**WRITE UP**

**Longest increasing subsequence:**

The Longest Increasing Subsequence (LIS) problem is to find the length of the longest subsequence of a given sequence such that all elements of the subsequence are sorted in increasing order.

**Steps:**

1. To find the LIS using recursion, max\_ending, max are the two temporary variables are taken to store the values.
2. max\_ending is to find length of LIS ending with element arr[n-1].
3. Max is to find overall maximum as the LIS may end with an element before arr[n-1].
4. The value of LIS of full array of size n is stored in max which is our final result
5. Recursively get all LIS ending with arr[0], arr[1] ..arr[n-2].
6. If arr[i-1] is smaller than arr[n-1], and max ending with arr[n-1] needs to be updated, then update it.
7. Compare max\_ending with the overall max and return length of LIS ending with arr[n-1]

**GitHub link:**

Under the repository TEST ABC, the file LIS is the project file

[**https://github.com/PradeepaSaravanan/TEST\_ABC.git**](https://github.com/PradeepaSaravanan/TEST_ABC.git)