

Homework #4

In this assignment you will implement an algorithm that uses stacks to add numbers of any size.

The largest magnitude of integers is limited. We are not able to add 18,274,364,583,929,273,748,525 and 8,129,498,165,026,350,236 because integer variables cannot hold such large values, let alone their sum.

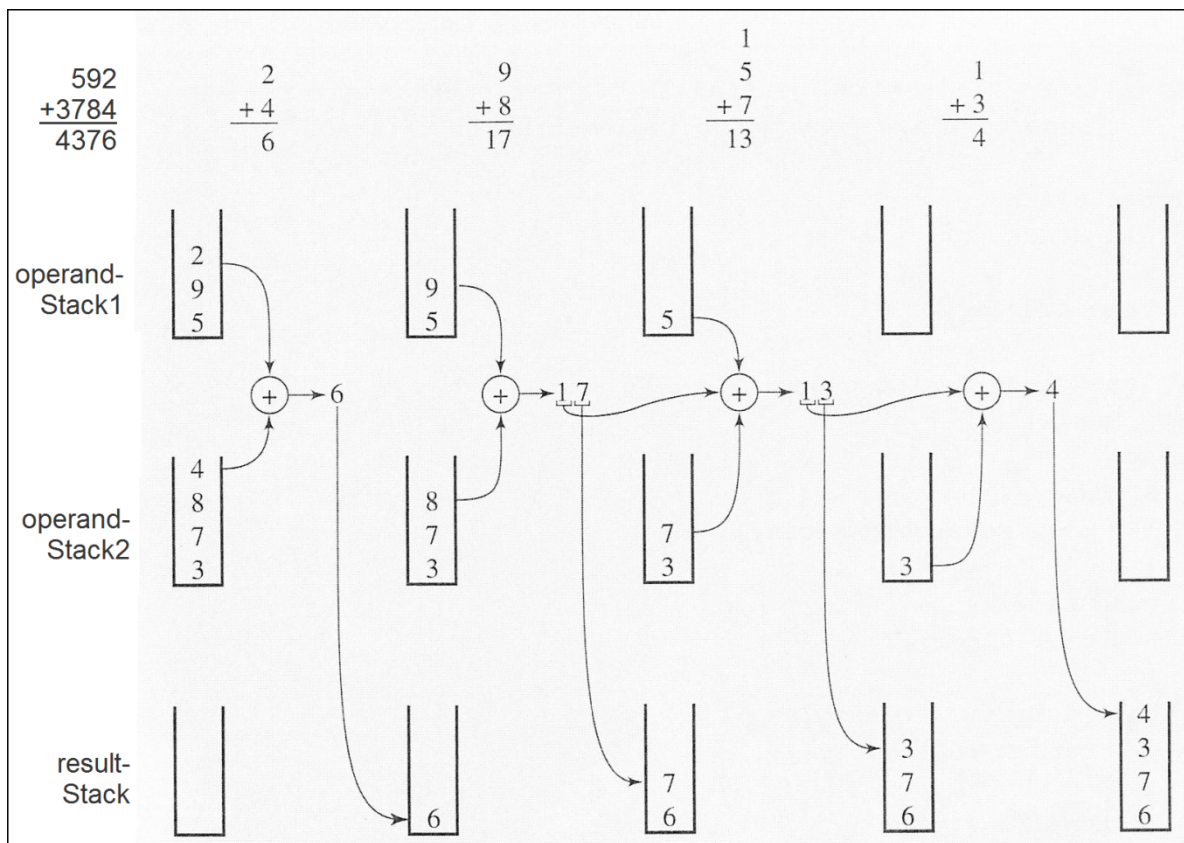
The problem can be solved if we treat these numbers as strings of numerals, store the numbers corresponding to these numerals on two stacks, and then perform addition by popping from the stacks. The pseudocode for this algorithm is as follows:

```

addLargeNumbers (number1, number2)
  read the numerals of the first number and store them on one stack
  read the numerals of the second number and store them on another stack
  var result := 0
  while at least one stack is not empty
    pop a numeral from each nonempty stack and add them to result
    push the unit part of addition onto a new stack called the result stack
    store the carry part of the addition in result
  push result onto the result stack if it is not zero
  pop numbers from the result stack and display them

```

The following diagram shows an example of adding numbers 592 and 3784:



All code implemented in this assignment should be in a class called `Homework4`. You may use the data structures and algorithm code from the lecture notes.

- a) **(6 point)** Implement the `addLargeNumbers` method with the following prototype:

```
public static void addLargeNumbers(String num1, String num2);
```

This method should output the result of adding the two numbers passed in as strings. Here is an example call to this method with the expected output:

```
// Sample call to addLargeNumbers
Homework4.addLargeNumbers("592", "3784");

// Expected output: 4376
```

- b) **(3 points)** Implement a `main` method that demonstrates adding at least three pairs of large numbers (numbers larger than can be represented by a `long`).
- c) **(1 point)** Make sure your source code is well-commented, consistently formatted, uses no magic numbers/values, and follows programming best-practices.

Turn in all source code, program output, diagrams, and answers to questions in a single PDF document.