Wyatt Duberstein

SE 317

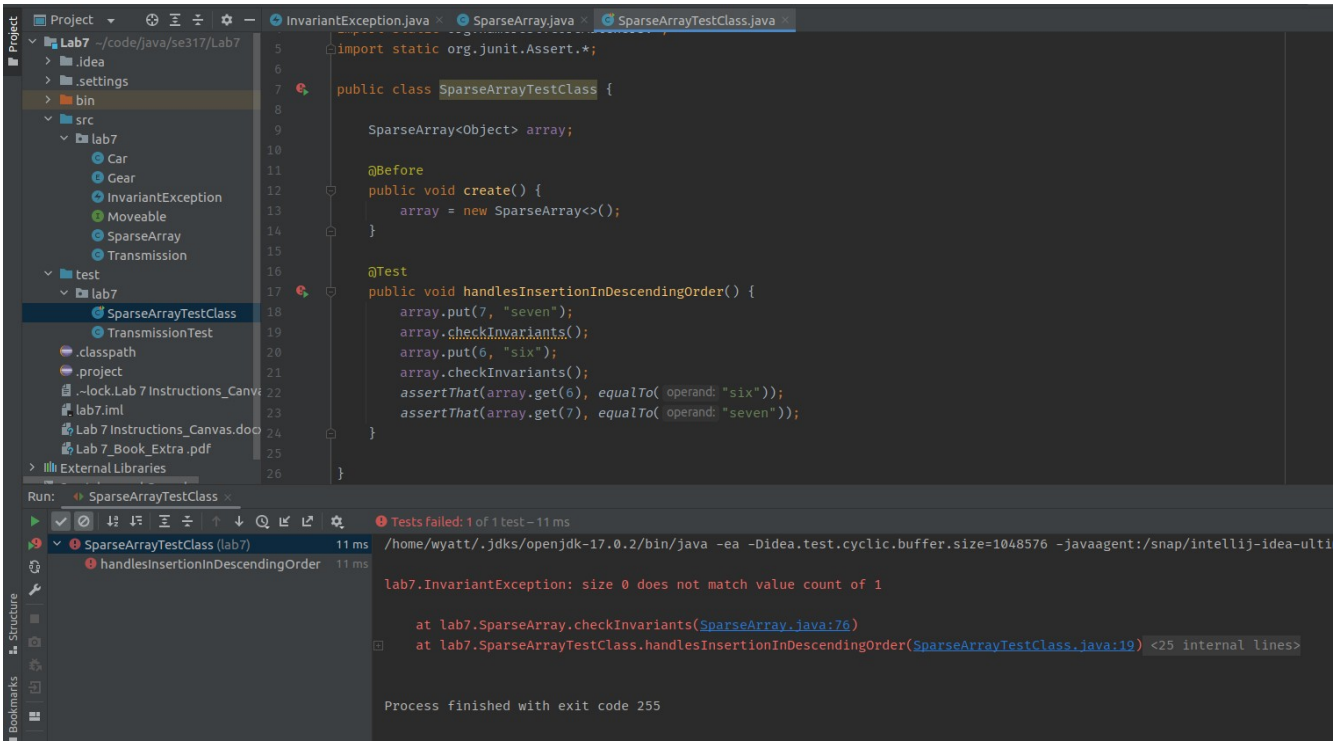
07/02/2022

Lab #7

1) Implement a binarySearch method

```
int binarySearch(int n, int[] nums, int size) {
    int first = 0, last = size, mid = size / 2;
    while (first <= last) {
        if (nums[mid] < n) {
            first = mid + 1; // Change index and try again
        } else if (nums[mid] == n) { // If number is found, return it
            return mid;
        } else {
            last = mid - 1; // Change index and try again
        }
        mid = (first + last) / 2; // Cut in half and try again
    }
    return -1; // Returns -1 if the number is not found in the index
}
```

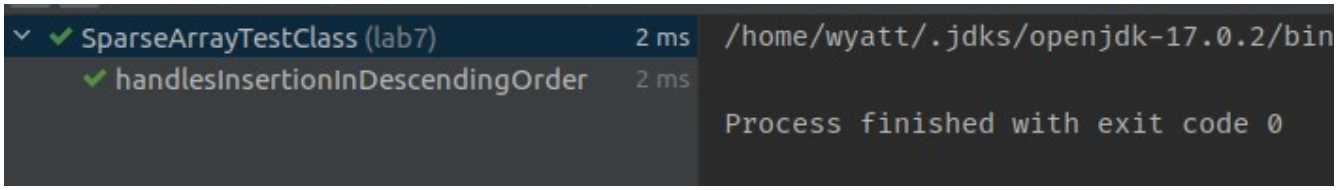
2) Add the code snippet to SparseArray.java and run the new test



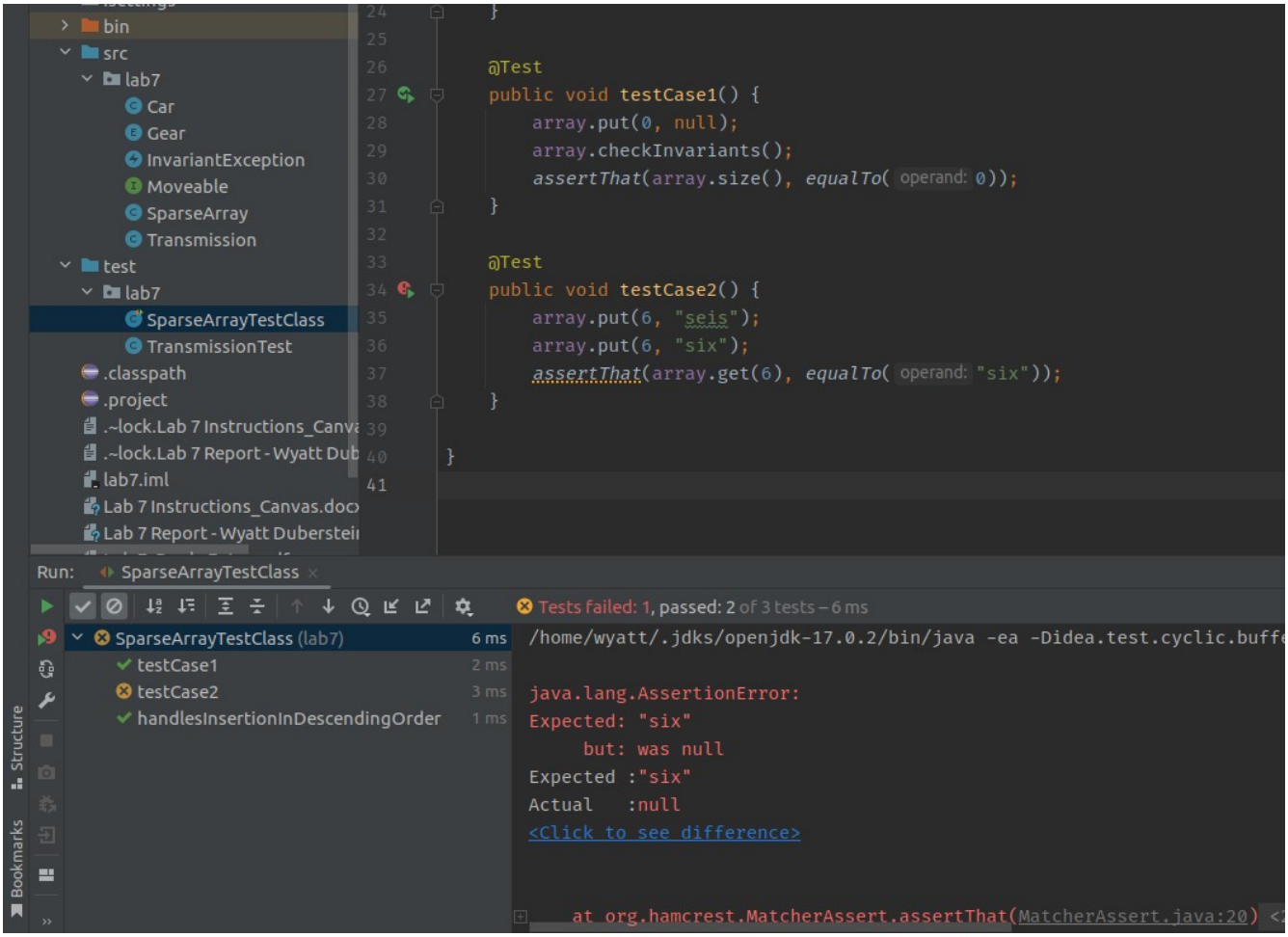
3) What is the problem? And fixed code

The problem is in the put method, the size variable of the array is not being updated properly and at the right time. I fixed it by incrementing the size variable of the array at the end of the put method which then caused the test to pass.

```
public void put(int key, T value) {
    if (value == null) return;
    int index = binarySearch(key, keys, size);
    if (index != -1 && keys[index] == key)
        values[index] = value;
    else
        insertAfter(key, value, index);
    size++;
}
```



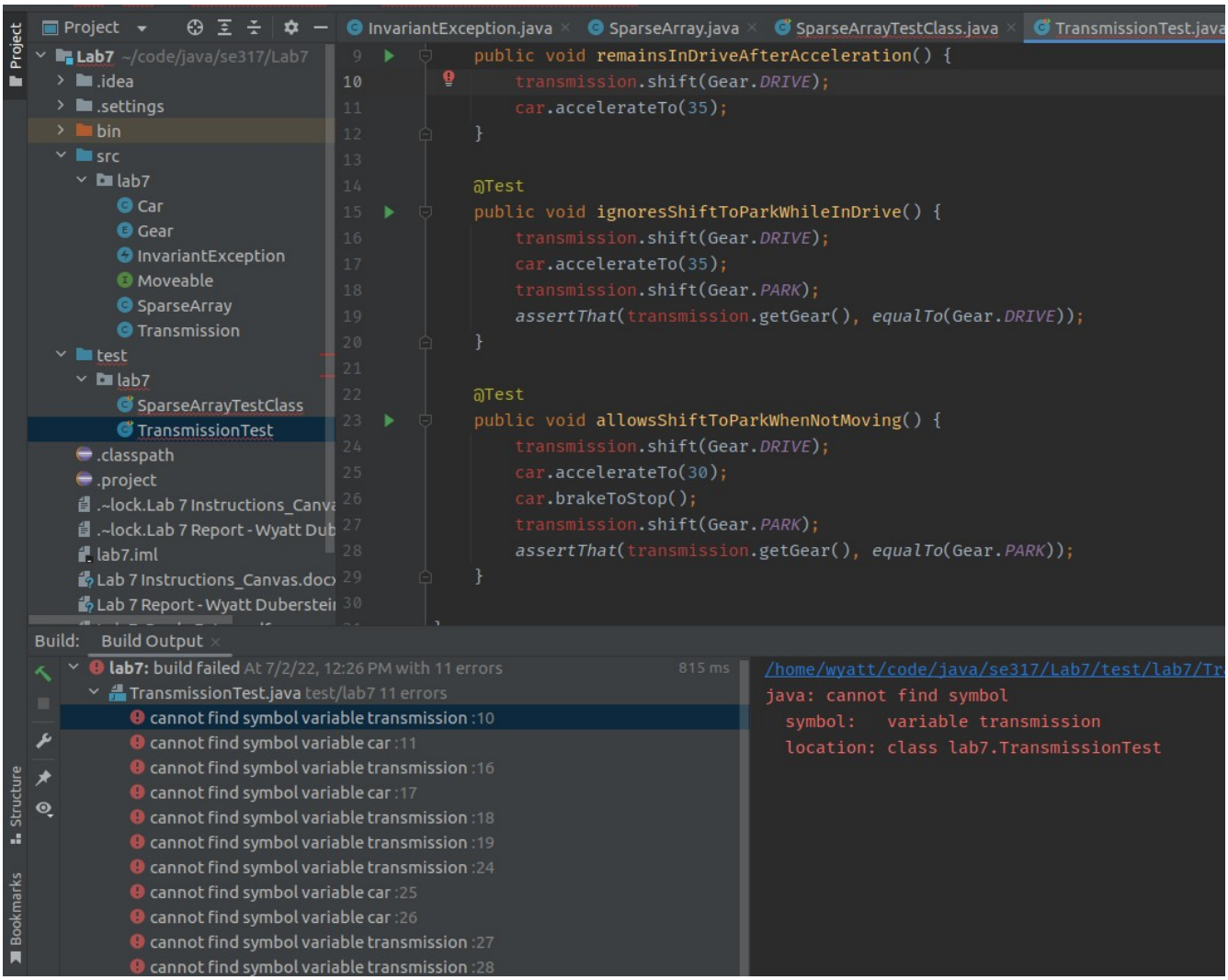
4) Write two test cases, run the code, and take screenshots of the results.



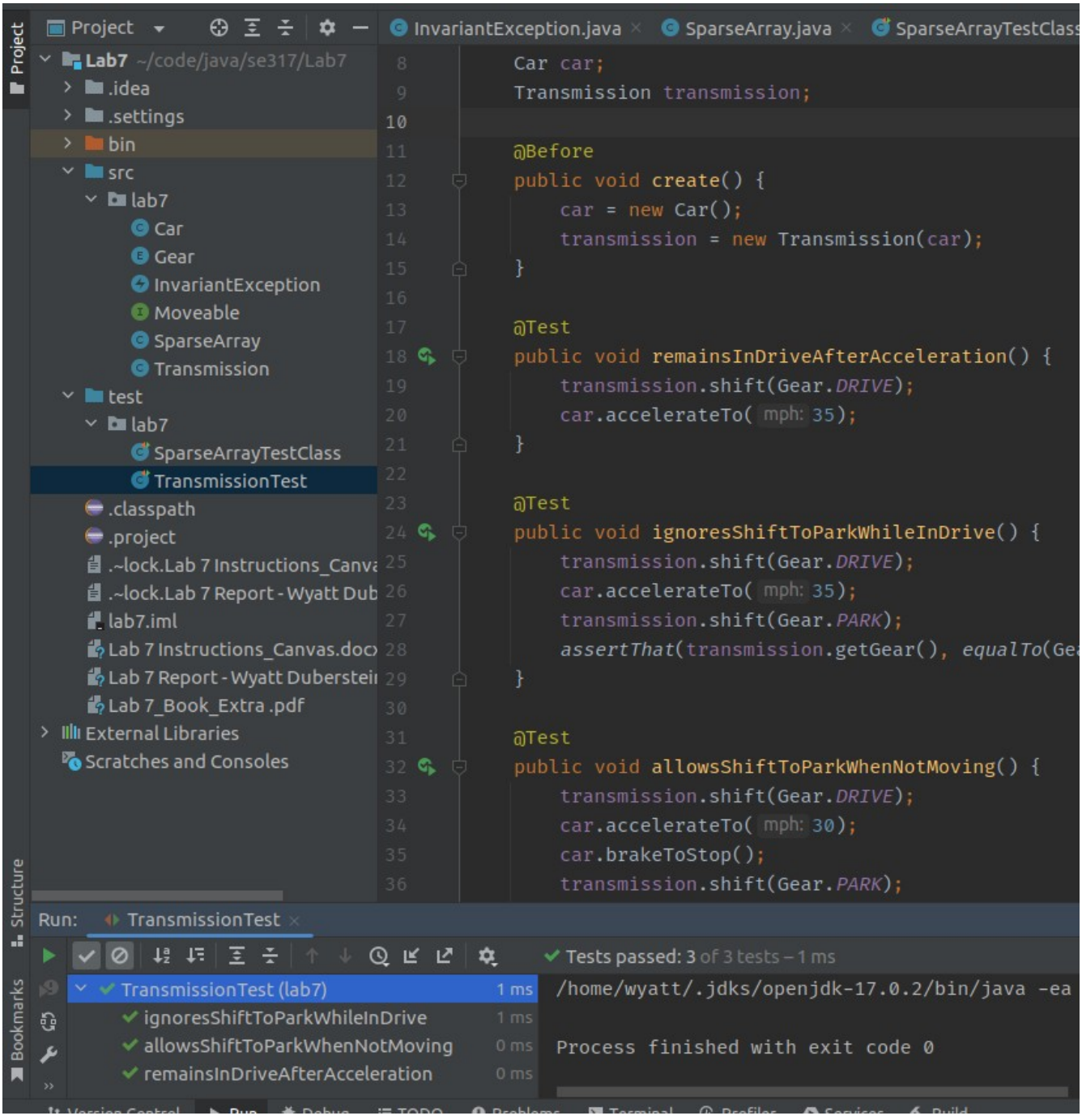
The First test case passes, but the second test case fails because in the put method, if the two if conditions are met, it enters the values into the values array but not into the keys array for some reason. As I am not required to fix this by the homework document, I will not fix this.

Part 2:

1) TODO: Inspect and run the code. Take a screenshot of the code and the output



2) This code will not run, add an @before method and a create method and post screenshot of code with completed tests:



Part 3:

1) What does the checkvariants() method do?

This method gets all of the nonNull values from the array, this includes any number that is not null or not zero. So basically It gathers any values that are in the array, then checks if the size variable for the array is equal to the amount of objects in the array. If it is not equal, then there is too many or too few objects in the array and the size variable does not accurately reflect the amount of objects in the array. This then throws an exception. If the two values are the same, then nothing happens.

2) What does the Transmission.java class do?

The transmission class represents the transmission of a moveable object, in this case a Car object. This method has 2 methods, one that returns the gear of the car, and one that shifts the gear of the car. The shift() method has a special check in it that makes sure that if the car is moving at a speed greater than 0 and is in park, then the car cannot shift.