

## Numbers as Strings

---

### Due Date:

This assignment is due by Friday, January 21 *by the beginning of lecture*.

---

### Purpose:

This laboratory assignment is a chance to review problem decomposition by methods. This is likely the last laboratory assignment to be focussed solely on static methods. Use this assignment to practice good functional decomposition.

---

### Numbers to Strings

⇒ Implement a static `String numberToString(int n)` that converts a integer  $n$  between 1 (inclusive) and one billion (exclusive) into a `String`. The `numberToString` method can cover a larger range if you wish. It should take appropriate action if the argument is out of range.

For instance `numberToString(312000789)` should return three hundred twelve million seven hundred eighty-nine.

*Carefully* think about how to decompose this problem. For instance, if you have a method that can handle the range 1–999, how can you use that method to create another method that handles the range 1–999 999?

Aim to create methods that have well-defined purposes, but which have no more than around twenty lines of code each. Exploit small private static arrays of `Strings` if you can.

⇒ Implement a test method that provides *automated* testing of the `numberToString` method. Here, “automated” means that the test method does not require human input. Think carefully about what test cases provide good automated testing.

⇒ Determine the total number of ‘w’*s* in the words “one, two, three, . . . , thirty-four million nine hundred ninety-nine thousand nine hundred ninety-eight, thirty-four million nine hundred ninety-nine thousand nine hundred ninety-nine, thirty-five million.”