**package** lesson04.task419;  
**import** java.util.\*;  
  
**public class** LongestIncreasingSubsequence {  
  
 **public static void** main(String[] args) {  
 List<Integer> source = *getSourceSequence*();  
  
 LongestIncreasingSubsequence lis = **new** LongestIncreasingSubsequence();  
 List<Integer> result = lis.getLongestIncreasingSubsequence(source);  
  
 System.***out***.println(**"Source sequence \n"**+source);  
 System.***out***.println(**"Longest increasing subsequence \n"**+result);  
 System.***out***.println(result);  
  
 }  
  
 **private** List<Integer> getLongestIncreasingSubsequence(List<Integer> source) {  
  
 List<Integer> resultList = **new** ArrayList<>();  
 List<Integer> tempList = **new** ArrayList<>();  
  
 **for** (**int** i = 0; i < source.size() - 1; i++) {  
  
 **if** (source.get(i) < source.get(i + 1)) {  
 **if** (tempList.isEmpty()) tempList.add(source.get(i));  
 tempList.add(source.get(i + 1));  
 }  
  
 **if**(source.get(i) >= source.get(i + 1)||(i+2==source.size())) {  
 **if** (resultList.size() < tempList.size()) {  
 resultList = List.*copyOf*(tempList);  
 tempList.clear();  
 }  
 }  
 }  
  
 **return** resultList;  
 }  
  
 **private static** List<Integer> getSourceSequence() {  
 *//3,4,10-15 is longest subsequence* List<Integer> mockList = Arrays.*asList*(1,2,3,2,5,6,7,3,4,10,11,12,13,14,15,9,10);  
 **return** mockList;  
 }  
  
  
}