Lab2 Report

Becky.Song

DUE 17/09

* Abstract

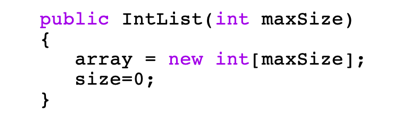
In the first part of Lab2, we write a test file that using the method of unit testing to test on all methods in the IntList class. Including getMaxSize(), getPos(), remove() and insertBefore() . By using the test file, I write, the second part is to find out errors in the little Timmy’s code and correct them. The goal is to make sure little Timmy’s code can pass all the assertions in the test file.

* Error 1

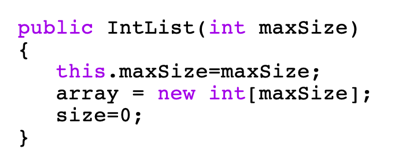


In the test file, I created a new intList with max size of 10. The getMaxSize() method should return 10 instead of 0.

The constructor code is :

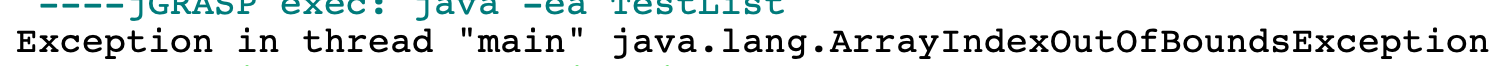
The previous code claim that the default max size is 0. In the constructor, the value of the max size should change to the input. However the original constructor didn’t change the value of the max size.

The fixed code is :

Add “this.maxSize = maxSize”, the value of the max size can be reset.

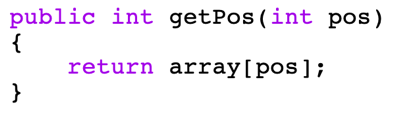
“ this ” refer to the new object of the IntList.

* Error 2

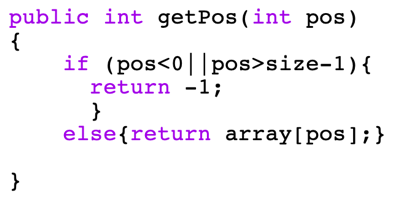


I created a list: [1,2,3,4,5] with the max size of 5. The assertion is that when I want to get value on the index of 5 the list, it should return -1 because the max index is 4 and index 5 is out of range.

The original code is :

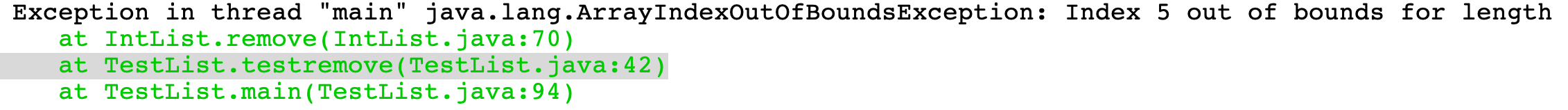
The code simply returns the value on the required position with out limitation.

The fixed code is:

With if and else statement, the getPos method can determain if the pos input is valid. When the pos is smaller than 0 (the position doesn’t exist) or it bigger than size-1 (since the index starts at 0, the max index is one less than the actual size of the list), it returns -1.

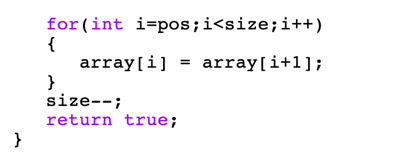
Else, it will return the value on the position.

* Error 3

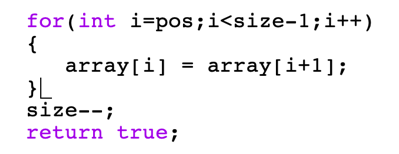


I created a list [1,2,3,4,5] and want to remove the value of the index 1 and the rest of the list should move one index and become [1,3,4,5]. The error is out of bounds.

Original code is:

The logic of the code is that every value before the input position should stay the same as before. So the int i starts from the pos. And every value after the input position should move to the position before the current position. If the loop stop at when i equals size-1, i+1 will equal to the original size. However array[i+1] is out of range because the max index is size-1.

Fixed code is:

Change the limitation of the loop from i<size to i<size-1. In this way, the last value of the new list will equal to the last value of the old list.

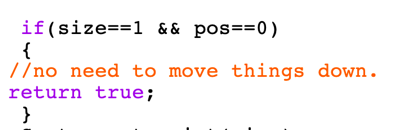
* Error 4

图片包含 屏幕截图

描述已自动生成

Here I created a new list with max length of 1 and insert a number into it. When remove the only number in the list and check the length of it, should return 0.

The original code is:

When the list size is 1 and remove the value on index 0, the if statement simply return true but change nothing at all.

The fixed code is:

When the if statement is true, the size should decrease one.

* Error 5

图片包含 屏幕截图

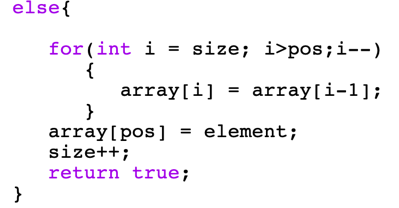
描述已自动生成

I created a list with 5 max size but only fulfilled 4 position which is [1,2,3,4] and then I insert number 7 at position2. The new list should be [1,2,7,3,4]. Using getPos method, the value on the position 3 is 3, not the original 4.

The original code is:

The logic is similar as the remove method, values before the insert position stays the same, and values after the insert position should slide to the position after the current position. In the original code if set i<pos, the command changes values before the input position but not values after the input position.

The fixed code is:

Changing i<pos to i>pos, so that command can shift values after the input position.