**100 Days of Code**

**DAY 1:**

**Longest Increasing Subsequence**

Given a sequence, find the length of the longest increasing subsequence from a given sequence .  
The longest increasing subsequence means to find a subsequence of a given sequence in which the subsequence's elements are in sorted order, lowest  
to highest, and in which the subsequence is as long as possible. This subsequence is not necessarily contiguous, or unique.

**Note:** Duplicate numbers are not counted as increasing subsequence.

For example:  
 LIS for { 10, 22, 9, 33, 21, 50, 41, 60, 80 } is {10, 22, 33, 50, 60, 80}.

# K’th Smallest/Largest Element in Unsorted Array

Given an array and a number k where k is smaller than size of array, we need to find the k’th smallest element in the given array. It is given that ll array elements are distinct.

Input: arr[] = {7, 10, 4, 3, 20, 15}

k = 3

Output: 7

Given an array A[] and a number x, check for pair in A[] with sum as x

Write a program that, given an array A[] of n numbers and another number x, determines whether or not there exist two elements in S whose sum is exactly x.

**Day 2:**

# Minimum number of jumps to reach end

Given an array of integers where each element represents the max number of steps that can be made forward from that element. Write a function to return the minimum number of jumps to reach the end of the array (starting from the first element). If an element is 0, then cannot move through that element.

**Example:**

Input: arr[] = {1, 3, 5, 8, 9, 2, 6, 7, 6, 8, 9}

Output: 3 (1-> 3 -> 8 ->9)

# Find the first repeated character in a string

Given a string, find the first repeated character in it. We need to find the character that occurs more than once and whose index of **second occurrence** is smallest.

**Examples:**

Input: ch = "geeksforgeeks"

Output: e

e is the first element that repeats

Input: str = "hello geeks"

Output: l

l is the first element that repeats

**Find median in a stream**

Given an input stream of **N** integers the task is to insert integers to stream and print the median of the new stream formed by each insertion of **X** to the stream.  
  
**Example**:  
Flow in stream : 5, 15, 1, 3   
5 goes to stream --> median 5 (5)   
15 goes to stream --> median 10 (5, 15)   
1 goes to stream --> median 5 (5, 15, 1)   
3 goes to stream --> median 4 (5, 15, 1, 3)