



Foo is a wrapper class for SA that  
 - maintains/imposes an invariant such that elements  
 are ordered (ascending) "

add (Add an element to a list such that  
 ascending order is preserved)

// Temp is a bit  
 redundant upon hindsight

- Parameters:

- int[] list // original list
- int newVal // new value to be added

- Return

- newList // original list w/ new value, all in ascending order

- int index = 0; // index to add newVal at

- int temp = 0; // to determine  
 location of index

- Increment temp until list[temp] > newVal or temp >= list.length

- if temp = list.length, increment it by 1 (after having run the above loop)

- Set index to temp

- Return newList as the array generated by add-at-index (int[] list, int index)

add-at-index (Foo's version of add-at-index)  
 w/ Params int[] list, int index, int newVal)

Examples:

[2, 3, 5, 7], 6

6 < 2, false

6 < 3, false

6 < 5, false

6 < 7, true

↑  
 index at 3

add-at-index

[2, 3, 5, 0, 7]

[2, 3, 5, 6, 7]

Add-at-index  
 method available  
 in ArrayList

should this just automatically  
 add it to the appropriate  
 index, not  
 considering an  
 "index" param?

[2, 3, 5, 7], 13

13 < 2 X

13 < 3 X

13 < 5 X

13 < 7 X

↑  
 index at 4

add-at-index

[2, 3, 5, 7, 0]

[2, 3, 5, 7, 13]