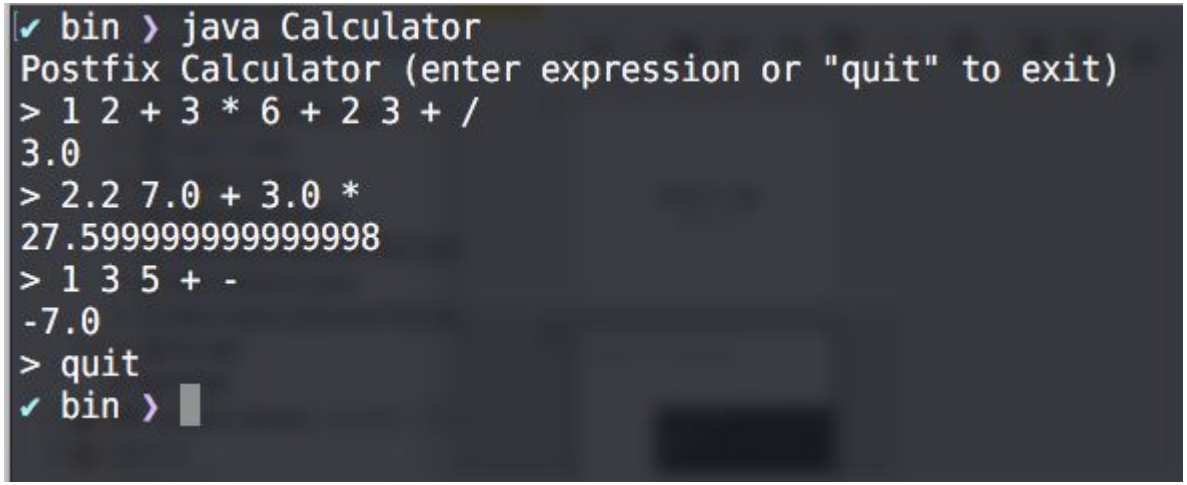
ICS-211 Lab

Assignment 5

Assignment 5 - Postfix Calculator

- As per the assignment instructions, you need:
  - A `Calculator` class
  - The method: `double calculate(String expr)`
  - An interface (command line or GUI) to take user input.
- Parse expression: `String.split`
- Read input: `BufferedReader`
- There are many other ways to read and parse input



Example

```
1 3 5 + - 4 6 + *
```

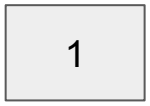
R0 = ?
R1 = ?
Acc = ?

Stack

Example

1 3 5 + - 4 6 + *

R0 = ?
R1 = ?
Acc = ?

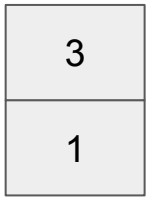


Stack

Example

1 3 5 + - 4 6 + *

R0 = ?
R1 = ?
Acc = ?



Stack

Example

1 3 5 + - 4 6 + *

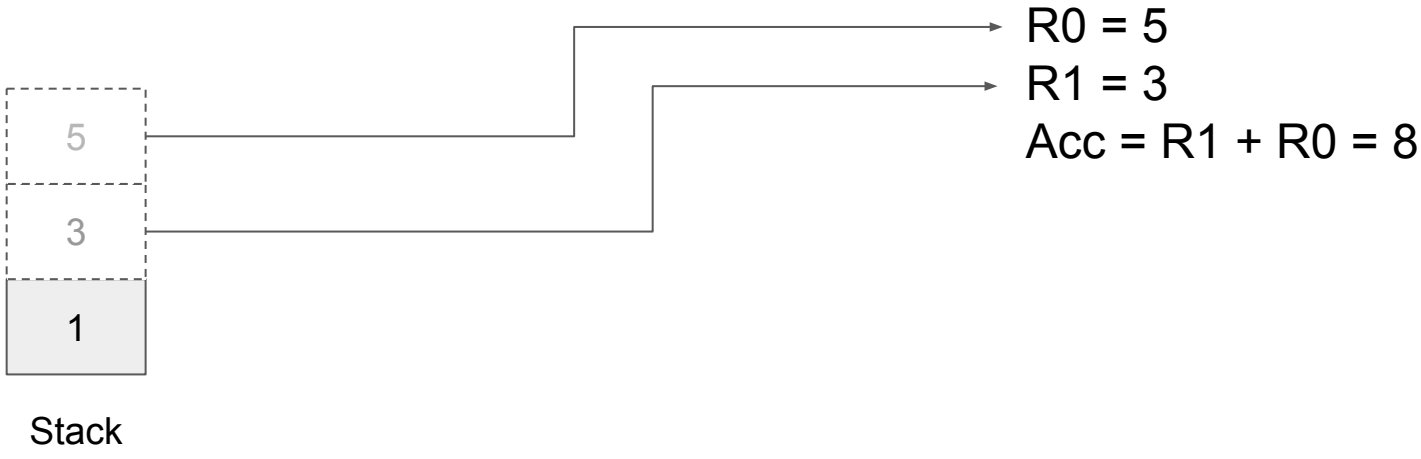
5
3
1

Stack

R0 = ?
R1 = ?
Acc = ?

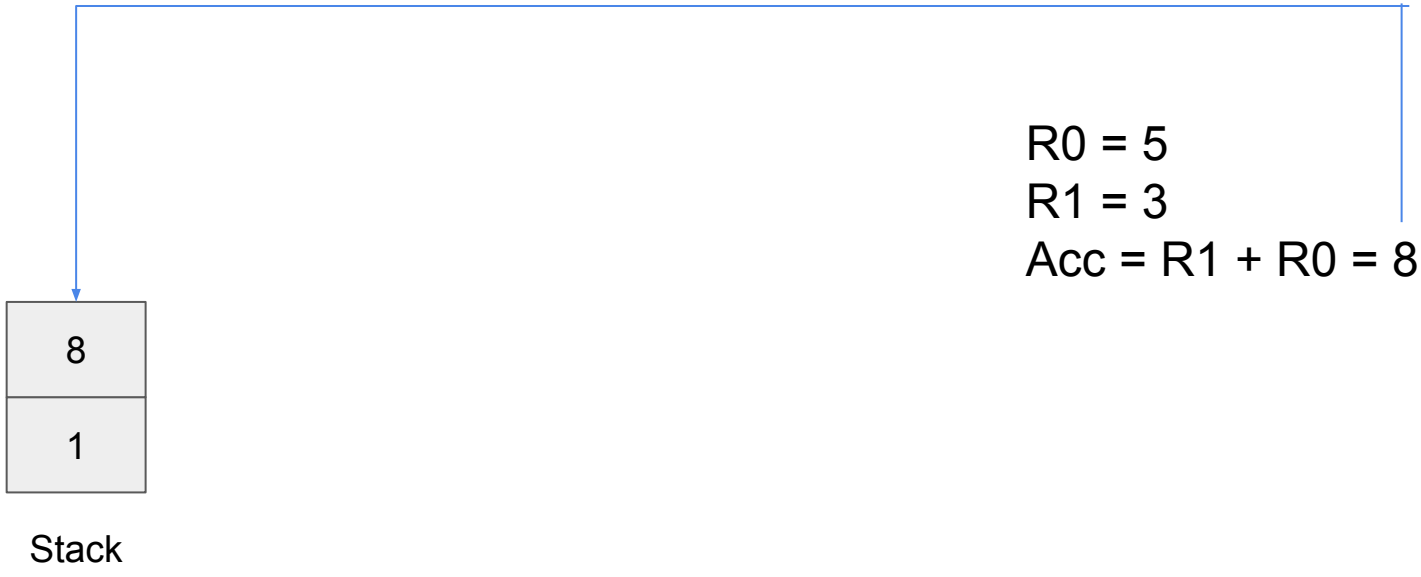
Example

1 3 5 + - 4 6 + *



Example

1 3 5 + - 4 6 + *



Example

1 3 5 + - 4 6 + *

R0 = 8
R1 = 1
Acc = R1 - R0 = -7

-7

Stack

Example

1 3 5 + - 4 6 + *

R0 = 8
R1 = 1
Acc = R1 - R0 = -7

4
-7

Stack

Example

```
1 3 5 + - 4 6 + *
```

6
4
-7

Stack

R0 = 8
R1 = 1
Acc = R1 - R0 = -7

Example

1 3 5 + - 4 6 + *

R0 = 4
R1 = 6
Acc = R1 + R0 = 10

10
-7

Stack

Example

1 3 5 + - 4 6 + *

R0 = 10
R1 = -7
Acc = R1 * R0 = -70

-70

Stack

Next Extra Credit (A05)

- Worth 10 points
- There will *not* be extra credit for A06
- Implement a generic stack class

```
class MyStack <E> {
        /**
         * Returns true if the stack is empty, false otherwise.
         */
        boolean empty()

        /**
         * Returns the top element of the stack.
         */
        E peek()

        /**
         * Add 'item' to the top of stack.
         */
        push(E item)

        /**
         * Removes top item on stack and returns it.
         */
        E pop()
}
```

- You can use your `MyArrayList` or `MyLinked` list.
- Java containers (i.e., `java.util.Stack`, `java.util.ArrayList`, etc.) ***can't*** be used

- You must use `MyStack` in your `Calculator` implementation