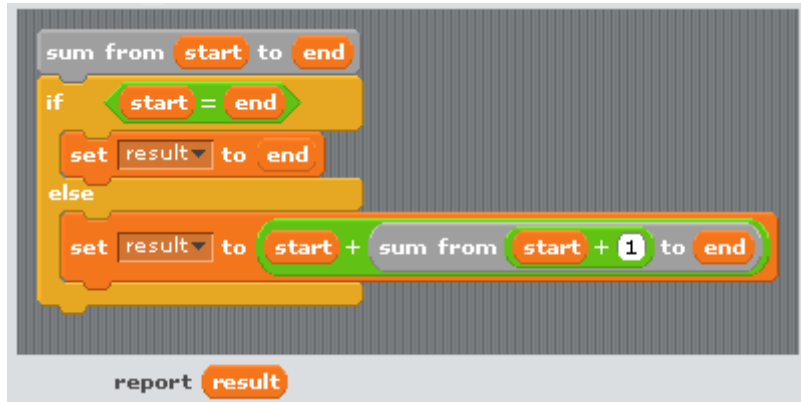# Lab 11

In today's lab you're going to see some different styles of recursion. Sometimes we want to do recursion by just delegating some work to do. Today we're going to delegate the work we need done PLUS the current answer! Sometimes it can be helpful to have the current answer with us.

☐ Below is recursive program to add up all of the numbers between two numbers. For example "sum from 1 to 3" is 1 + 2 + 3 = 6. "sum from 2 to 5" is 2 + 3 + 4 + 5 = 14. As a warm up, write a block that does the same thing as sum in a non-recursive way.
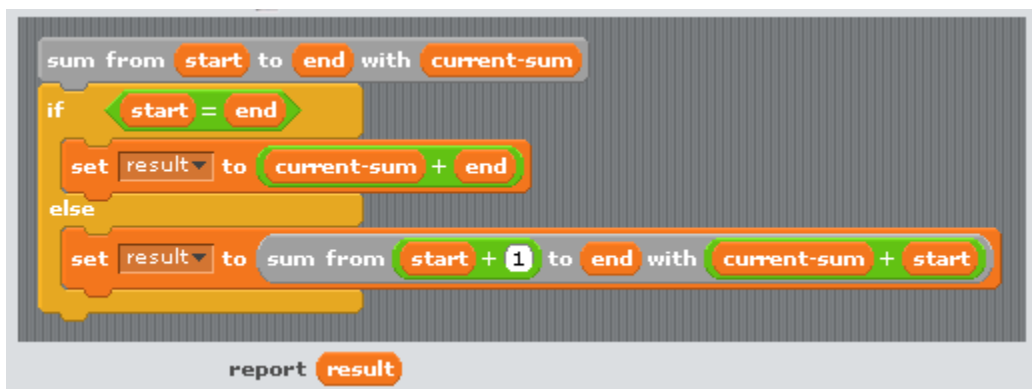
```
sum from start to end
if      start = end
    set result to end
else
    set result to start + sum from start + 1 to end

report result
```

☐ Below is another recursive version of "sum". We wrote the block "sum2" and all it does is delegate to "sum from %start to %end with %current-sum". This delegation is different than our typical delegation. In our typical delegation we always need to make the problem smaller. All sum2 is going to do is say "hey why don't you deal with this – and I'll just tell you that current-sum should start at zero and you can do the rest."

  ○ The real code is in "sum from %start to %end with %current-sum". Make sure you understand how it works.
  ○ Why would you even want to write "sum2"?

```
sum2 from start to end

report sum from start to end with 0
```

```
sum from start to end with current-sum
if      start = end
    set result to current-sum + end
else
    set result to sum from start + 1 to end with current-sum + start

report result
```

- For the next exercise, you're going to need something that removes the first letter from a sentence, word, or number. Here we've provided a non-recursive version.

```
remove first letter of (word)
set (index) to 2
set (result) to [ ]
repeat until ((index) > (length of (word)))
    set (result) to (join (result) (letter (index) of (word)))
    change (index) by 1
report (result)
```

  o Could you write a recursive version? Why or why not?
  o Create a reporter block like is shown below. It should use recursion to copy the letters from a range within a sentence. For example "Copy of letters 4 to 9 from Happy-Birthday" should report "py-Bir"

```
copy of letters (start) to (end) from (sentence)
```

  o Could you write "remove first letter" using copy of letters? (This uses the same big idea of the first step being that you delegate almost the entire problem!)
- We've provided a recursive version of "sum of digits from %sentence" that takes a number and then adds up the digits. For example, "sum of digits from 248" should be 2 + 4 + 8 = 14. Or "sum of digits from 100000000001" should be 2.
  o Write a recursive version that keeps track of the sum as you're going. (This will involve two blocks.)

```
sum of digits from (sentence)
if ((length of (sentence)) = 0)
    set (result) to 0
else
    set (result) to ((letter 1 of (sentence)) + (sum of digits from (remove first letter of (sentence))))
report (result)
```

Homework

For homework you will submit

- A recursive version of "remove first letter" using "copy of letters".
- Your re-written version of "sum of digits from %sentence"