Department of Computer Science
King Saud University

Course title: CSC-212 (Data Structures)
Semester: Summer, 2007
Instructor: Dr. Muhammad Hussain

Final Exam

Question 1. (20)
A departmental store (e.g. Panda) maintains a record of the customers who visit and buy something from the store in a sorted order, sorted on the last name of a customer. In addition to the name of a customer, his telephone number, and the number of his visits are kept in his record. When a customer visits the store the first time, he is inserted in the record in the sorted order. On his other visits to the store, his 'number of visits' is incremented by one. When the 'number of visits' for a customer reaches 20, his name is added to the list of those people who get 10% discount on everything what they buy. When the 'number of visits' for a customer reaches 30, his name is deleted from the record of customers. All these operations must be executed efficiently. Suggest a suitable ADT for this purpose and describe it as follows:

(a) Give its graphical representation.
(b) Give its representation as C++/JAVA class declaration.
(c) Specify various operations.

Question 2. (20)

(a) Express the following time functions in big-O notation, giving C and n_0
  (i) $4n^3 + 2n^2 + 3$ (ii) $(n^2 + 1)(n + 1)$
(b) Give implementation of an operation to delete the previous node in a list using specification for ADT List (Link List).

  delete_prev (l : List)

  Precondition/Requires: The list l should not be empty or the current element should not be the first element.

  Results/Actions: The element previous to the current element is deleted from the list and the element previous to the deleted element is made the current element. If the deleted element is the first element, then its next element is made current.

Question 3. (20)

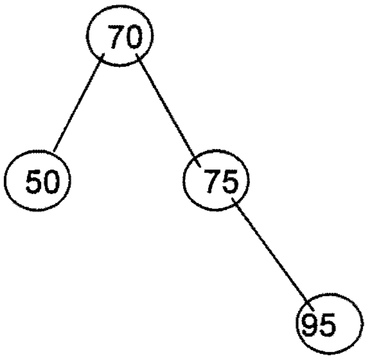
(a) If you have to perform insert, delete, find and find_min operations on set of data than can fit into main memory, then which of the following ADTs is most suitable:

(i) Hash table

(ii) AVL Tree

(iii) B+-Tree

(b) Is the following tree an AVL tree? If not, convert it into an AVL tree.



(i) Insert the node 125 into the above tree, if modification is needed to convert it into AVL tree after inserting 125, tell the name of the modification and draw the tree after modification.

(ii) Insert the node 85 into the above tree, if modification is needed to convert it into AVL tree after inserting 85, tell the name of the modification and draw the tree after modification.

Question 4. (20)

Using the hash function **H**(key) = key mod 7, and

(a) linear rehashing

(b) Separate chaining

insert the integers: 89, 33, 56, 9, 79, 26 in the hash table of TableSize=13

After inserting the values, determine

(i) The load factor

(ii) The average number of probes needed to find a value that is in the table

(iii) The number of probes to find 15

Comment on which collision detection method is better in respect of memory usage.

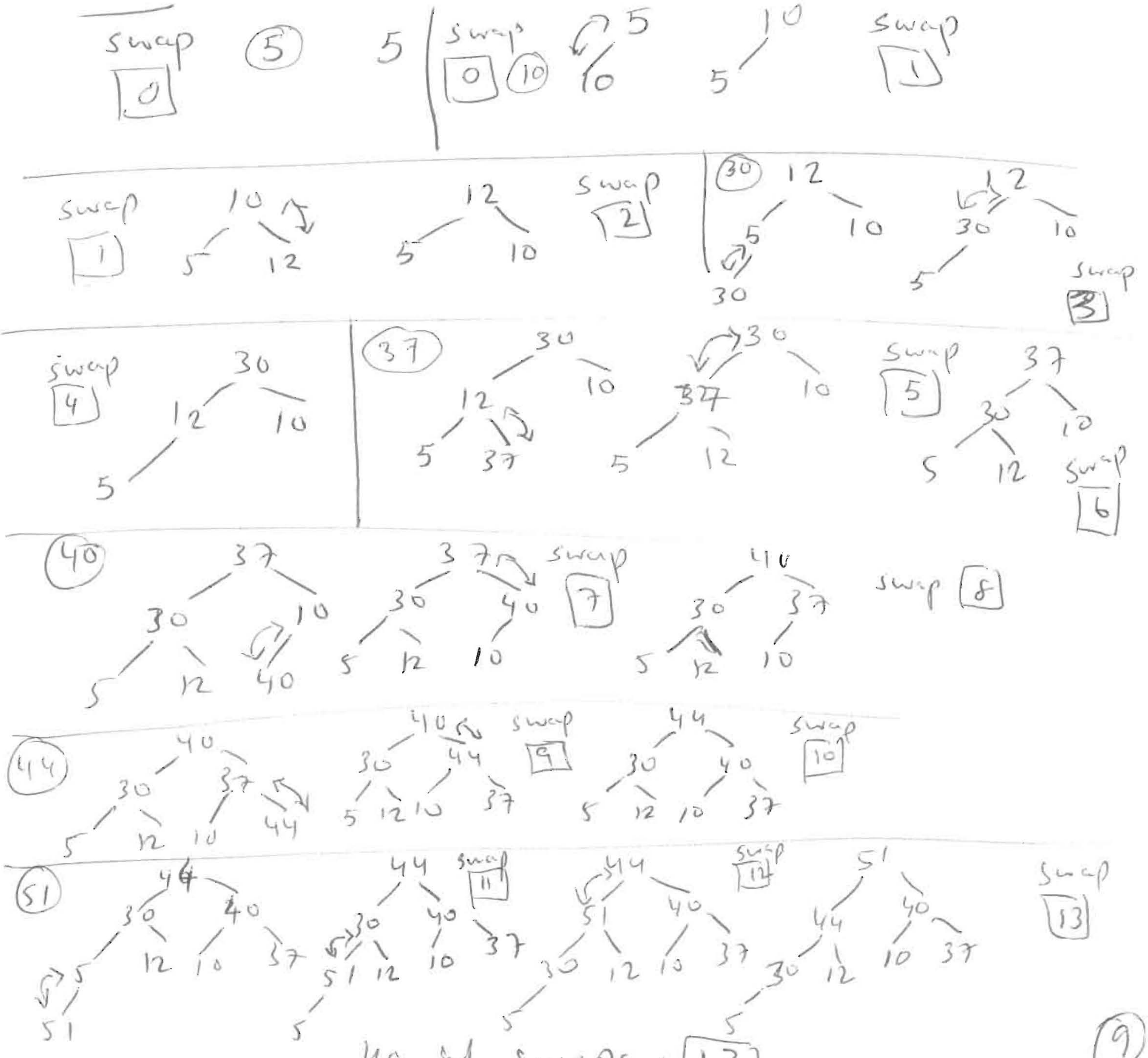
Question 5. (20)

(a) Construct a max heap from the sequence of integers: 5, 10, 12, 30, 37, 40, 44, and 50. How many swap operations are required to construct the heap.

(b) Enqueue the following elements with the given priority into a priority queue implemented as a heap: 10, 12, 1, 14, 6, 5, 8, 15, 3, 9, 7, 4, 11, 13, and 2. Assume that a higher number indicates higher priority.

(c) In the priority queue obtained in part (b) perform two Dequeue (Serve) operations and show the queue after each operation.

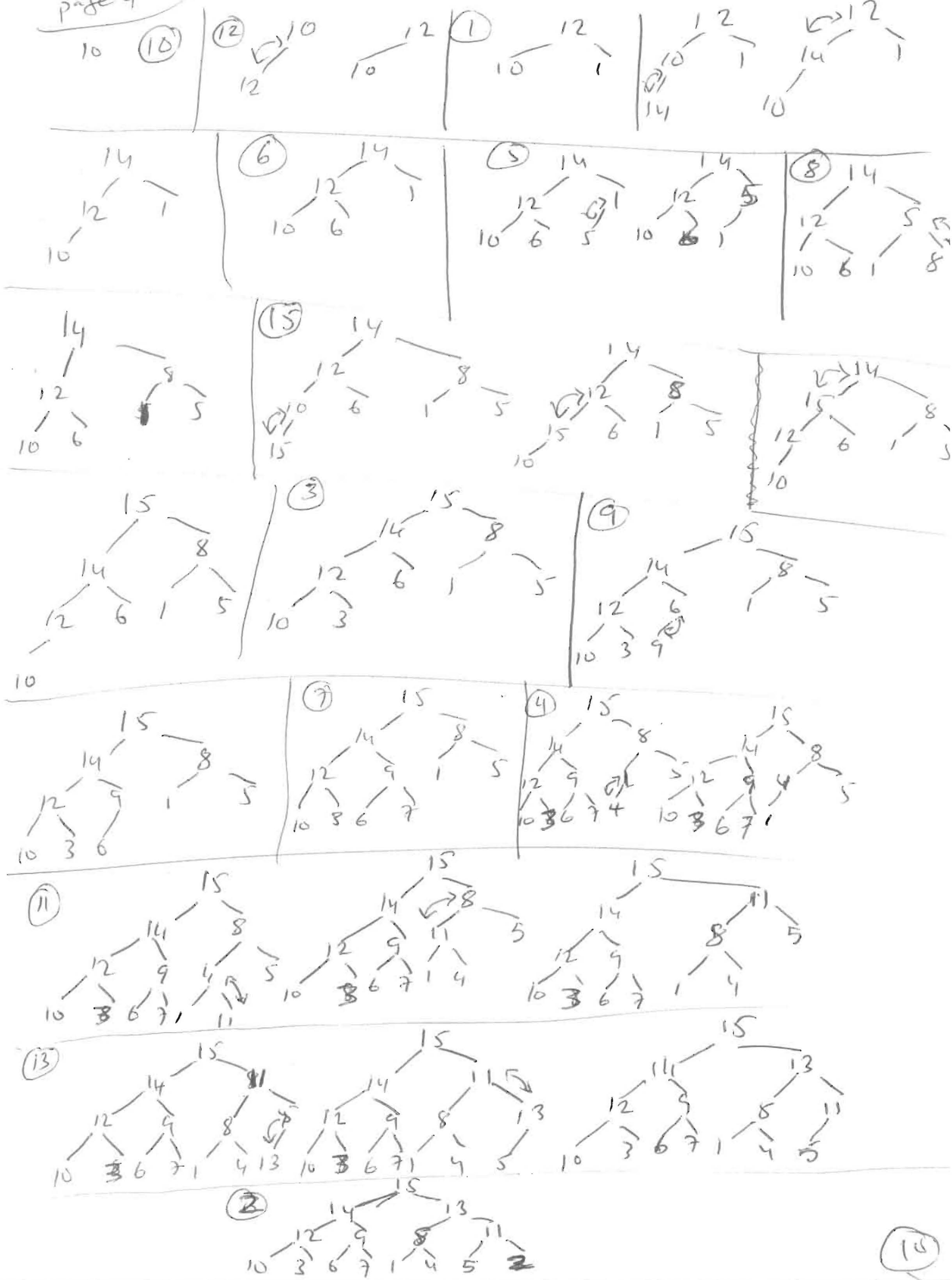
Q5-a max heap 5, 10, 12, 30, 37, 40, 44, 51.



no of swaps : 13

Q5-b
page 4

Enqueue 10, 12, 1, 14, 6, 5, 8, 15, 3, 9, 7, 4, 11, 13, 2



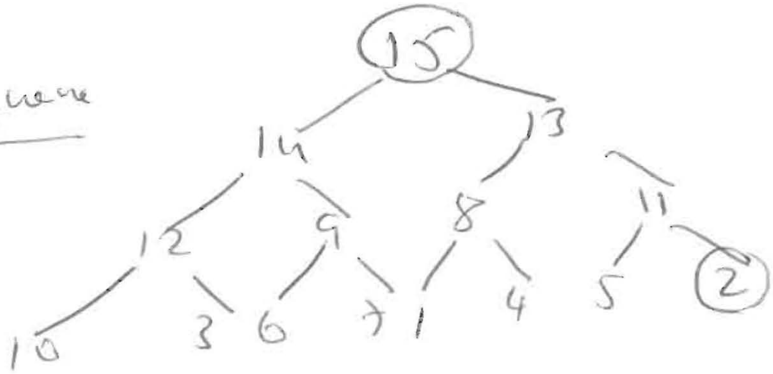
14

#5-c

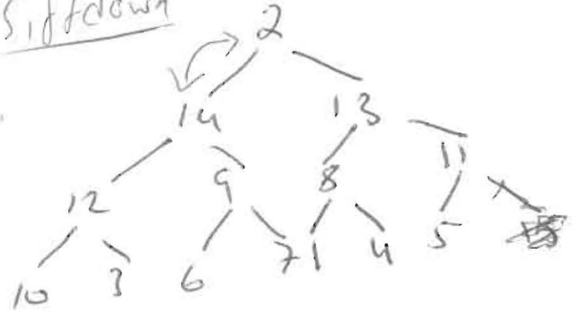
page 4

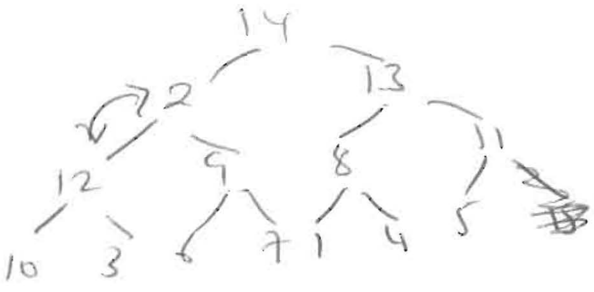
Two serve (dequeue).

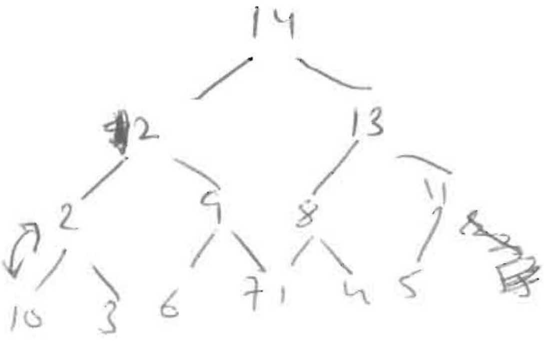
first dequeue

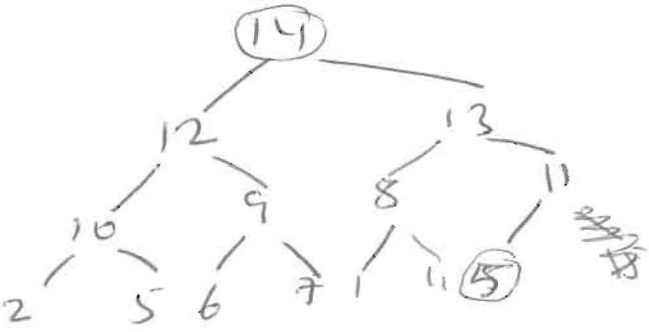


Siftdown



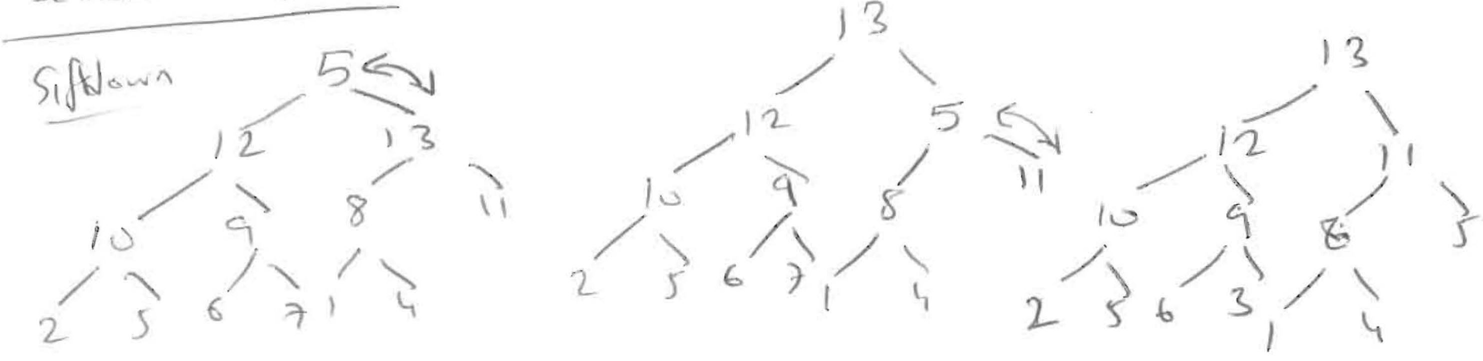






Second dequeu

Siftdown



done