Expand All Collapse All

Inversions

- **My `generate()` function returns a different permutation of length *n* with exactly *k* inversions. Is that OK?**

  Yes. In general, many different permutations of length n have exactly k inversions. The autograder will accept any such one.

- **What should `generate()` return if $n = k = 0$?**

  An array of length 0. This is a common corner case in methods that return arrays.

- **What should `generate()` do if $k > n(n-1) / 2$?**

  As indicated in the assignment specification, your can assume this is not the case. If you wish to handle such situations, we would recommend either returning `null` or throwing an `IllegalArgumentException`.

- **Any hints on how to implement `generate()` to run in linear time?**

  Use a *greedy* approach. If $k \geq n-1$, put element $n-1$ first in the permutation, so that it is inverted with $n-1$ elements; otherwise put it last in the permutation, so that it is inverted with 0 elements. How can you continue this approach to determine where element $n-2$ should go?

- **Is there a linear-time algorithm for counting the number of inversions in an array (or permutation) of length *n*?**

  No such algorithm is known. However, there is an $n \log n$ algorithm that uses the same *divide-and-conquer* technique that we'll encounter when we discuss sorting algorithms and mergesort.

Ramanujan numbers

- **How do I achieve a running time that is proportional to $n^{1/3}$?**

  Since you are searching for integers a, b, c, and d such that $n = a^3 + b^3 = c^3 + d^3$, you need only consider values for a between 1 and $n^{1/3}$. For a given value of a, the only possible way to choose b to make $a^3 + b^3 = n$ is $b = (n - a^3)^{1/3}$.

- **How do I parse a command-line argument that is a long integer?**

  Use `Long.parseLong()`.

- **Is $91 = 6^3 + (-5)^3 = 4^3 + 3^3$ a Ramanujan number?**

  No. The cubes must be positive.

- **What is the smallest integer that can be expressed as the sum of two fourth powers in two different ways?**

  $635{,}318{,}657 = 59^4 + 158^4 = 133^4 + 134^4$. It is conjectured that no integer can be expressed as the sum of two fifth (or higher) powers in two different ways.

Maximum square submatrix

- **Any hints on solving the problem in n^2 time?**

  Use *dynamic programming*. Specifically, for each row i and column j, compute the size of the largest contiguous square submatrix whose lower right endpoint is (i, j). What are the base cases? How can you compute this quantity for row i and column j if you know it for smaller values of i and j.

- **Given an *n*-by-*n* matrix of 0s and 1s, how hard is it to find the largest contiguous rectangular submatrix of all 1s?**

  It can also be done in n^2 using dynamic programming, but the algorithm is more complicated.

- **Given an *n*-by-*n* matrix of positive and negative integers, how hard is it to find a contiguous rectangular submatrix that maximizes the sum of its entries?**

  It can be done in n^3 time using dynamic programming and a few other tricks. However, no n^2 or $n^{2.999}$ algorithm is known for the problem.