NOTES: STACKS LAB, PART 2

UMD CS DEPARTMENT

1. Overview

Last week, you created a simplified stack implementation using Java arrays as a *backing store*. No doubt, you have had some time to think about the limitations of your implementation—such as having to restrict the size of the stack ahead of time, etc. Naturally, more robust implementations of the Stack interface are available in the Java language, and you will use one for this particular assignment because this lab is about *using* stacks as opposed to implementing them.

The purpose of this week's lab is to use a stack to implement an algorithm that converts non-negative base-10 integers into their equivalent values in different bases (< 10). For example, your program might be given the number 12 in base-10, and asked to convert this number into its equivalent value in base-2, which would be 1100. Naturally, you need to know how to do this, i.e., you need to understand the algorithm for translating non-negative base-10 numbers into base-n numbers, where $n < 10$.

Algorithm 1 Convert a (non-negative) base-10 number, n, to base-b, where $b < 10$.

```
function CONVERT($n, b, st, sb$)        ▷ Convert n to base b, given stack $st$ and StringBuilder $sb$.
    if $n = 0$ then
        $sb.append(n)$
        return $sb.toString()$           ▷ Return the String representing the converted number.
    end if
    repeat
        $r \leftarrow$ remainder($n, b$)
        $n \leftarrow$ quotient($n, b$)                    ▷ Replace $n$ with the quotient.
        $st.push(r)$                               ▷ Push each remainder onto the Stack
    until $n \leq 0$
    while $st$ is not empty do           ▷ Digits appear in the Stack, but in reversed order: unwind
        $sb.append(st.pop())$
    end while
    return $sb.toString()$        ▷ Recall: this function returns a String representing the number.
end function
```

1.1. Using the algorithm.

Naturally, you will need to modify this algorithm to fit the method signature provided by the project. Your Teaching Assistants may also provide you with some additional guidance in this regard. Have fun!