LM Data Structures, Algorithms, and Databases (34140, 34141, 34139, 36989)

Exercise Sheet Solution

Week 07 (Stack, Queue, Heaps)

Q1. [Stack] Describe the output of the following series of stack operations: push(5), push(3), pop(), push(2), push(8), pop(), pop(), push(9), push(1), pop(), push(7), push(6), pop(), pop(), push(4), pop(), pop()

Answer

	Stack (LIFO)
push(5)	[5]
push(3)	[5, 3]
pop()	[5]
push(2)	[5, 2]
push(8)	[5, 2, 8]
pop()	[5, 2]
pop()	[5]
push(9)	[5, 9]
push(1)	[5, 9, 1]
pop()	[5, 9]
push(7)	[5, 9, 7]
push(6)	[5, 9, 7, 6]
pop()	[5, 9, 7]
pop()	[5, 9]
push(4)	[5, 9, 4]
pop()	[5, 9]
pop()	[5]

Q2. [Queue] Describe the output for the following sequence of queue operations: enqueue(5), enqueue(3), dequeue(), enqueue(2), enqueue(8), dequeue(), dequeue(), enqueue(9), enqueue(1), dequeue(), enqueue(7), enqueue(6), dequeue(), dequeue(), enqueue(4), dequeue(), dequeue().

Answer

	Queue (FIFO)
enqueue(5)	[5]
enqueue(3)	[5, 3]
dequeue()	[3]
enqueue(2)	[3, 2]
enqueue(8)	[3, 2, 8]
dequeue()	[2, 8]
dequeue()	[8]
enqueue(9)	[8, 9]
enqueue(1)	[8, 9, 1]
dequeue()	[9, 1]

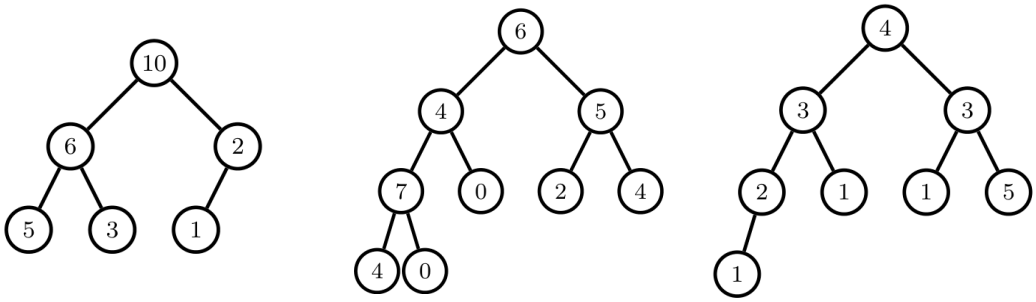
enqueue(7)	[9, 1, 7]
enqueue(6)	[9, 1, 7, 6]
dequeue()	[1, 7, 6]
dequeue()	[7, 6]
enqueue(4)	[7, 6, 4]
dequeue()	[6, 4]
dequeue()	[4]

Q3. [Mod and div] Calculate the following:

1. 20 div 3 = ? 20 mod 3 = ?
2. 21 div 7 = ? 21 mod 7 = ?

Answers: (1) 6 and 2, (2) 3 and 0

Q4. Decide which of the trees are max heap.



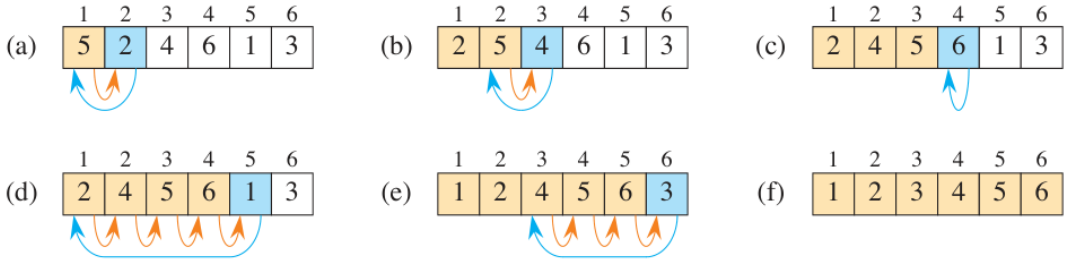
(i) (ii) (iii)

Answer: (i)

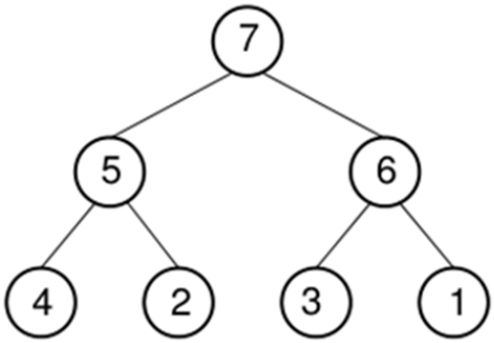
Q5. Show step by step working of Insertion sort algorithm for the given array A

(5, 2, 4, 6, 1, 3)

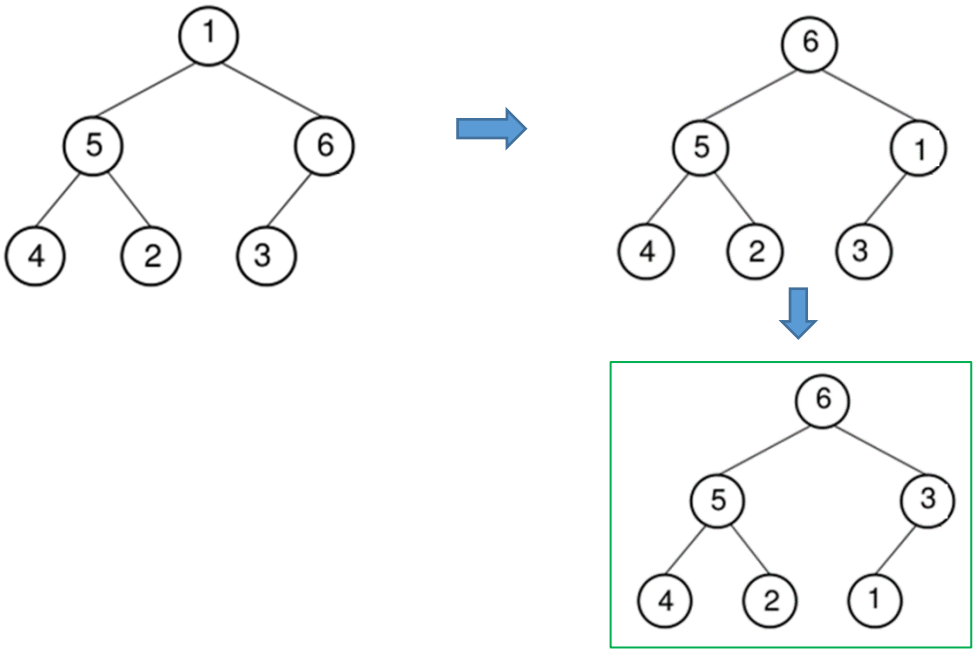
Answer



Q6. Show the heap that results from deleting the maximum value from the max-heap of following Figure.



Answer: We can move the element in the last position in the heap (the current last element in the array) to the root position and then order the heap.



Q7. When writing code or text, it's important to have parentheses balanced and properly nested. For example:

- The string (((())())() has properly nested pairs of parentheses.
- The strings)()(and ()) do not have properly nested pairs of parentheses.

Identify the data structure you will use and write an algorithm that returns true if a string contains properly nested and balanced parentheses, and false if otherwise.

Answer

```
function checkParenthesesBalance(string):
    stack = empty stack

    for each character in string:
        if character is '(':
            push character onto stack
        else if character is ')':
            if stack is empty:
                return false  # Unmatched closing parenthesis
            else:
                pop from stack

    if stack is empty:
        return true  # All parentheses are matched
    else:
        return false  # Unmatched opening parenthesis
```

https://www.geeksforgeeks.org/check-for-balanced-parentheses-in-python/

Q8. Let Q be a non-empty queue, and let S be an empty stack. Using only the stack and queue functions and a single element variable X, write an algorithm to reverse the order of the elements in Q.

Answer:

```
def reverse(Queue Q, Stack S):
#Dequeue elements from the queue Q and pushes them onto the stack S
    while not Q.isEmpty():
        X = Q.dequeue()
        S.push(X)

#Once all elements are moved to the stack, pops elements from the stack S and
enqueues them back into the queue Q
    while not S.isEmpty():
        X = S.pop()
        Q.enqueue(X)
```