מבחן במבנה נתונים - דוגמא - מספר 2 - תשובות

Q1. (20%)
Describe a way to implement a queue, using two stacks. Explain how to implement Enqueue and Dequeue. No need to write code in your answer, just describe the algorithm.
The amortized running time of Enqueue and Dequeue should be $O(1)$.

Enqueue is implemented by pushing to s1.
Dequeue is implemented by popping from s2. If s2 is empty, the content of s1 is transferred to s2 before the pop.

Q2. (20%)
t is the root of a binary tree. Write an algorithm (in pseudocode) that returns the height of the tree if the tree is balanced and -1 if it is not balanced. The running time of the algorithm is $O(n)$, where *n* is the number of nodes in the tree.

```
isBalanced(t)
if t=nil
then return 0
l←isbalanced(t.left)
r←isBalanced(t.right)
if r=-1 or l=-1 or |l-r|>1
then return -1
else if r>l
       then return r+1
       else return l+1
```

Q3. (20%)

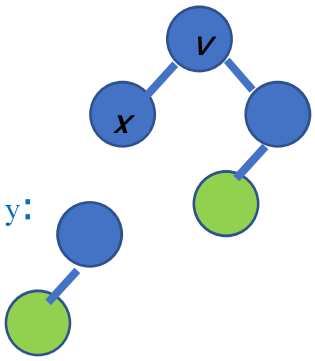
Let *p* be a node in a binary tree. Write, in pseudocode, an algorithm that finds the successor of *p* in pre-order-traversal.

```
preOrderSucc(p)
if p.left≠nil
then return p.left
if p.right≠nil
then return p.right
while p≠nil do
     if p.parnet≠nil and p.parent.left=p and
p.parent.right≠nil
     then return p.parent.right
     else p←p.parent
return nil
```

Q4. (20%)

Is the function *remove* from a binary search tree a commutative function?
In other words: does removing node *x* and then removing node *y* from a binary search tree always gives the same tree we would get from removing *y* first and the removing *x*?
Prove your answer.

No.



Removing x and then y:

Removing y and then x:

Q5. (20%)

Write an **efficient** C function that searches for x in an array a[] - that represents a minimum heap. The size of the array is n.

If x is in the array, the function should return 1. If x is not in the array, the function should return 0.

The header of the function is:

```
int search(int[] a, int n, int x)
```

What is the asymptotic running time of function *search*?

If a[] would represent a maximum heap (instead of a minimum heap), what changes would you need to make in function search?

```
int search(int[] a, int n, int x)
{
     for {int i=0;i<n;i++)
          if (a[i]==x)
               return 1;
     return 0;
}
```

The running time is O(n) and no change is needed if the heap is a maximum heap.