

Instructions

Implement the merge sort algorithm.

Begin with the starting template: [ta10-template.py](#).

Implement the `merge_sort` function.

CORE REQUIREMENTS

1. Correctly split the list into two halves.
2. Correctly handle the base case, and the recursive calls to `merge_sort` each half.
3. Correctly merge the two sorted halves back together. (This is the meat of the assignment...)

Stretch Challenges

After completing the above steps. Make sure that everyone on the entire team is to this point and understands the material. Then, if you have time, move on to the following stretch challenges.

The stretch challenge is to implement quicksort.

1. Have any necessary methods and a base case in place.
2. Choose a partition item and put it in its proper place with everything less than it on the left, and everything greater on the right.
3. Complete the recursive calls to finish the algorithm.

Instructor's Solution

As a part of this team activity, you are expected to look over a solution from the instructor, to compare your approach to

that one. One of the questions on the I-Learn submission will ask you to provide insights from this comparison.

Please DO NOT open the solution until you have worked through this activity as a team for the one hour period. At the end of the hour, if you are still struggling with some of the core requirements, you are welcome to view the instructor's solution and use it to help you complete your own code. Even if you use the instructor's code to help you, you are welcome to report that you finished the core requirements, if you code them up yourself.

After working with your team for the one hour activity, [click here for the instructor's solution](#).