

Recursion With Arrays

Add up all the elements in an array:

Iterative version:

```
int sum = 0;
for (int i = 0; i < array.size(); i++)
    sum += array[i];
```

Recursive version:

Base case: If `array.size() == 1`, return `array[0]`

Recursive case: If `array.size() > 1`:

- Compute the sum of all the elements in the sub-array from index 1 to the end → `sum(array[1:])` in Python syntax.
- Add `array[0]` to this sum from above.
- Return this value.

Code:

```
public static int sumOfArrayList(ArrayList<Integer> list)
{
    return sumOfArrayList(list, 0);
}

// Helper function for above.
public static int sumOfArrayList(ArrayList<Integer> list, int leftIdx)
{
    if (leftIdx == list.size()-1) { // if there's only one element in our "sub-list"
        return list.get(leftIdx);
    }
    else {
        int smallerSum = sumOfArrayList(list, leftIdx + 1);
        return list.get(leftIdx) + smallerSum;
    }
}
```

Find the maximum element in an array:

Iterative:

```
largest = array[0]
for (int i = 1; i < array.size(); i++)
    if (array[i] > largest)
        largest = array[i]
return largest
```

Recursive:

Base Case:

Recursive Case: