

CSDS 233 Spring Session 10

SI Leader: Jakob Danninger

2/23/2022

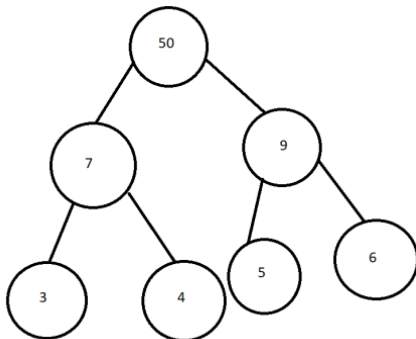
Disclosure: This is a supplement to class, not a replacement. This should not be your only study activity for exams, it should aid you in studying. I do not have access to the actual exam so questions here will differ from those on the exam.

Session Objectives:

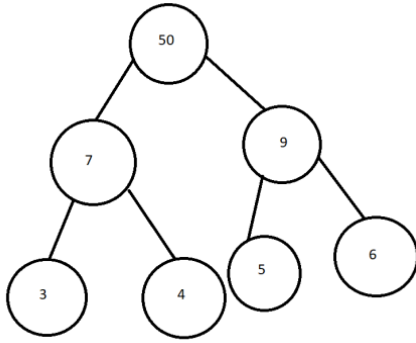
- 1) Basics of heaps, sift up, and sift down
- 2) Recognize hash functions
- 3) Understand how to use hashing to store data in an array

Questions

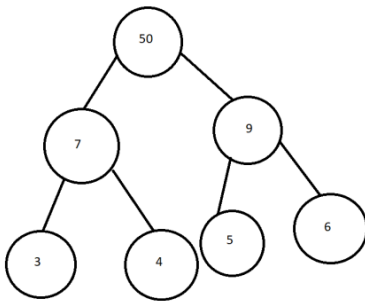
- 1) Write the array for the following heap



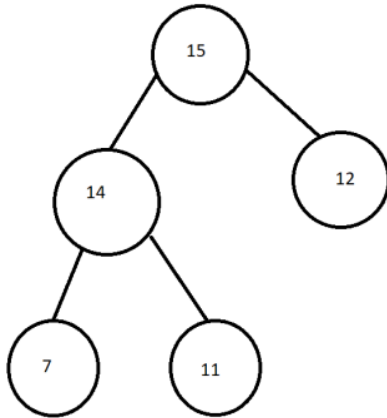
- 2) Remove 50 from the heap, show what value replaces 50 (before sifting), and then show what is the outcome after sifting. Write the finalized array of the new heap



3) Add 16 to the heap and sift to reorganize it



4) Add 16 to the heap and sift to reorganize it



5) For a node of index i , answer the following questions:

a. What is the index of i 's left child:

b. What is the index of i 's right child:

c. Parent node index:

6) Draw the max-on-top heap for the following operation: add 4, 3, 2, 9, 6, 7, 5

STOP HERE: We will switch to hashes

1) What is a hash function?

2) What is load factor?

3) What is the load factor of this array:

| 0 | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| 1 | 3 | | 4 | | |

4) Define the following

a) Linear probing

b) Quadratic probing

c) chaining

5) Draw the hash table array as a result of adding 50, 700, 76, 85, 92, 73, 101 using linear probing. The array is of size 7 and the hash function is $\text{key} \% 7$

6) Draw the same array but using quadratic probing

7) Draw the hash table array of size 7 as a result of adding "Harold", "Erman", "Chris", "Misha", using hash function $h = \text{string length} \% 7$ and linear probing

8) What is the advantage of using hash tables with low load factors

9) What is an easy reliable way to make hash functions