@author Rufina Talalaeva

3.1. a)

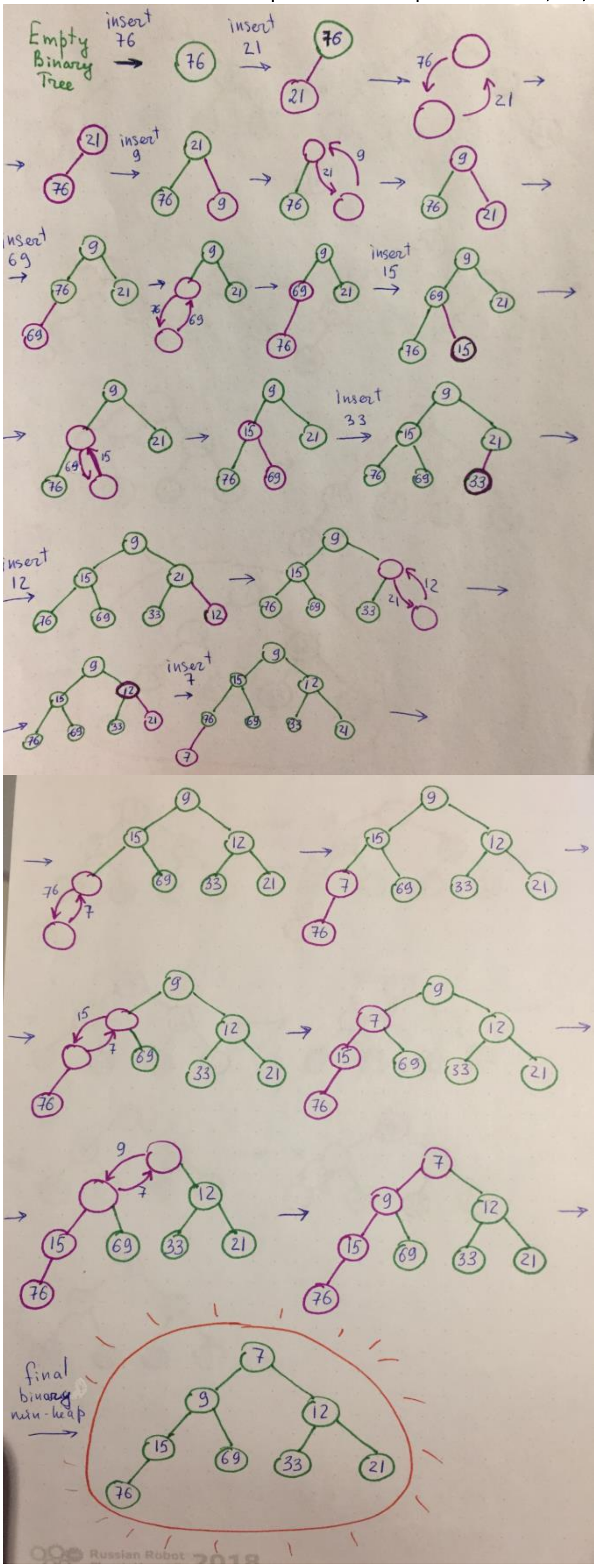
<u>Algorithm of insertion is taken from the lecture:</u>

1. Add element to the bottom level

2. Compare the added element with its parent; if they are in correct order, stop

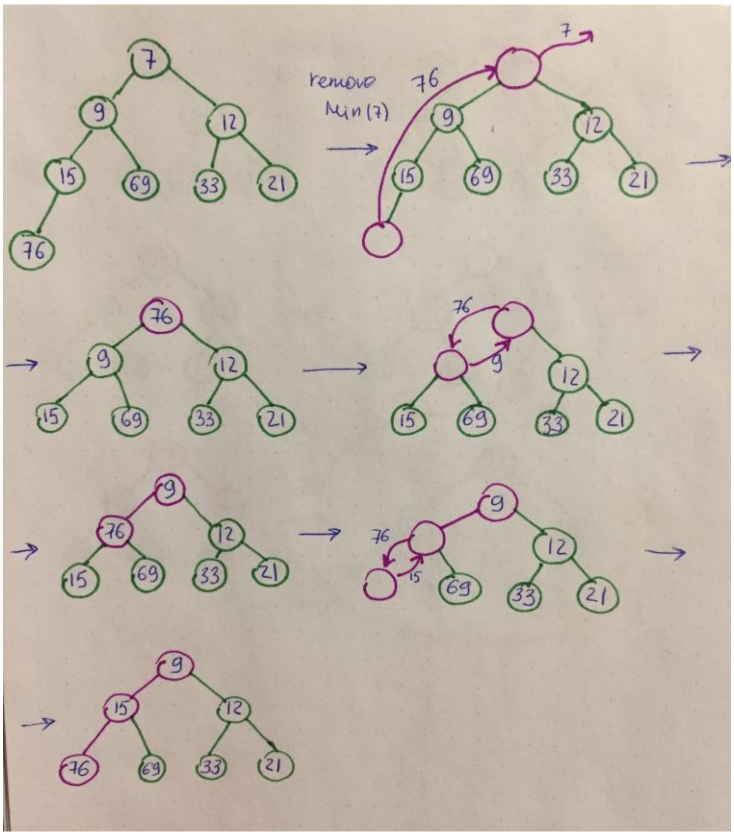
3. If not, swap the element with its parent and return to previous step

Pictures below show all steps in insertion process of 76, 21, 9, 69, 15, 33, 12, 7.

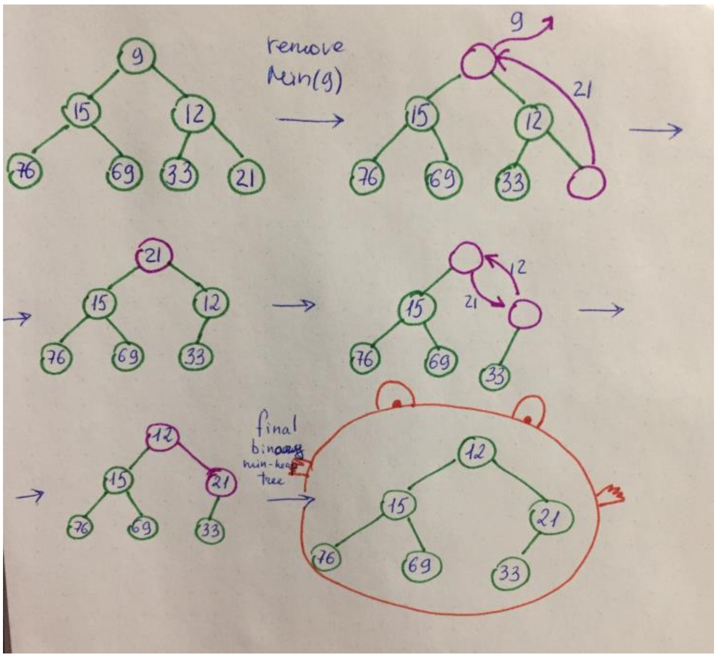


b)*<u>Algorithm of insertion is taken from the lecture:</u>*
1. Replace root with the last element on the bottom level
2. Compare the swapped element with the smaller child; if they are in correct order, stop
3. If not, swap the element with the child and return to previous step
Pictures below show all steps in deletion process of two minima.

After first removal:



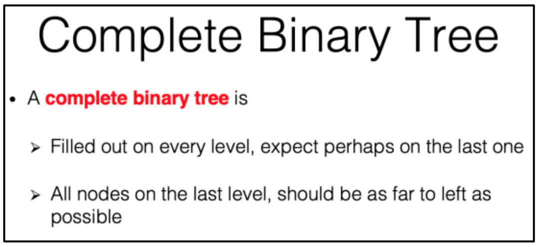
After second removal:



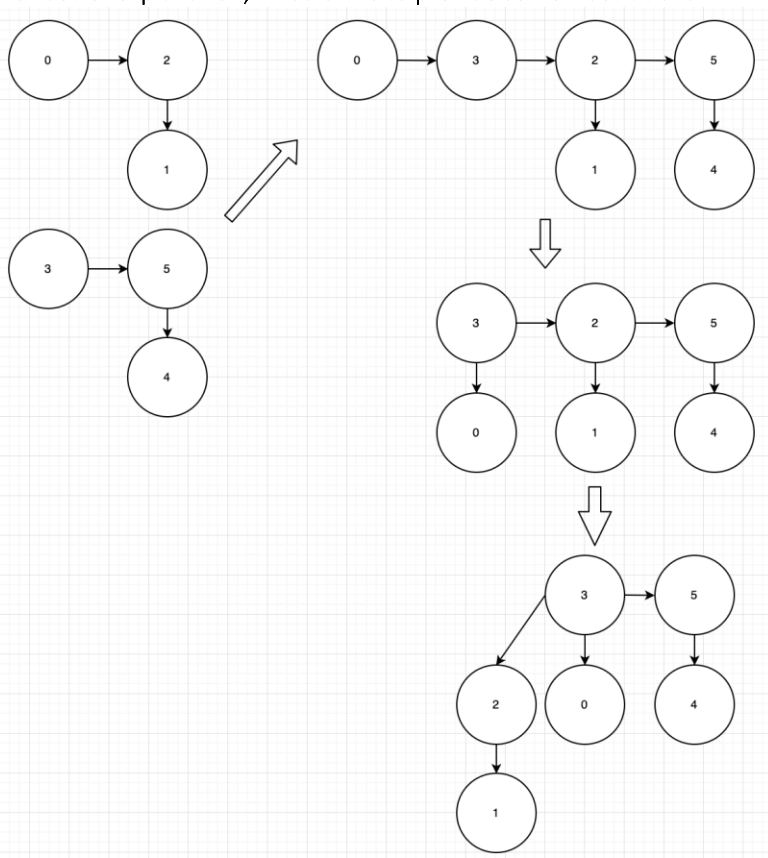
c)**Answer: True.**
Binary heap is a complete Binary Tree. According to the slide 16, lecture 9, Binary Tree has *the maximum amount* of elements *until the level (d-1)* (filled out on every level, except last one, in our case, level d). *The minimum* amount of elements *on the last level d is 1*. => We will count all nodes on the previous depth (on each level n there are 2^n nodes, because it is *Binary Heap*) and add 1 from the last depth, it will be the minimum amount of elements in Binary Tree with depth d:

$$\sum_{n=0}^{n=d-1} 2^n + 1 = \frac{1*(2^d - 1)}{2-1} + 1 = 2^d - 1 + 1 = 2^d$$

As we see, in a Binary Heap of depth d has at least 2^d nodes. => Answer: True.



3.2
a) We are merging trees of both binomial heaps after the arranging them in the increasing order of degrees. In the initial heaps there was at most one tree of certain degree. But when we are merging them there could be situation that there are three roots of the same degree in a row. Let merge first two nodes in one tree, and leave third node as it is. **BUT!** This contradicts to the definition of binomial heap <- we will get node of higher degree before the node of lower degree. That's why if there is a sequence of three binomial trees with the same degree we march first instead of merging it to the next tree.
For better explanation, I would like to provide some illustrations.



The last heap is not binomial heap by definition! Thus, we can't merge first two nodes.

b) **Answer: $\log(n+1) - 1$.**
Let **d** will be the height of a Binary Heap storing **n** keys.
We saw in the previous subtask the minimum amount of elements in the Binary Heap of depth d.
Let's see what is the maximum amount of elements, it is when we have all nodes on each level, even on the last:

$$\sum_{i=0}^{i=d} 2^i = \frac{1 * (2^{d+1} - 1)}{2 - 1} = 2^{d+1} - 1$$

So, $n = 2^{d+1} - 1 \Leftrightarrow \frac{n+1}{2} = 2^d \Leftrightarrow d = \log(n+1) - 1$. Thus, the maximum height of a Binary Heap with n nodes is $\log(n+1) - 1$.

3.3

-in-degree, out-degree and degree of c & f.

	c	f
in-degree	1	2
out-degree	1	2
degree	2	4

-The definition of strongly connected graph is from Discrete Mathematics and Its Applications 7edition, Kenneth H. Rosen(p. 685). Our graph is not strongly connected, because there is no path from b to a.

> A directed graph is *strongly connected* if there is a path from a to b and from b to a whenever a and b are vertices in the graph.

There are 3 connected components:

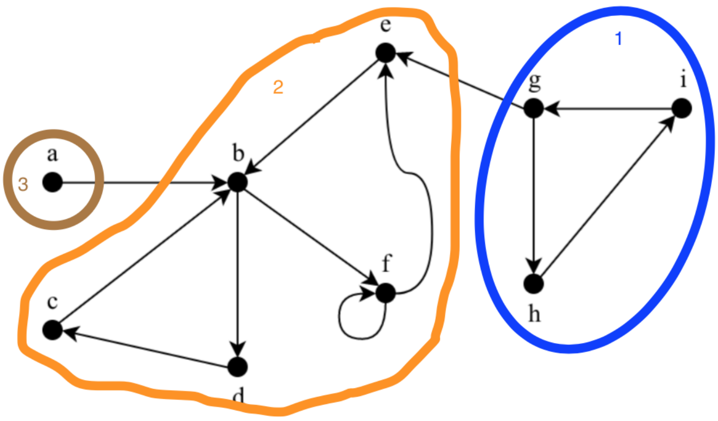


Figure 1: Graph Γ.

-Adjacency matrix(rows are initial cities, columns are destiny cities).

	a	**b**	**c**	**d**	**e**	**f**	**g**	**h**	**i**
a	0	1	0	0	0	0	0	0	0
b	0	0	0	1	0	1	0	0	0
c	0	1	0	0	0	0	0	0	0
d	0	0	1	0	0	0	0	0	0
e	0	1	0	0	0	0	0	0	0
f	0	0	0	0	1	1	0	0	0
g	0	0	0	0	1	0	0	1	0
h	0	0	0	0	0	0	0	0	1
i	0	0	0	0	0	0	1	0	0

Any undirected graph has an even number of vertices of odd degree by using the Handshaking Theorem.

<u>By contradiction:</u>

Let us suppose that there is a graph with odd number of vertices of odd degree.

Then let V1 will be the set of even degree vertices and V2 will be the set of odd degree vertices in an undirected graph G(V, E) with m edges. Then by Handshaking Theorem $2m = \sum_{v \in V} \deg(v) = \sum_{v \in V1} \deg(v) + \sum_{v \in V2} \deg(v)$.

Because deg(v)-even for $v \in V1$, the first term in the right-hand side of the equality is even. By our assumption, the second term in the sum is odd. But! The sum of 2 terms on the left-hand side of the equality is even(2m) => Contradiction! => There is no such undirected graph with odd number of vertices of odd degree. => <u>Any undirected graph has an even number of vertices of odd degree.</u>

<u>By induction:</u>

1. Base case:

In any undirected graph for 0 vertices there is 0 number of vertices of odd degree.

2. Assumption:

Any undirected graph has an even number of vertices of odd degree.

3. Induction step:

If we will add an edge:

-Between two vertices of even degree -> The number of vertices of odd degree will increase by 2, and the number of vertices of even degree will decrease by 2. So, the number of vertices of odd degree will still remain even.

- Between two vertices of odd degree -> The number of vertices of odd degree will decrease by 2, and the number of vertices of even degree will increase by 2. So, the number of vertices of odd degree will still remain even.

- Between node of even degree and node of odd degree -> The number of vertices of odd degree and the number of vertices of even degree will not change. So, the number of vertices of odd degree will still remain even.

If we will add a vertex, it will have degree 0; So, only number of vertices of even degree changes, but not of odd degree. So, the number of vertices of odd degree will still remain even.