

b.

```
heapSort {  
    buildHeap (...);  $\rightarrow O(n)$   
    for (int i = size - 1; i >= 0; --i) {  $\rightarrow O(n)$   
        swap (...);  
        heapify (...);  $\rightarrow O(\log n)$   
    }  $\rightarrow O(n \log n)$   
}
```

\Rightarrow time complexity: $O(n \log n)$

c.

Since there is no new array or variable allocated in the memory, and all the operations are done based on the original input array; hence, the space complexity is $O(1)$.

d.

No. It will not change the time and space complexity.

Reason: 1. Build a min-heap.

2. loop through the array from left to right (start from the second element), inside the loop, heapify the array in each iteration