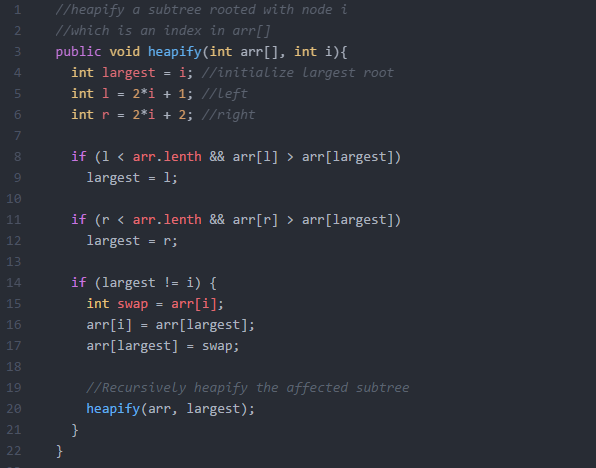
Klevis Tefa

Part A

Question A.1

1. 3
2. 2
3. [1, 1, 2, 3, 4, 5, 9, 6]
4. On the recursive call of sort(0, 2) we’ll have k = split(0,2) = 1. For the other half, on the recursion of sort(4,7) we’ll have k = split(4,7)=5.

Question A.2



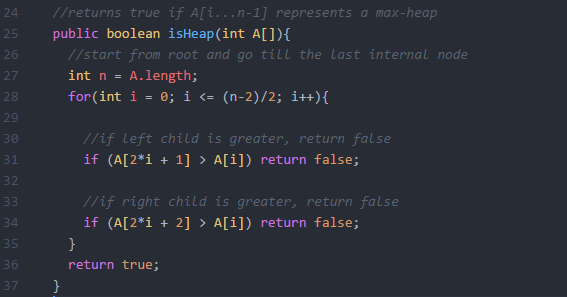
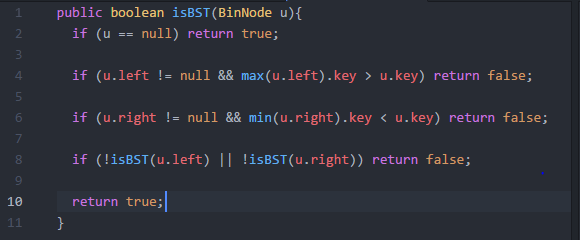
Question A.3

1. For the worst case scenario in which Insertion Sort has to sort an array that is in reverse order. So in that case we have that goes through a loop n times. However in that loop insertElement is called each time the loop is executed. In the case where the array is in reverse order the method insertElement. NOT COMPLETED !!!!!
2. [10, 9, 8, 7, 6, 5, 4, 3, 2, 1]🡪[9, 10, 8, 7, 6, 5, 4, 3, 2, 1] 🡪 [8, 9, 10, 7, 6, 5, 4, 3, 2, 1] 🡪 [7, 8, 9, 10, 6, 5, 4, 3, 2, 1]🡪[6, 7, 8, 9, 10, 5, 4, 3, 2, 1]🡪[5, 6, 7, 8, 9, 10, 4, 3, 2, 1]🡪[4, 5, 6, 7, 8, 9, 10, 3, 2, 1]🡪[3, 4, 5, 6, 7, 8, 9, 10, 2, 1]🡪[2, 3, 4, 5, 6, 7, 8, 9, 10, 1] 🡪[1, 2, 3, 4, 5, 6, 7 , 8, 9, 10]

Question A.4

1. [9, 1, 2, 3, 4, 5, 6, 7 ,8] 🡪 [8, 1, 2, 3, 4, 5, 6, 7, 9] 🡪 [7, 1, 2, 3, 4, 5, 6, 8, 9] 🡪 [6, 1, 2, 3, 4, 5, 7, 8, 9] 🡪 [5, 1, 2, 3, 4, 6, 7, 8, 9] 🡪 [4, 1, 2, 3, 5, 6, 7, 8, 9] 🡪 [3, 1, 2, 4, 5, 6, 7, 8, 9] 🡪 [2, 1, 3, 4, 5, 6, 7, 8, 9] 🡪 [1, 2, 3, 4, 5, 6, 7, 8, 9]
2. If we choose the starting pivot to be the largest number for any iteration this will result to the biggest number of comparisons this will force Ω (n2) behavior.

Question A.5

1. 
2. 
3. 