Team Frog Hearts

Reviewed by: We Forgot

HW 46 -- Wrap the Wrapper

add - at - index is redundant or useless. An element's index is determined by it's value,

may be a use as a helper?

so there's no reason or purpose to specify the element's index in the method.

add - at - index (9, index), index is meaningless because 9's index is always 4

| 0 | 2 | 5 | 7 | —> | 0 | 2 | 5 | 7 | 9 |

add must perform the same thing as add - at - index now.

add (2) must start at the rightmost element, compare, and walk down the array until an element in the array is equal to or less than 1.

|0|2|5|7|9| 9 > 2 |0|2|5|7|9|9| 7 > 2 |0|2|5|7|7|9 ...
^ ^ ^

What would happen if the array is empty?

|0|2|5|5|7|9| 2 ≤ 2 |0|2|2|5|5|7|9|
^

add (6)

|0|2|2|5|7|9| 9 > 6 |0|2|2|5|7|9|9| 7 > 6 |0|2|2|5|7|7|9| 5<6
^ ^ ^

|0|2|2|5|5|7|9| —> |0|2|2|5|6|7|9|

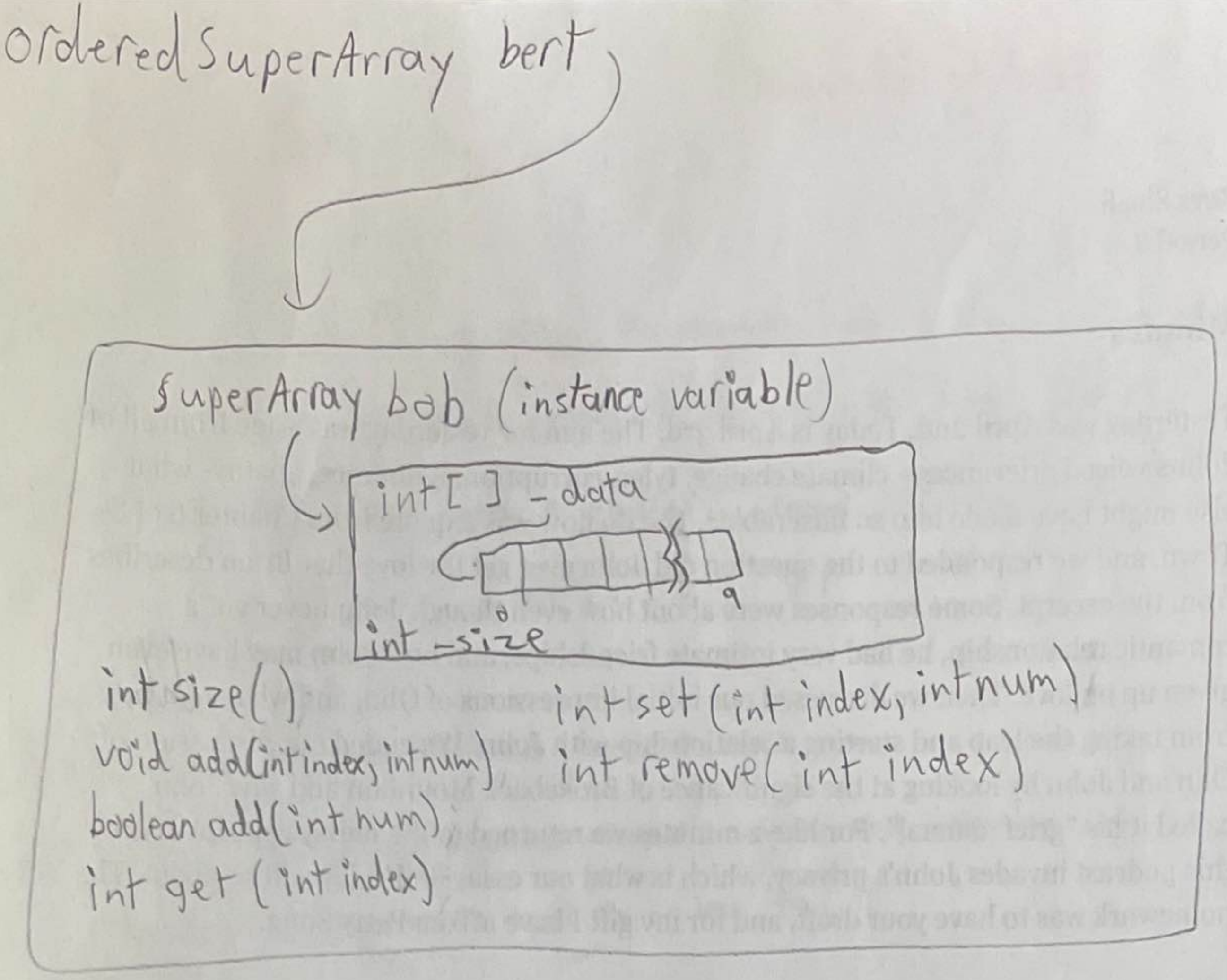
remove is unaffected. Removing an element from an index still keeps the array in numerical order. ✓

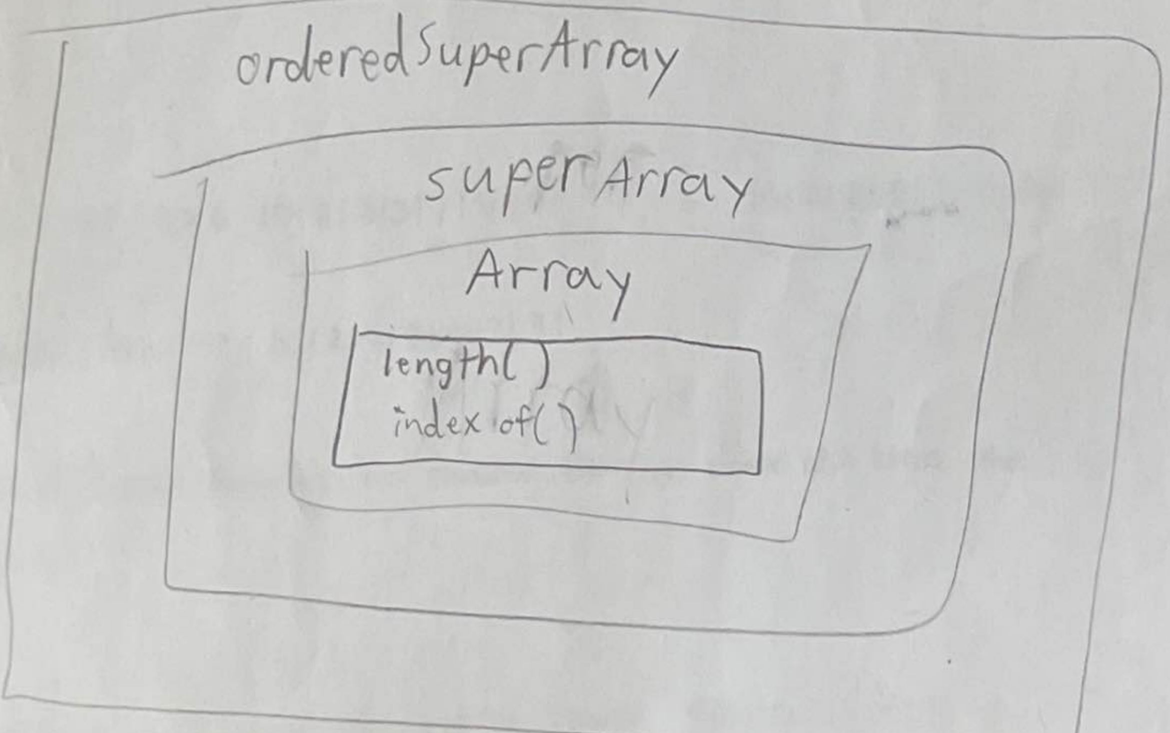
remove (5)

|0|2|2|5|6|7|9| 7 is not the rightmost element |0|2|2|5|6|9|9|
^

9 is the rightmost element |0|2|2|5|6|9|

Other methods such as set(element, index) to set an element in the array to something are also unaffected. However, it would be useful to implement an error detection system, checking if the previous number is less than or equal and the next number is greater than or equal. If the check fails, it should produce an error message. good idea


orderedSuperArray bert
superArray bob (instance variable)
int[ ] _data
0
9
int _size
int size()
int set (int index, int num)
void add(int index, int num)
int remove(int index)
boolean add(int num)
int get (int index)


orderedSuperArray
superArray
Array
length( )
index of( )

✓