**Project Name:** **Longest increasing subsequence**

**Developer Name: Kapil Davey**

**Project Description:**

1. We can find the longest subsequence using dynamic programming.
2. By using the dynamic programming approach, we are going to trace the original array at each position.
3. We will check each index of the array that whether it is suitable for the longest increasing subsequence or not.
4. First, we consider two variables, i and j, where 'i' variable will point to each position of the array and for every 'i', 'j' will start from the beginning.
5. We will consider two arrays named 'length' and 'subsequence', where the length array will store the length of each subsequence, and the subsequence will store the elements of subsequence.
6. The base condition of the dynamic programming is that we initialize each array cell with 1 value shown below. We are considering that every single element in an array is the longest increasing subsequence of length 1.

**Algorithm :-**

**Step 1-** Start

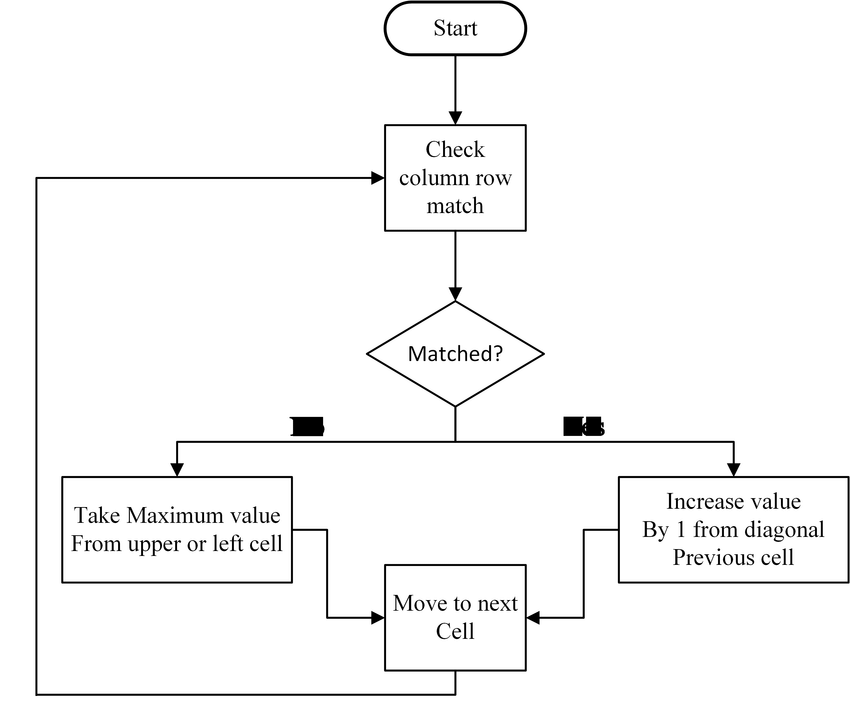
Step 2- write a function to find the longest increasing subsequence

Step 3- check whether current element is greater than the previous element in LIS

Step 4- Return the maximum number using max ()

Step 5-Print length of LIS

Step 6-Stop

**Flow of Project:**

**Conclusion:** This project is designed to find longest increasing subsequence using dynamic programming.

**GitHub Reciprocity Link**:- https://github.com/kapildavey/Phase1.git