

Exercise Set Week 4

1. Instead of binary heaps, suppose you had to implement ternary heaps (i.e., where each node has up to three children). Explain how you would implement such heaps using arrays, and how you can determine child and parent pointers. What are the advantages/disadvantages of ternary heaps over binary heaps?
2. Given a heap and a number k , design an efficient algorithm that outputs the top- k largest element from the heap. What is the running time of the algorithm? Can you design an algorithm that runs faster than $O(k \log n)$?
3. Can you use a binary search tree to simulate heap operations? What are the advantages/disadvantages of doing so?
4. Design an algorithm to convert a given binary search tree into a heap efficiently. What is the running time of your algorithm?

Exercise Set 4

1. Instead of binary heaps, suppose you had to implement ternary heaps (i.e., where each node has up to three children). Explain how you would implement such heaps using arrays, and how you can determine child and parent pointers. What are the advantages/disadvantages of ternary heaps over binary heaps?
2. Given a heap and a number k , design an efficient algorithm that outputs the top- k largest element from the heap. What is the running time of the algorithm? Can you design an algorithm that runs faster than $O(k \log n)$?
3. Can you use a binary search tree to simulate heap operations? What are the advantages/disadvantages of doing so?
4. Design an algorithm to convert a given binary search tree into a heap efficiently. What is the running time of your algorithm?