package LongestIncreasingSubsequences;

public class LIS {

public static void main(String[] args){

int[] nums = {1,3,2,9,6,10,5};

printLIS(nums);

}

public static void printLIS(int[] nums){

String[] paths = new String[nums.length];

int[] sizes = new int[nums.length];

for(int i=0; i<nums.length; i++){

sizes[i] = 1;

paths[i] = nums[i] + " " ;

}

int maxLength = 1;

for(int i=1; i<nums.length; i++){

for(int j=0; j<nums.length; j++){

if(nums[i]>nums[j] && sizes[i] < sizes[j] + 1){;

sizes[i] = sizes[j] + 1;

paths[i] = paths[j] + nums[i] + " ";

if(maxLength < sizes[i])

maxLength = sizes[i];

}

}

}

for(int i=1; i<nums.length; i++){

if(sizes[i] == maxLength)

System.out.println("Longest Increasing Subsequences : " + paths[i]);

}

}

}