Oral Exam Question Guide

1 Topic of Your Choice

Prepare one topic from the lecture in very detail (including definitions and proofs). Talk about this topic freely in the first half of the oral exam (about 15min). In the second part questions like the ones listed in the following (but not exclusively!) will be asked about the topics you have not chosen.

2 Randomized Algorithms and Data Structures

2.1 MAX 3-SAT

- Define the MAX 3-SAT problem.
- Describe a randomized algorithm for MAX 3-SAT and prove its expected quality.
- Is this algorithm a Monte Carlo or a Las Vegas algorithm?
- Can the algorithm be used to construct a LV or MC algorithm? If yes, how. If no, why.
- Why does an expected number of x satisfied clauses implies that there exists an assignment with x satisfied clauses?

2.2 k-Select, Approximate Median and QuickSort

- Describe a deterministic k-Select algorithm that runs in linear time.
- Describe the randomized k-Select algorithm. Prove its runtime.
- What is the relation between the Approximate Median algorithm and the k-Select algorithm?
- What would be the runtime of QuickSort if you get the perfect pivot element for free in each round?
- What would be the runtime of QuickSort if always the smallest element in the actual set is selected as pivot? What if always the second smallest element is selected?
- Prove that randomized QuickSort runs expectedly in $\mathcal{O}(n \log n)$.
- Sketch the idea why no deterministic or randomized sorting algorithm for general inputs can achieve a better theoretical runtime than randomized QuickSort.

2.3 SkipLists

- What problem is tackled with SkipLists?
- How are deterministic SkipLists constructed?
- How does one search for an element using SkipLists?
- What happens when searching for an element not contained in the list?
- What is the maximum number of operations in a SkipList with 2 levels? What about i levels?
- How are randomized SkipLists constructed?
- In what aspect can randomized SkipLists improve on deterministic SkipLists?
- What is the expected space consumption of a randomized SkipList data structure?

2.4 Randomized Hashing

- Define the dictionary problem.
- Why does hashing elements to random cells does not solve the problem?
- Show that for a large enough universe there exists a subset of size n such that all elements are hashed to the same location.
- Give an example of a hash function that would perform good on the universe being the set of natural numbers. Show a choice for S such that all elements in S are hashed to the same location using this hash function.
- When is a hash family called universal?
- What is the collision probability of two elements if the hash function was chosen u.a.r. from a universal hash family?
- Show that the hash family that contains all hash functions is universal.
- Why is this hash family not used in practice?
- Define the MaxCut problem. Describe a naive deterministic strategy to solve it to optimality.
- Describe a randomized algorithm for the MaxCut problem and investigate its expected quality.
- How can hashing be applied to the MaxCut problem in order to have a polytime algorithm which guarantees a solution quality similar to the one expected for the randomized algorithm?

2.5 Random Sampling, VC-dimension and ϵ-Nets

- Name applications for sampling.
- What does the Basic Sampling Theorem state?
- Define the HittingSet problem.
- What is a universal sample?
- There exists a polytime k-approximation for the HittingSet problem where all sets contain at most k elements. Why does this not help to find a universal sample?
- Define the VC-dimension.
- What does the VC-dimension mean in the context of a set system on shortest paths?
- What is the VC-dimension of a set of intervals in $\mathbb{R}$?
- What is the Vc-dimension of a set of circles?
- Give two examples for set systems with unbounded VC-dimension.
- What is the maximum number of different subsets of a set of size n? What is the maximum number of subsets if we know that the VC-dimension of the system is at most d?
- What is an ϵ-net? How is it related to a HittingSet?
- If for arbitrary ϵ an ϵ-net of size c/ϵ can be found in polytime, what does this mean for the approximability of the underlying HittingSet problem?
- What does the ϵ-Net Theorem state?
- Describe one application of ϵ-Nets in teh context of route planning.

3 Getting Beyond Deterministic Bounds via Randomization

3.1 The Closest Pair (CP) Problem

- Define the CP problem.
- Describe a naive deterministic approach and analyze its runtime.
- Describe a divide-and-conquer strategy for CP.
- Describe the incremental algorithm.
- Give an example of an input sequence of points for which the incremental approach reaches its worst-case runtime. How are input sequences characterized on which the incremental approach performs best?
- At which pint does randomization help to improve the incremental algorithm?
- Analyze the runtime of the randomized incremental CP algorithm.

3.2 The MinCut Problem

- Define the MinCut problem.
- Describe a naive deterministic strategy to solve the problem.
- What is a multigraph?
- What is the basic operation in the randomized algorithm?
- What can we say about the MinCut in the original graph and the MinCut in a graph after edge contraction?
- Assume the MinCut size is k. Give a lower bound on the number of edges in the graph.
- What does success in a round of the algorithm mean?
- What is the success probability in the first round?
- What is the success probability in the last round?
- What is the total success probability of the algorithm?
- What is amplification and how can it be used in the context of the MinCut algorithm?
- Define high probability.
- What is the runtime of a MinCut algorithm which is correct with high probability?
- Describe the idea behind FastCut.
- Describe the FastCut algorithm in detail.
- What is the runtime of the FastCut algorithm?

3.3 Sublinear Algorithms

- What is a connected component?
- How can the connected component of a node be identified in linear time?
- How does the sample algorithm work?
- What is a minimum spanning tree?
- How can we use the algorithm for estimating the number of connected components to estimate the weight of a minimum spanning tree?

3.4 Long Path Problems

- Define the Long Path problem in dependency of a parameter k.
- Describe a naive strategy to solve the problem.
- Describe the DAG based algorithm to find a long path in the graph.
- What is the probability that a long path in the original graph is also a long path in the DAG and why?
- What is the idea behind the colouring algorithm?
- Which of the two algorithms is more effective?

4 Randomization in Games and AI

4.1 The Cow-Path Problem

- Define competitiveness.
- Define the Cow-Path problem.
- Why is a strategy for the Cow-Path problem that does not alternate between lanes not competitive for sure?
- Describe a deterministic and a randomized strategy, if the cow is aware about the distance to the grass. What is the (expected) walking distance?
- Let $f(i)$ be the function that describes how far the cow walks on lane 1 (for even i) and on lane 2 (for odd i). What is the walking distance if $f(i) = i$? What would be a better function?
- How competitive is the deterministic strategy with a well chosen function $f(i)$?
- How competitive is the strategy when introducing randomization?
- What is the rough idea behind the Smartcow algorithm?

4.2 Robots Running Against Walls

- Describe the Wall problem.
- Why is there no competitive strategy for non-convex obstacles?
- Why is the robot allowed to squeeze between touching obstacles?
- Sketch the idea how an adversary would place obstacles in the way of the robot such that he has to take a path of a factor $\Theta(\sqrt{n})$ longer than the optimum.
- What is a y-path? How long is a y-path if no obstacles cross this y-coordinate?
- Describe an upwards sweep. What is the difference to a downwards sweep?
- When does a sequence of many upwards/downwards sweeps in a corridor does not bring the robot closer to the wall?
- Describe the idea of iterative doubling in this context.
- Why is it not a problem that we do not know W (the summed length of vertical parts of the optimal path)?

4.3 Bidding

- Describe the bidding setting.
- Describe a 4-competitive strategy.
- What is the idea behind the randomized strategy?

5 Online-Algorithms

5.1 Paging

- What is the Paging problem?
- Describe three deterministic strategies.
- What is the idea in the proof that shows that LRU is k-competitive?
- Prove that no deterministic strategy has a competitive ratio better than k.
- Describe the Marking algorithm.

5.2 When to Rent Skies

- Define the Ski-Rental problem.
- Provide a formula for the optimal solution.
- Describe a 2-competitive strategy.
- What happens in the worst case if you decide to buy skies immediately or if you decide to rent skies forever?
- Describe the randomized approach for Ski Rental.

5.3 The BahnCard Problem

- Define the BahnCard problem.
- How can the Ski Rental problem be expressed in terms of the BahnCard problem?
- What is the critical value?
- Describe the SUM algorithm.
- What are cheap and expensive phases?
- How are the costs for SUM compared to the optimal solution in a cheap interval?
- What changes in the randomized version R-SUM?

6 The Probabilistic Method

6.1 Independent Set

- Define the Independent Set problem.
- Prove a lower bound on the size of an Independent Set in a graph using the probabilistic method.