|  |
| --- |
| **Submitted By: Ahmed Dider Rahat**  Matriculation Number: 916146 |

Assignment D2

Software Setup for CS4DS

**Assignment Code:** After implementing all the assignment I pushed my code on my [Github](https://github.com/AhmedDiderRahat/csfb-wise2122/tree/main/assignment_4) link.

**Answer to the question no. 1**

After running the code and observng the documentation I have made the following decisions:-

**Case a:** list.sort() method only sort the exsiting list and return None. So, while assiginng and sorting a list in a same statement, the original list is sorted but the second list is assigning only the None from list.sort() method.

**Case b:** In this case, the list is first assigned to another list. As list assignment just assign the reference of one list to other, so list.sort() function sort both of the list.

**Case c:** In this case, sorted(list) method is used. This method sort the list and return the soterd iiterables instead of sorting the original list.

**Answer to the question no. 2**

I have implement the code using recursion. So, for the recusrssion process I consider the following assumtions.

1. **Base Conditions:** I assum base conditions are:
   1. if lower index is larger than upper then the element is not found.
   2. if the element is found in the middle of the given range.
2. **Recursive call:** If the middle point of the data is larger than the search element, we search the element into the left part otherwise into right part.



**Answer to the question no. 3**

**Time Complexity of binary search:** The binary search algorithm based on divide-and-conquere rule. So, each time its divide the data into two eual parts. So, the time complextiy follow function.

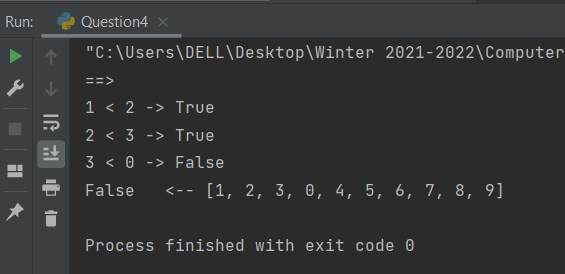
Best case: O(1)

Worst case: O()

Average case: O()

**Answer to the question no. 4**

After adding the \_log feature, I found the recurssive call immediatel stop when the first unsorted scenario found. The output of the log function given bellow:

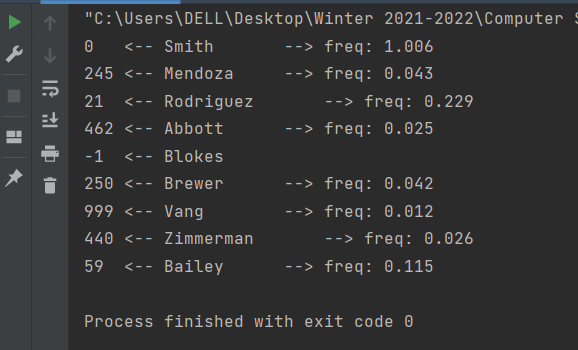


**Answer to the question no. 5**

I have imported the given ‘name dataset’ to my code base and implement my method for searching the name in the name list. The implementation file is Question6.py.

**Answer to the question no. 6**

The test function output is given bellow:



**Answer to the question no. 7**

For name searching problem I used python filter function to get the index of a given name. So, the time-complexity of the function is same as the filter function.

Best case: O(1)

Worst case: O(n)

Average case: O(n/2)