

# Pre-class Work 1:

# Programming Instructions

Implementing Data Structures

## Readings

See Ed for the background readings.

## Implement indexOf(int value) [Programming Question]

Work in the pcm1-indexOf folder (Client and ArrayIntList)

Write a new method for the `ArrayIntList` class called `indexOf` that returns the index of a particular value in the list. The method should return the index of the first occurrence of the target value in the list. If the value is not in the list, it should return `-1`. For example, if a variable called `list` stores the following values:

```
[42, 7, -9, 14, 8, 39, 42, 8, 19, -3]
```

then the call `list.indexOf(8)` should return `4` because the index of the first occurrence of the value `8` in the list is at index `4`.

Notice that we are using 0-based indexing. The call `list.indexOf(2)` should return `-1` because the value `2` is not in the list.

**Important:** The entire `ArrayIntList` class is given for reference, but you only need to implement the `indexOf` method at the top of the class.

## Implement add(int index, int value) [Programming Question]

Work in pcm1-add folder (Client and ArrayIntList)

Write a new method for the `ArrayIntList` class called `add` that takes an integer index and a value to add and that inserts the given value at the given index, shifting subsequent values to the right. For example, if a variable called `list` stores the following values:

```
[3, 19, 42, 7, -3, 4]
```

and we make the following call:

```
list.add(2, 17);
```

then it should store the following sequence of integers after the call:

```
[3, 19, 17, 42, 7, -3, 4]
```

Write this method `add(int index, int value)`.

Then, replace the old implementation of `add(int value)` with a single call to `add(int index, int value)`.

**Important:** You may assume that the passed in index is valid. We've provided exception code in the scaffold that takes care of this. We'll talk more about exception checking later!

**Important:** The entire `ArrayIntList` class is given for reference, but you only need to implement the `add` methods at the top of the class.

