TA: Jade Cheng

ICS 241

Recitation Lecture Notes #10

October 30, 2009

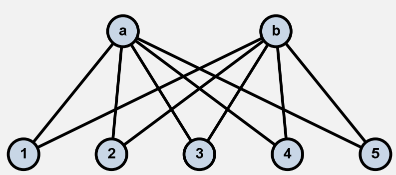
Recitation #10

Question: Can five houses be connected to two utilities without connection crossing? [Chapter 9.7 Review]

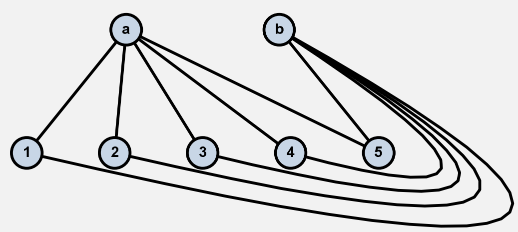
Answer: Yes. Construct a graph model for this problem, we have the following situation. Is $k_{5,2}$ a planar graph? According to Kuratowski's Theorem, A graph is nonplanar if and only if it contains a subgraph homeomorphic to $k_{3,3}$ or k_5. Apparently we don't have any such subgraph homeomorphic for $k_{5,2}$.

Question: Can you draw the model described above without any crossing. [Chapter 9.7 Review]

Answer: Let's use a, b to represent the utilities that the houses are going to share, and numeric numbers to represent the five houses. We are able to simply "drag" the edges and express the left hand side graph as the right hand side graph, which doesn't have any edges crossing each other.



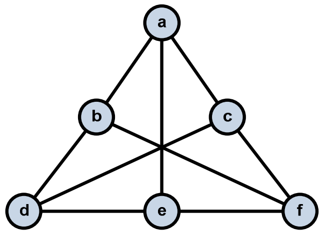
Original graph



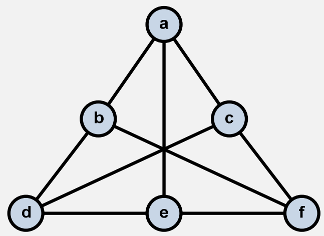
Graph without edge crossing

Question: determine whether the given graph is planar. [Chapter 9.7 Review]

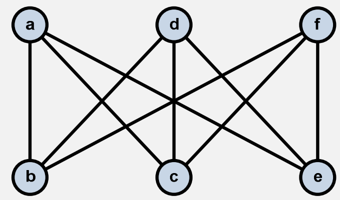
a.



Answer: No, this graph is not a planar. We can't represent it in 2-D without having edges crossing each other. In order to prove this, we can show that the given graph is actually $k_{3,3}$, which is proven to be an nonplanar in the text book Example 3. Also, we can apply Turatowski's Theorem to derive the same conclusion.

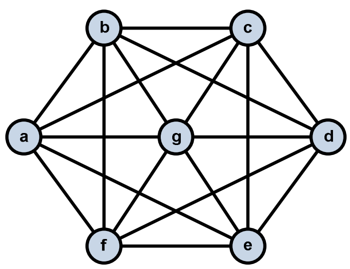


Original graph

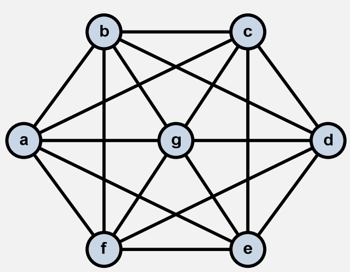


"Common" look of $k_{3,3}$

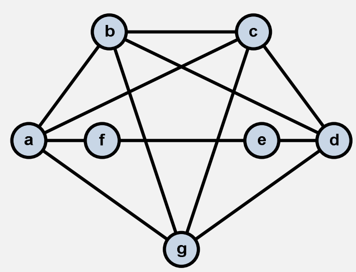
b.



Answer: No, this graph is not a planar. We can't represent it in 2-D without having edges crossing each other. In order to prove this, we can show that the given graph contains a subgraph homeomorphic to k_5, and then apply Turatowski's Theorem to derive the conclusion.



Original graph



A subgraph homeomorphic to k_5

Question: Suppose that a connected planar graph has six vertices, each of degree four. Into how many regions is the plane divided by a planar representation of this graph? [Chapter 9.8 Review]

Review: *<u>Euler's formula:</u>* Let G be a connected planar simple graph with e edges and v vertices. Let r be the number of regions in a planar representation of G. Then $r = e - v + 2$.

Answer: We should be able to plug in the values in the Euler's formula, if the number of vertices and the number of edges are given. We have $v = 6, degree(v_i) = 4$.

$$v = 6, degree(v_i) = 4$$

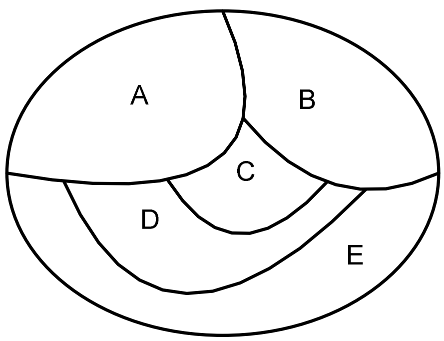
$$\Rightarrow e = \frac{v \times degree(v_i)}{2} = \frac{6 \times 4}{2} = 12$$

$$\Rightarrow r = e - v + 2 = 12 - 6 + 2 = 8\,.$$

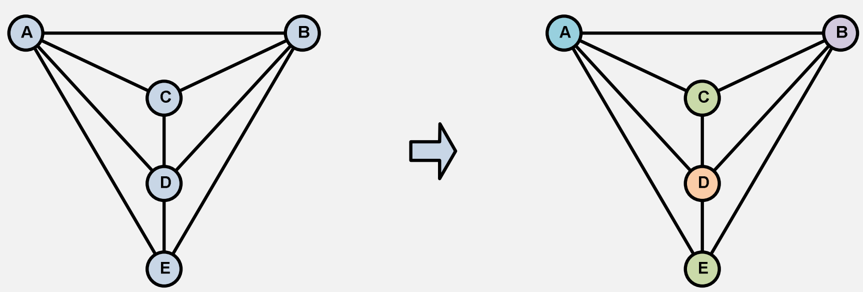
So the solution is 8 regions raised by this planar graph.

Question: Construct the dual graph for the map shown. Then find the number of colors needed to color the map so that no two adjacent regions have the same color. [Chapter 9.8 Review]

a.



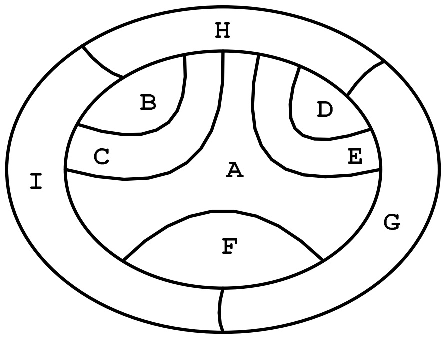
Answer: The number of vertices is the number of regions on the map. The number of edges is the number of sharing boarders on the map. So we obtain the following dual graph. And this graph can be colored using 4 colors.



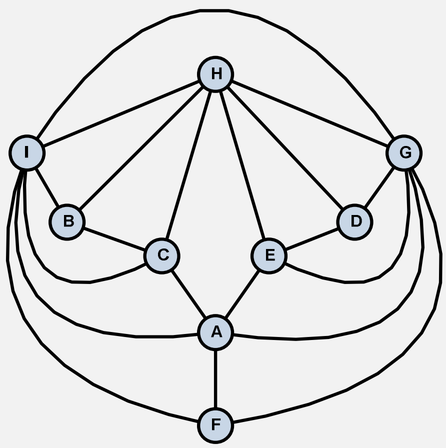
Dual graph

chromatic number is 4

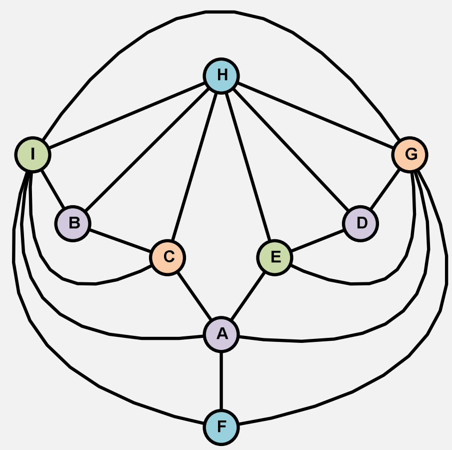
b.



Answer: According to the Four Color Theorem, the chromatic number of a planar graph is no greater than four. The given dual graph for the given map is apparently a planar graph. So four colors should be enough, although the map is contains many regions.



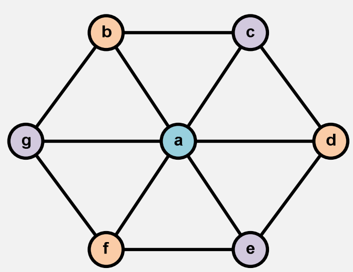
Dual graph



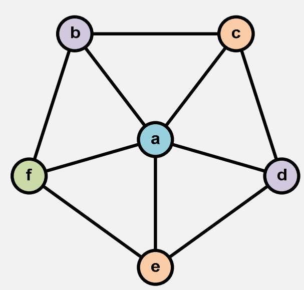
chromatic number is 4

Question: What is the chromatic number of W_n? [Chapter 9.8 Review]

Answer: If n is even, we need 3 colors. If it's odd we need 4. The vertex in the center has to be by itself because it's adjacent to all rest of the vertices. Then we need to have at least two colors for the vertices on the outside ring. They can take turns and not be the same as the neighbor. This wouldn't be enough for the odd n's though, because there will be an extra vertex that can't take either of the two colors. Let's look at an example.



W_6 needs 3 colors (6 is even)



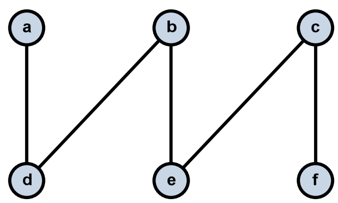
W_5 needs 4 colors (5 is odd)

Question: Which of these graphs are trees? [Chapter 10.1 Review]

Review: *Definition:* A tree is a connected undirected graph with no simple circuits.

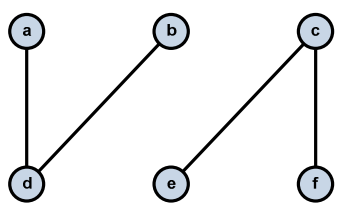
Theorem 1: An undirected graph is a tree if and only if there is a unique simple path between any two of its vertices

a.



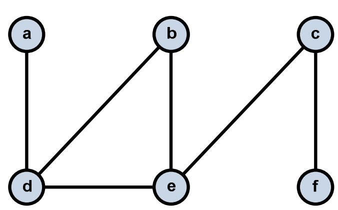
Answer: Yes, this is a tree. We can find one and only one pair wise simple path for all pairs of vertices on the single path connecting all vertices.

b.



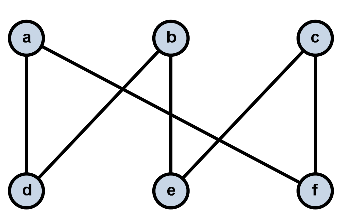
Answer: No, this is not a tree, because it is not a connected graph.

c.



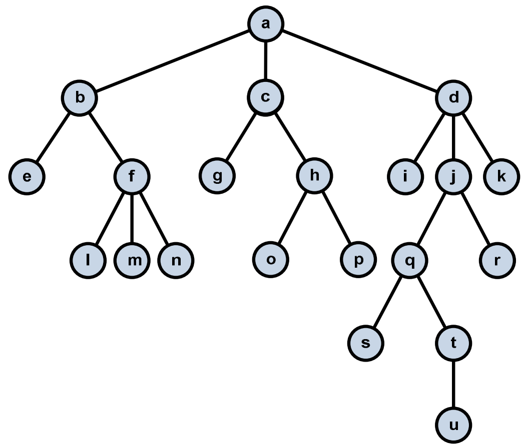
Answer: No, this is not a tree, because we see a circle consisting vertices b, e, d.

d.



Answer: No, this is not a tree, because there is one big circle made up from all vertices and edges in the graph..

Question: Answer these questions about the rooted tree illustrated. [Chapter 10.1 Review]



a. Which vertex is the root?

Answer: vertex a.

b. Which vertices are internal?

Answer: vertices $a, b, c, d, f, g, h, j, q, t$.

c. Which vertices are leaves?

Answer: vertices $e, l, m, n, g, o, p, i, s, u, r, k$.

d. Which vertices are children of ***j***?

Answer: vertices q, r.

e. Which vertex is the parent of ***h***?

Answer: vertex c.

f. Which vertices are siblings of ***o***?

Answer: vertices p.

g. Which vertices are ancestors of $\boldsymbol{m}$?

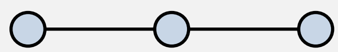
Answer: vertices f, b, a.

h. Which vertices are descendants of $\boldsymbol{b}$?

Answer: vertices e, f, l, m, n.

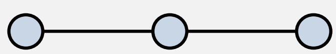
Question: Find the number of nonisomorphic unrooted trees with three vertices. [Chapter 10.1 Review]

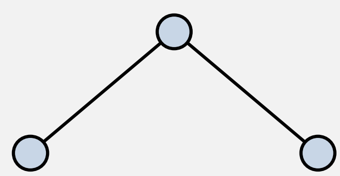
Answer: There is only one unrooted tree with three vertices:



Question: Find the number of nonisomorphic rooted trees with three vertices.

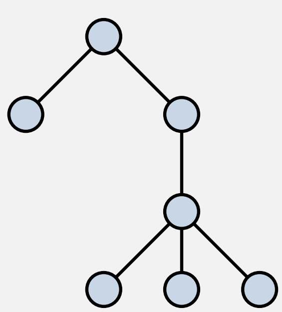
Answer: There are two nonisomorphic rooted trees with three vertices:



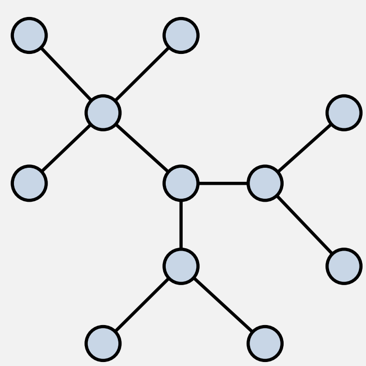


Question: How many edges does a tree with 10,000 vertices have? [Chapter 10.1 Review]

Answer: 9999, according to one property of trees that a tree with v vertices has $v - 1$ edges. Let's look at another two simple examples:



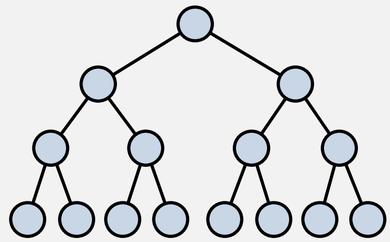
$v = 7, e = 6$



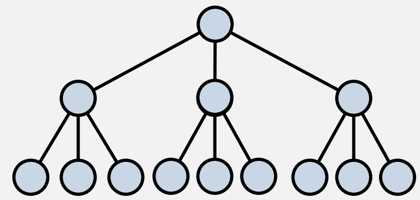
$v = 11, e = 10$

Question: How many vertices does a full binary tree with 1000 internal vertices have? [Chapter 10.1 Review]

Answer: 2001, according to one property of trees that a full m-ary tree with i internal vertices contains $n = mi + 1$ vertices. Let's look at another two simple examples:



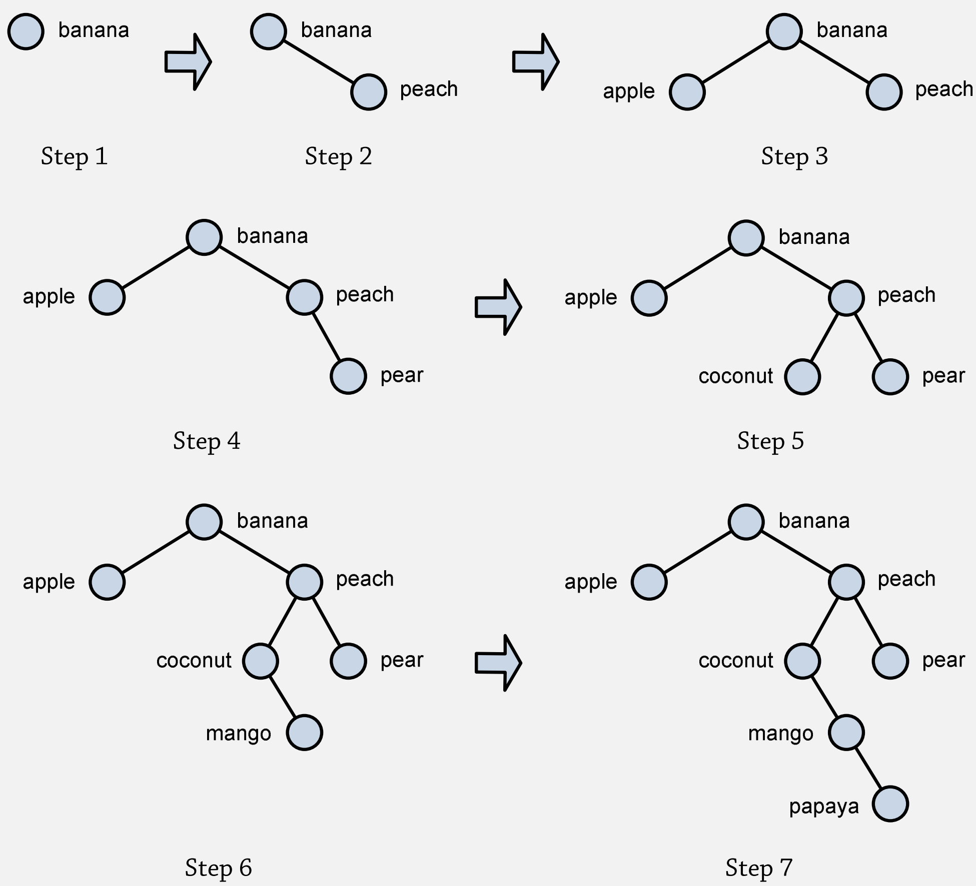
$m = 2, i = 7, n = 15$



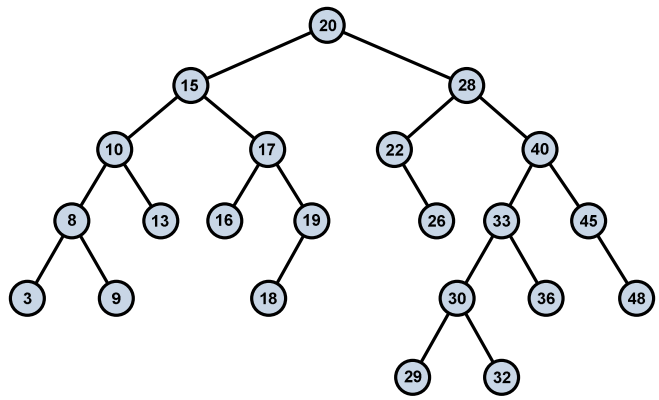
$m = 3, i = 4, n = 13$

Question: Build a binary search tree for the words "banana", "peach", "apple", "pear", "coconut", "mango" and "papaya" using alphabetical order? [Chapter 10.2 Review]

Answer: According to Binary Search Tree Insertion Algorithm, as we insert, we compare to the root of the current subtree and go left if to be insert value is less than the subtree root value and go right if otherwise. Repeat this procedure until we reach a point that the next node that we are going to compare the value with does not exit. In this case we place our value here.

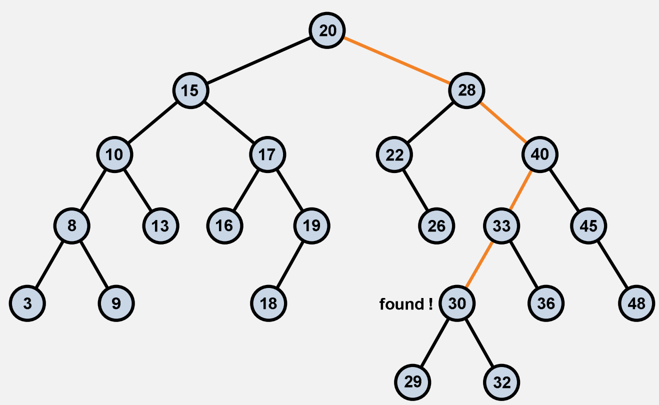


Question: Search for the word 30, and 14 in the following binary tree. [Chapter 10.2 Review]

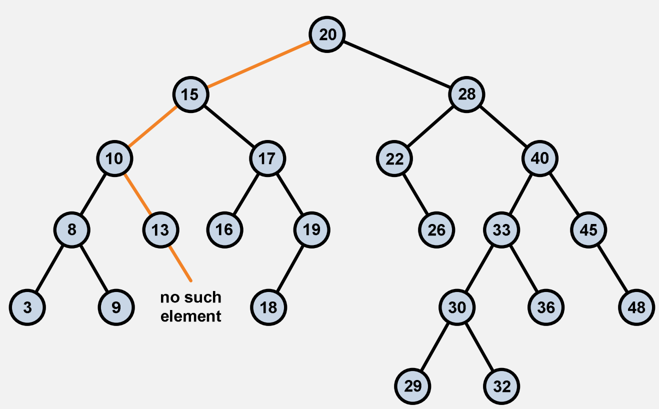


Answer: According to Binary Search Tree Search Algorithm, as we search, we compare the value the search for with the root of the current subtree, If they are the same then we found our value, if the value to find is smaller than the current root then go left, otherwise go right. Repeat this process until we find the value we are looking for or until we reach the point that the next node we are going the compare with does not exist.

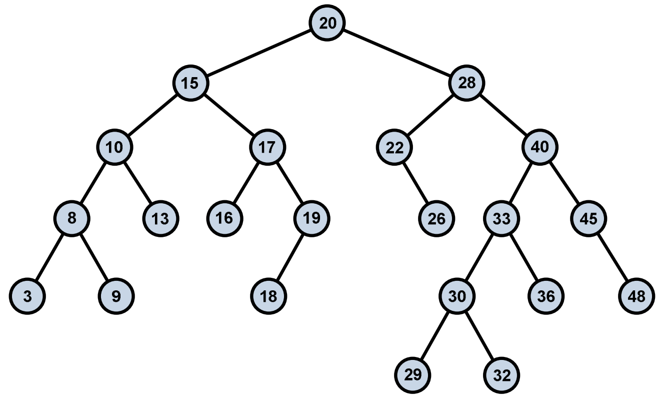
Looking for 30:



Looking for 14:

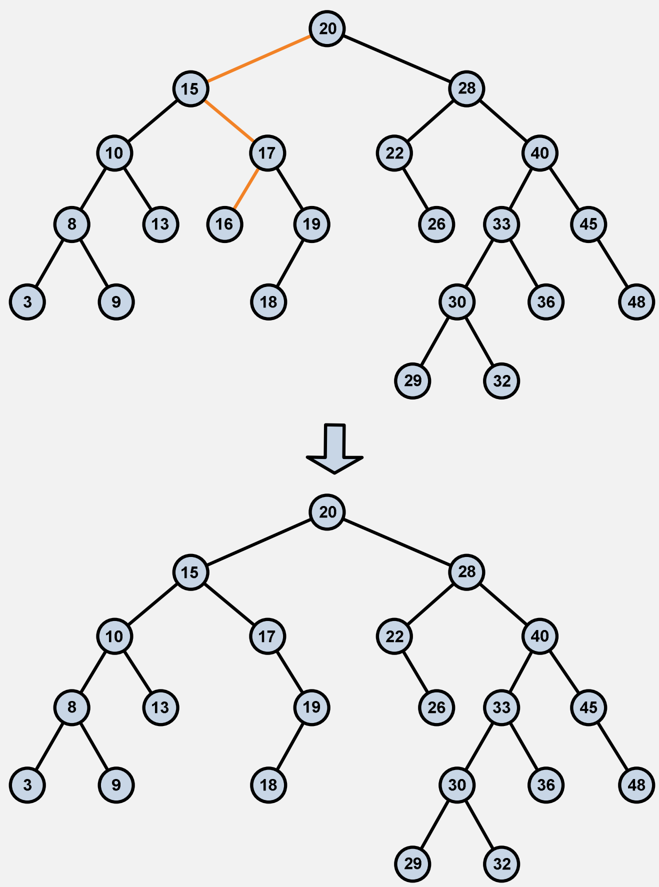


Question: Delete for the word 16, and 33 in the following binary tree. [Chapter 10.2 Review]

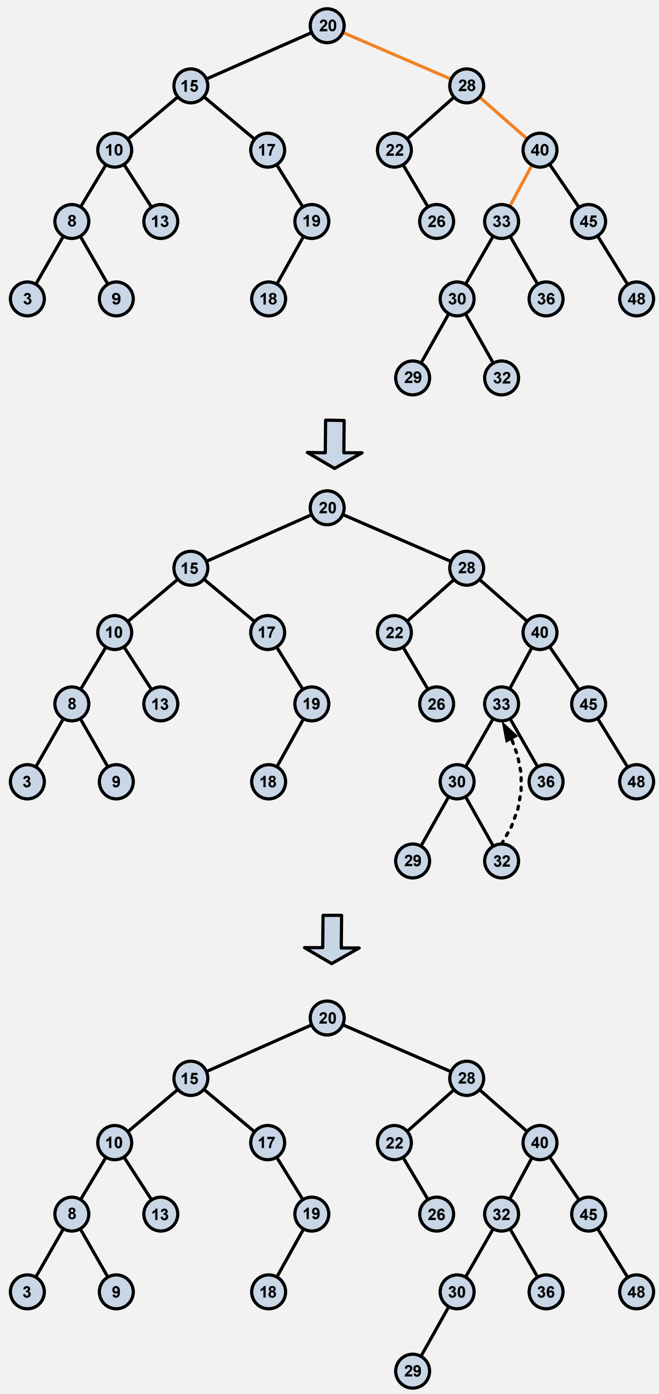


Answer: According to Binary Search Tree deletion Algorithm, first we need to search for the element to be deleted. The procedure in the previous question handles this step. Now we assume we found the element. If this element is a leaf element, then we just delete is the remaining structure is still a tree. If the element is an internal node, then we can't just delete it and disconnect it children node. Instead, we replace it with it left most value on the right subtree or the right most value on the left subtree.

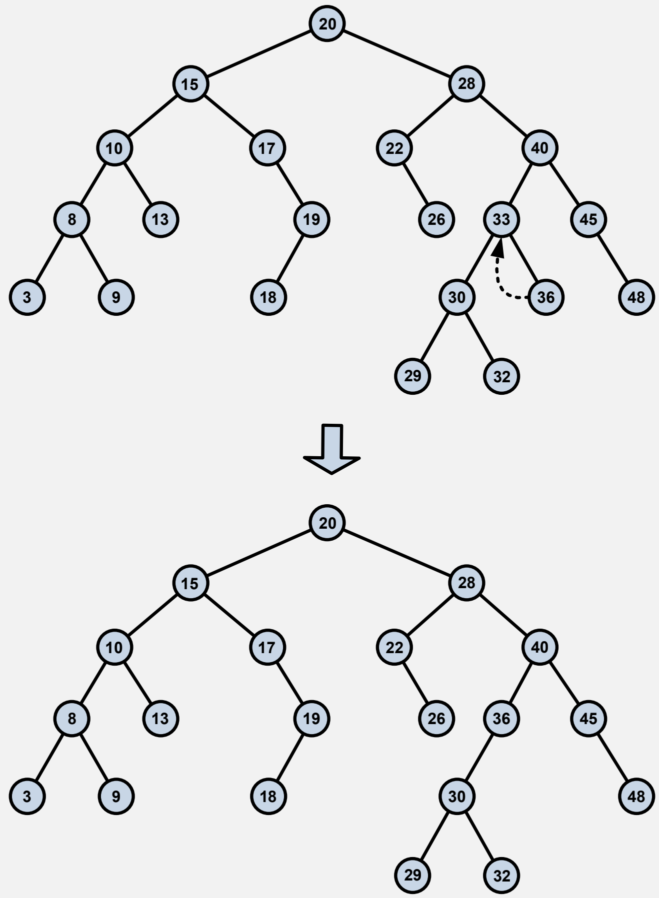
Deleting 16:



Deleting 33:



We replaced 33 with its right most child of the left subtree. We can replace the left most node of the right subtree:



As we can see, all of the resulting graph structures remained as Binary Search Trees because we took consideration of the tree properties and reestablished them.