I write another cpp file called “train” to implement the main function.

In the Railroad() method, firstly, I use a vector whose type is LinkedStack to store the holding stacks. I use the numberofTracks to set the size of the vector. Then do loops to deal with the input cars according to the number of cars. In the meantime, I also use a target number to indicate the number that should be output next. While traversing the input array, if the current number is not equal to the target number, then I will use the putinHold() method to put the number into one stack. In the putinHold() method, I traverse all the stack and get numbers on the top of the stack from left to right. If the number, which is going to input, is small then the top number I get from one stack, then I will push the number into current stack. Otherwise, I will go to the next stack. If the number id bigger than any top number, then I will store that number into a new holding stack and back to the Railroad() method to deal with the next number in the input array. If the current number is equal to the target number, then I will output it directly. After that, I will use the outputFromTrack() method to see can any number from stack be output. In the outputFromTrack() method, I simply get all top number from all stacks to estimate whether any one is equal to the target number. If there is no equal, then I will deal with the next number from input. Otherwise, I will output the eligible number and search all tracks again to find out the next target number.