educative

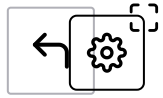
Solution Review: Find Minimum Value in List

This review provides a detailed analysis of the different ways to find a minimum value in a list.

We'll cover the following

Solution #1: Sort the list

```
1  def find_minimum(lst):
2      if (len(lst) <= 0):
3          return None
4      lst.sort()  # sort list
5      return lst[0]  # return first element
6
7
8  print(find_minimum([9, 2, 3, 6]))
```



This solution sorts the list in ascending order using `.sort` function and returns the first element which is also the minimum. Also, if the list is empty, `None` is returned.

> We used the generic Python `.sort()` function here, but in a real interview, you should implement your own sort function if you're going to use this solution. Learn about the famous sorting method, Merge sort (https://www.educative.io/edpresso/merge-sort-in-python).

Let's implement the sorting function below and call that function in the `find_minimum` function:

```
22                # Move the iterator forward
23                i += 1
24            else:
25                my_list[k] = right[j]
26                j += 1
27            # Move to the next slot
28            k += 1
29
30        # For all the remaining values
31        while i < len(left):
32            my_list[k] = left[i]
33            i += 1
34            k += 1
35
36        while j < len(right):
37            my_list[k]=right[j]
38            j += 1
39            k += 1
```

```
40
41
42  def find_minimum(lst):
43      if (len(lst) <= 0):
44          return None
45      merge_sort(lst)  # sort list
46      return lst[0]  # return first element
47
48
49  print(find_minimum([9, 2, 3, 6]))
```

Time Complexity

The build-in sort function `sort` and the `mergeSort` are in $O(nlogn)$. Since we only index and return after that, which are constant time operations, this solution takes $O(nlogn)$ time.

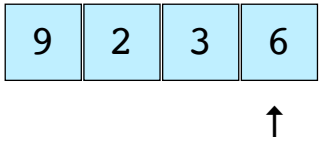
Solution #2: Iterate over the list

```
 1  def find_minimum(lst):
 2      if (len(lst) <= 0):
 3          return None
 4      minimum = lst[0]
 5      for ele in lst:
 6          # update if found a smaller element
 7          if ele < minimum:
 8              minimum = ele
 9      return minimum
10
11
```

```
print(find_minimum([9, 2, 3, 6]))

```

Start with the first element which is **9** in this example and save it as the smallest value. Then, iterate over the rest of the list and whenever an element that is smaller than the number already stored as `minimum` is come across, set `minimum` to that number. By the end of the list, the number stored in `minimum` will be the smallest integer in the whole list.




```
6 > 2
minimum = 2
```

4 of 4

Also, if the list is empty, `None` is returned.

Time Complexity

Since the entire list is iterated over once, this algorithm is in linear time, $O(n)$.

Back

Challenge 5: Find Minimum Value in List

Next

Challenge 7: Find Second Maximum V...

Mark as Completed

Report an Issue

Ask a Question (https://discuss.educative.io/tag/solution-review-find-minimum-value-in-list__introduction-to-lists__data-structures-for-coding-interviews-in-python)