

Algorithms SV worksheet 3

Bálint Molnár

August 30, 2025

1 Trees

1. Prove that, in a binary search tree, if node n has two children, then its successor has no left child.
2. Prove that, if a key is not in a bottom node, its successor, if it exists, must be.
3. Write pseudocode for insertion and deletion to a red-black tree (without converting to a B-tree)
4. What is the time complexity of merging two binary heaps?

2 Hash Tables

1. Make a hash table with 8 slots and insert into it the following values: 15, 23, 12, 20, 19, 8, 7, 17, 10, 11.
Use the hash function $h(k) = (k \bmod 10) \bmod 8$, and resolve collisions by chaining
2. How can you handle deletions from an open addressing table? What are the problems of the obvious naïve approach?
3. One idea to avoid handling collisions is to make the hash table large enough so that the chances of having a collision are negligible. Explain why it is not feasible.
Hint: [REDACTED]
4. Find another use of hash functions in Computer Science/Software Engineering. Explain what properties of hash functions are required for this use case, and why you also need them for hash tables.

3 Exam Questions

1. <https://www.cl.cam.ac.uk/teaching/exams/pastpapers/y2008p10q9.pdf>
2. <https://www.cl.cam.ac.uk/teaching/exams/pastpapers/y2008p1q11.pdf>