

Data Structures and Algorithms

Exercises Week 6

1. Define the Heap data structure; say what type of structure it is, how it is filled, how the entries are ordered.
2. Show construction of a heap as objects with the following keys are inserted, one by one, to an initially empty heap. You can use <https://visualgo.net/en/heap> to help if you wish.

23, 45, 12, 84, 39, 17, 10

At each step in the construction show the heap (as a tree) with an explanation of the action or operation performed.

3. Describe how entries are removed, in key sequence, from a heap. Use your heap from Exercise 2 to illustrate the process.
4. Explain why the worst-case time to insert an entry or to remove an entry is $O(\log n)$ for a heap with n entries.
5. Explain the `heapBuild` operation. Illustrate your answer by showing each stage in the operation when the data of Exercise 2 is loaded, using `heapBuild`, into an initially empty heap.
6. Is the heap obtained in Exercise 5 the same as that from Exercise 2? Explain why that should be so?
7. Show the array representation of the heaps from Exercises 2 and 5.
8. Verify the equation, given in part 10 of the notes, for the worst-case run time for `heapBuild`. Complete the following table by calculating the maximum number of swaps for heaps with 1, 3, 7, 15, and 31 entries.

heap size n	tree height h	max. no. swaps	$n - h - 1$
1	0	0	0
3	1		
7	2		
15	3		
31	4		

(See part 10 of the notes for the method of calculating the number of swaps for entries at each "level" of the tree.)

9. Show the sequences of keys obtained by each of the four traversals:

- (a) pre-order,
- (b) post-order,
- (c) in-order,
- (d) breadth-first

of the tree from Exercise 2.