**Analysis on implemented Mobile Manufacturing problem**

Implementation of given Mobile Manufacturing problem statement is done using below functions, to perform respective operations.

We opted for Greedy Algorithm to implement the solution as defined in the problem statement.

List of operations:

1. \_init\_ -> for initialization of the class Operations
2. calculate -> Return manufacturing, assembling
3. partition -> It selects a pivot element and makes left element smaller than it and right are larger than it.
4. quick\_sort -> Returns input as a sorted ascending order of the manufacturing timings

**Time complexities:**

1. **\_Init\_ function**

Which have 1 regular assignment statement, executes once.

Hence complexity would be O(1).

Time complexity of \_init\_ is O(1).

1. **calculate function**

We have loop within a loop.

Hence Time complexity of calculate is O(n2).

1. **Partition function**

This function has a loop within a loop, hence

Time complexity of calculate is O(n2).

1. **quick\_sort**

This function internally calls \_quick\_sort function, which internally calls partition function and the itself (\_quick\_sort) recursively in a sequence.

As we know, Time complexity of partition function is O(n2).

Time complexity of \_quick\_sort is O(n) as it calls itself recursively.

Considering higher boundary,

Time complexity of \_insert is O(n2).