McGill | School of Continuing Studies École d'éducation permanente

Assignment 2

Due date: July 19th, 2021

`

At 11:55am

(Group of 3)

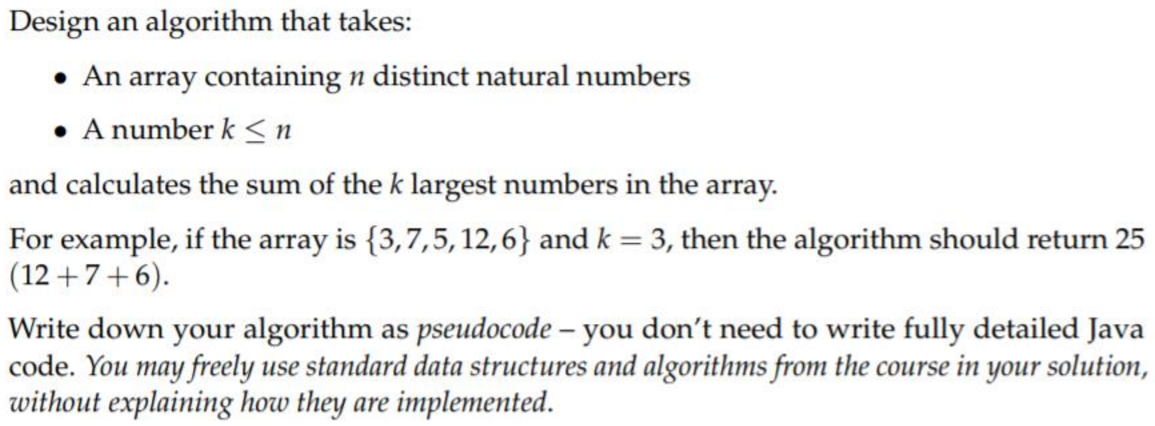
General instructions:

This is not an individual Assignment. Team of 3. Only **one team member** is required to submit the project. Please make sure to mention the name for all 3 members.

Very important: If there are more than one submission per group, I will choose one submission and that submission will be graded.

No late submission is accepted.

Exercise – 1



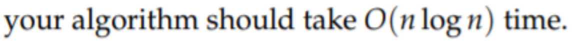
Design an algorithm that takes:

- An array containing n distinct natural numbers
- A number $k \leq n$

and calculates the sum of the k largest numbers in the array.

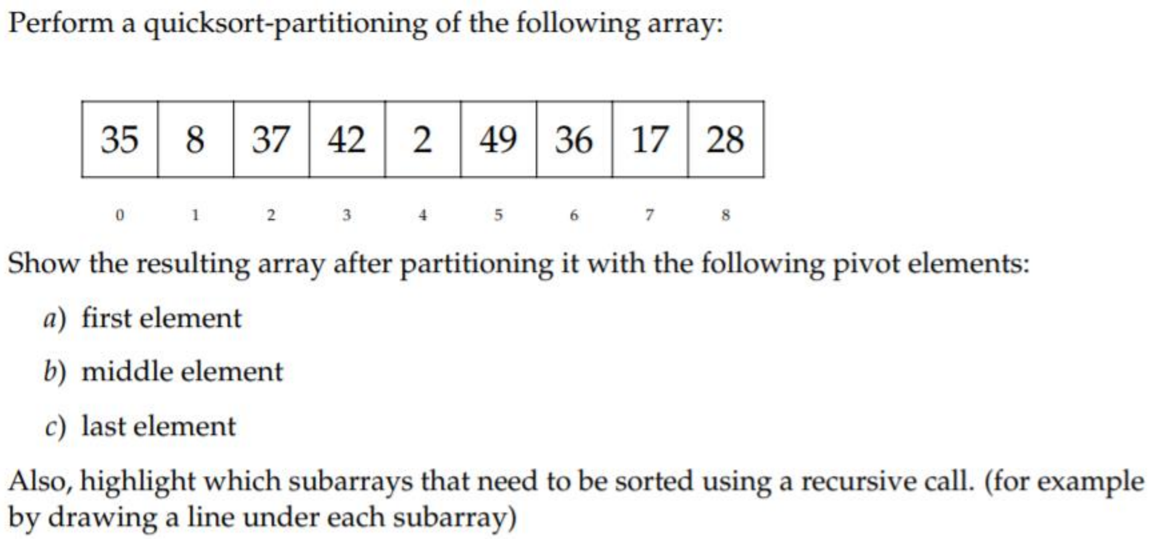
For example, if the array is $\{3,7,5,12,6\}$ and $k = 3$, then the algorithm should return 25 $(12+7+6)$.

Write down your algorithm as *pseudocode* – you don't need to write fully detailed Java code. *You may freely use standard data structures and algorithms from the course in your solution, without explaining how they are implemented.*



your algorithm should take $O(n \log n)$ time.

Exercise – 2



Perform a quicksort-partitioning of the following array:

35	8	37	42	2	49	36	17	28
0	1	2	3	4	5	6	7	8

Show the resulting array after partitioning it with the following pivot elements:

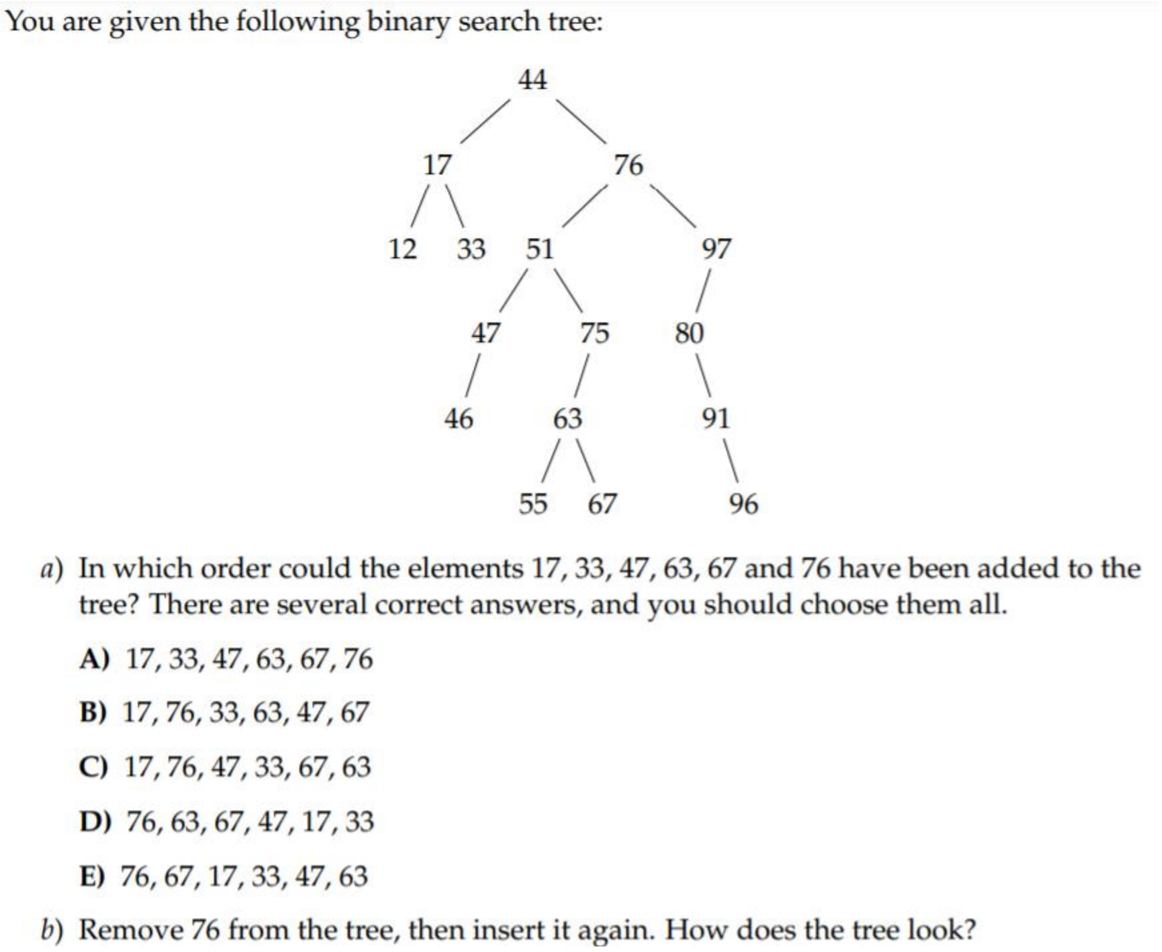
$a)$ first element

$b)$ middle element

$c)$ last element

Also, highlight which subarrays that need to be sorted using a recursive call. (for example by drawing a line under each subarray)

Exercise - 3



You are given the following binary search tree:

44
17
76
12
33
51
97
47
75
80
46
63
91
55
67
96

a) In which order could the elements 17, 33, 47, 63, 67 and 76 have been added to the tree? There are several correct answers, and you should choose them all.

A) 17, 33, 47, 63, 67, 76

B) 17, 76, 33, 63, 47, 67

C) 17, 76, 47, 33, 67, 63

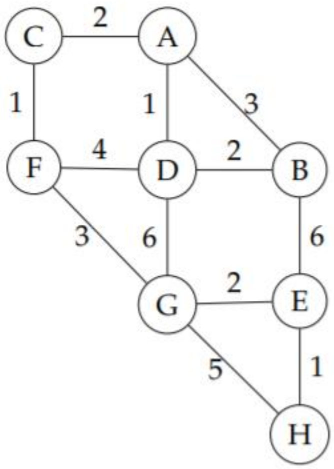
D) 76, 63, 67, 47, 17, 33

E) 76, 67, 17, 33, 47, 63

b) Remove 76 from the tree, then insert it again. How does the tree look?

Exercise – 4

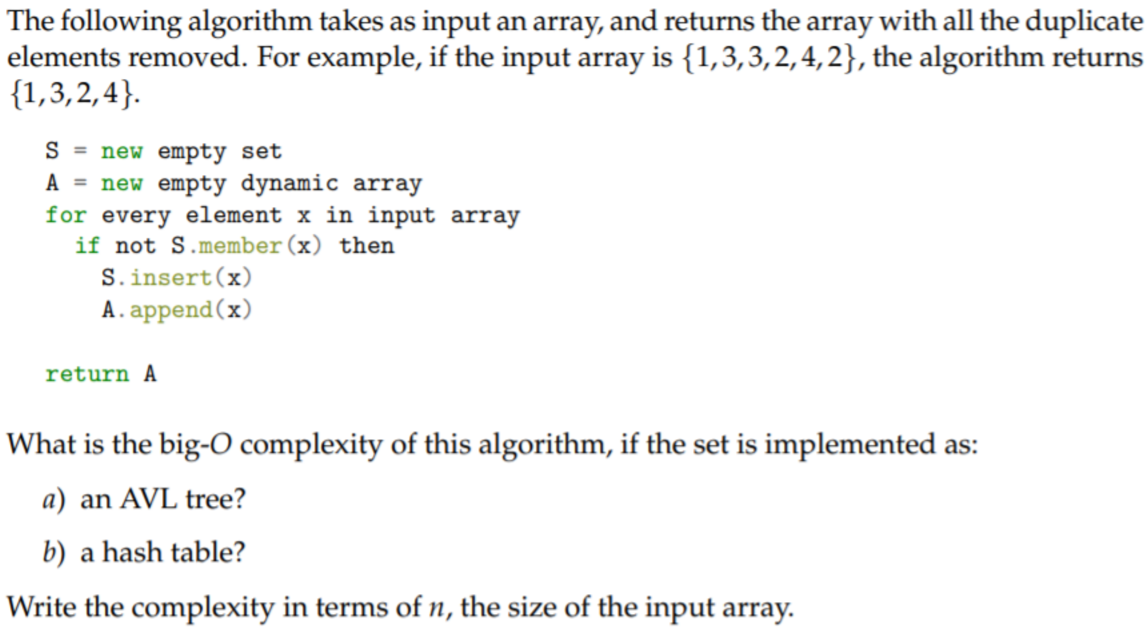
You are given the following undirected weighted graph:



Compute a minimal spanning tree for the following graph by manually performing Prim’s algorithm using D as starting node.

Your answer should be the set of edges which are members of the spanning tree you have computed. The edges should be listed in the order they are added as Prim’s algorithm is executed. Refer to each edge by the labels of the two nodes that it connects, e.g. DF for the edge between nodes D and F.

Exercise – 5



The following algorithm takes as input an array, and returns the array with all the duplicate elements removed. For example, if the input array is $\{1, 3, 3, 2, 4, 2\}$, the algorithm returns $\{1, 3, 2, 4\}$.

```
S = new empty set
A = new empty dynamic array
for every element x in input array
  if not S.member(x) then
    S.insert(x)
    A.append(x)

return A
```

What is the big-O complexity of this algorithm, if the set is implemented as:

a) an AVL tree?

b) a hash table?

Write the complexity in terms of n, the size of the input array.