Exercise 1 Stacks and queues

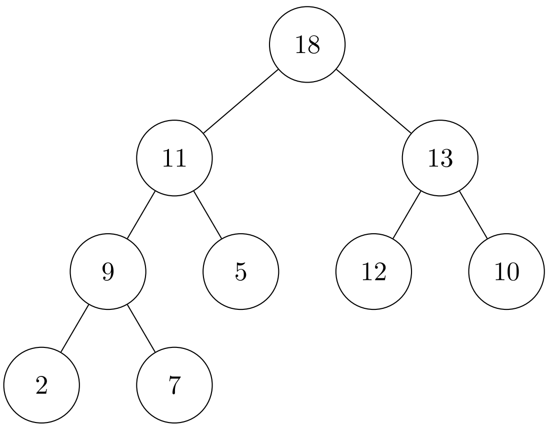
1. Using Figure 10.2 as a model, illustrate the result of each operation in the sequence PUSH(S, 4), PUSH(S, 1), PUSH(S, 3), POP(S), PUSH(S, 8) and POP(S) on an initially empty stack S stored in array $S[1:6]$ (CLRS 10.1-2)
2. Using Figure 10.3 as a model, illustrate the result of each operation in the sequence ENQUEUE(Q, 4), ENQUEUE(Q, 1), ENQUEUE(Q, 3), DEQUEUE(Q), ENQUEUE(Q, 8) and DEQUEUE(Q) on an initially empty queue Q stored in array $Q[1:6]$ (CLRS 10.1-4)
3. Show how to implement a queue using two stacks. Analyze the running time of the queue operations. (CLRS 10.1-7)

Exercise 2 Linked lists

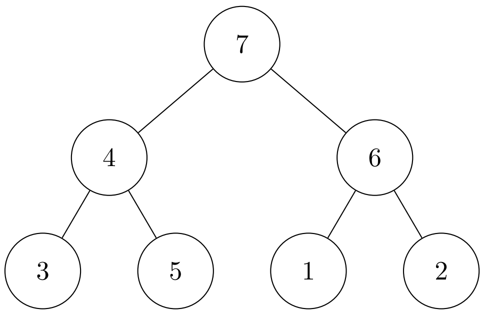
1. Explain why the dynamic-set operation INSERT on a *singly* linked list can be implemented in $O(1)$ time, but the worst-case time for DELETE is $\Theta(n)$. (CLRS 10.2-1)
2. Implement a stack using a *singly* linked list. The operations PUSH POP should still take $O(1)$ time. Do you need to add any attributes to the list? (CLRS 10.2-2)
3. Give a $\Theta(n)$-time nonrecursive procedure that reverses a singly linked list of n elements. The procedure should use no more than constant storage beyond that needed for the list itself. (CLRS 10.2-5)

Exercise 3 Heaps

1. What are the minimum and maximum numbers of elements in a heap of heigh h? (CLRS 6.1-1)
2. Is an array that is in sorted order a min-heap? (CLRS 6.1-5)
3. Are the trees below valid max-heaps? Why / why not?
4. Is the array with values $\langle 33, 19, 20, 15, 13, 10, 2, 13, 16, 12\rangle$ a max-heap? (CLRS 6.1-7)
5. Show that, with the array representation for storing an n-element heap, the leaves are the nodes indexed by $\lfloor n/2\rfloor + 1$, $\lfloor n/2\rfloor + 2$, ..., n. (CLRS 6.1-8)



(a)



(b)