## Objective

To enumerate all the permutations of the n objects not necessarily distinct and display the output based upon the verbose mode entered.

## Implementation

This program implements the algorithm for permutation of n numbers, given in "A Draft of Section 7.2.1.2: Generating All Permutations” in "The Art of Computer Programming Pre-Fascicle 2b" By Dr. Donald E. Knuth”.

The essence of the algorithm is as follows:

1. We sort the input set of numbers. Visit this first permutation.
2. The next step is to find *j* as follows:
   1. Start by setting *j* = *n*-1.
   2. Decrease *j* till you find.
3. Next step involves finding l as follows.
   1. Start by setting *l*=*n*.
   2. Decrease *l* till you find.
   3. Swap (in the swap step it is made sure that two numbers are not equal this ensure that duplicate permutation is not generated.)
   4. Visit this permutation.

### Additional work:

* while generating permutations, we have generated permutations in lexicographical order. This was done by adding a step after 3c in the above algorithm. By reversing the array elements from. This gives the permutations in lexicographical ordering.
* it also works for negative numbers e.g. if set of number is {-1,2,2,3,4}.

How To run :

PermComb.java :

This file contains a JAVA class implementing enumeration of permutations of set of numbers (not distinct) and showing output depending upon the verbose mode.

The program accepts inputs from standard input (stdin)

1. First user need to input total number of objects (n),

2. Second argument is verbose mode (v), and

3. Third argument is the list of numbers (up to to n numbers) may not be distinct.