

The worst case for merge sort is where the greatest number of comparisons possible is done in the Merge() function every time it is called. This is where left and right is made up of alternating pairs for example: [1, 3] and [2, 4] where 1 compares to 2, then 2 compares to 3, then 3 compares to 4 and finally 4 is appended as left is now empty.

This logic can be applied at every call of Merge in order to produce the worst case initial list from a sorted list. For example, let us take [1, 2, 3, 4, 5, 6, 7, 8].

List = [1, 2, 3, 4, 5, 6, 7, 8]: Left = [1, 3, 5, 7] + Right = [2, 4, 6, 8]

List = [1, 3, 5, 7]: Left = [1, 5] + Right = [3, 7]

List = [2, 4, 6, 8]: Left = [2, 6] + Right = [4, 8]

Therefore, worst case list for $n = 8$ is [5, 1, 7, 3, 6, 2, 8, 4]

Recall also that this is a hybrid merge sort so a selection sort must be performed on sublist smaller or equal to $n = 4$. Therefore, worst case for selection sort occurs when $n = 4$

Therefore, worst case for the hybrid algorithm is where there is a maximum amount of comparisons in Merge for each call of Merge and also where there is the maximum number of sublists where $n = 4$ so to invoke selection sort worst case. Therefore, for worst case, the list should be of length 2^n where n is an integer greater than 1.