanning Trees*

- Give Kruskal's algorithm for the computation of a minimum spanning tree and show the execution of this algorithm on an example graph.
- Analyze the runtime of Kruskal's algorithm and show how union and find operations can be supported efficiently.
- Give the Jarník-Prim algorithm for the computation of a minimum spanning tree and show the execution of this algorithm on an example graph.

Exercise 5 *P and NP*

- Characterize the classes P and NP.
- What does it mean that a problem is NP-hard?
- How do you prove that a problem is NP-complete?
- Show that the Traveling Salesman Problem is NP-complete. You can assume that the Hamiltonian Cycle problem is NP-complete.
- Give three other examples of NP-complete problems.