Question 1

First, there is no error in the first half of the code. It is a correct choice to eliminate a specific value through a pointer, but in the final else part, it will enter an endless loop because they did not perform a delete action, such as in the first When I run the code for the second time, I find the position of the value to be deleted and change it to the smallest value on the right (so there will be an infinite loop here that will continue to change to the smallest value on the right), but because in the else The position of the pointer has not changed. In other words, the current position of the pointer is still the position of the value to be deleted. What we need to do is simply add a node=node. Right after performing the value conversion. Enter the next fork and continue to run until the end or a single branch is deleted using the pointer.

Question 2

Arrays: The advantages of arrays are mainly reflected in the fact that the first is because of the fast access speed (easier to retrieve), and the second is the faster efficiency. For example, we can easily read the information stored in any location. However, its memory usage efficiency is low, and its length cannot be expanded.

Linked List: Compared with the other two, his advantage is mainly reflected in the content change and deletion, we can quickly delete specific elements. For example, when we need to temporarily add a String information to the top, we only need to use add, and then he will link this memory to the existing linked list. It does not need to use too much memory (if it is an array, you need to move all the values).

Binary Tree: First of all, BST can output all the information in an orderly manner (you can know the order of the information), and it is convenient to query the maximum and minimum values or a specific value, and the double-linked list structure of BST can better store and more More information. For example, when we use BST to count a set of data, we can easily get the median of this set of values (some specific values, such as maximum and minimum).