

Tafita Rakotozandry  
CS150 Lab  
08/26/2020

## **Lab 2 Report**

### **INTRODUCTION**

The goal of this lab is to familiarize with random number generation in Java , code execution time measurement and Unit testing. In this lab, we created two classes: RandomIntegersContainer and ExperimentController. The RandomInteger creates an array of length 10. It is possible to integer in that array using addToBack,addToFront and addAt methods. The class ExperimentController is used to measure the execution time of these three methods.

### **Unit Tests**

The method addToBack,addToFront,addAt and numEntries are tested one by one . The method addToBack,addToFront and addAt are tested two times because there are cases where the size of the array is enough to insert a new value and there are other cases where the size is too small to insert new values. In that case, the array is doubled in size. Each of one of these cases were unit tested. The screenshot of the unit test is below.

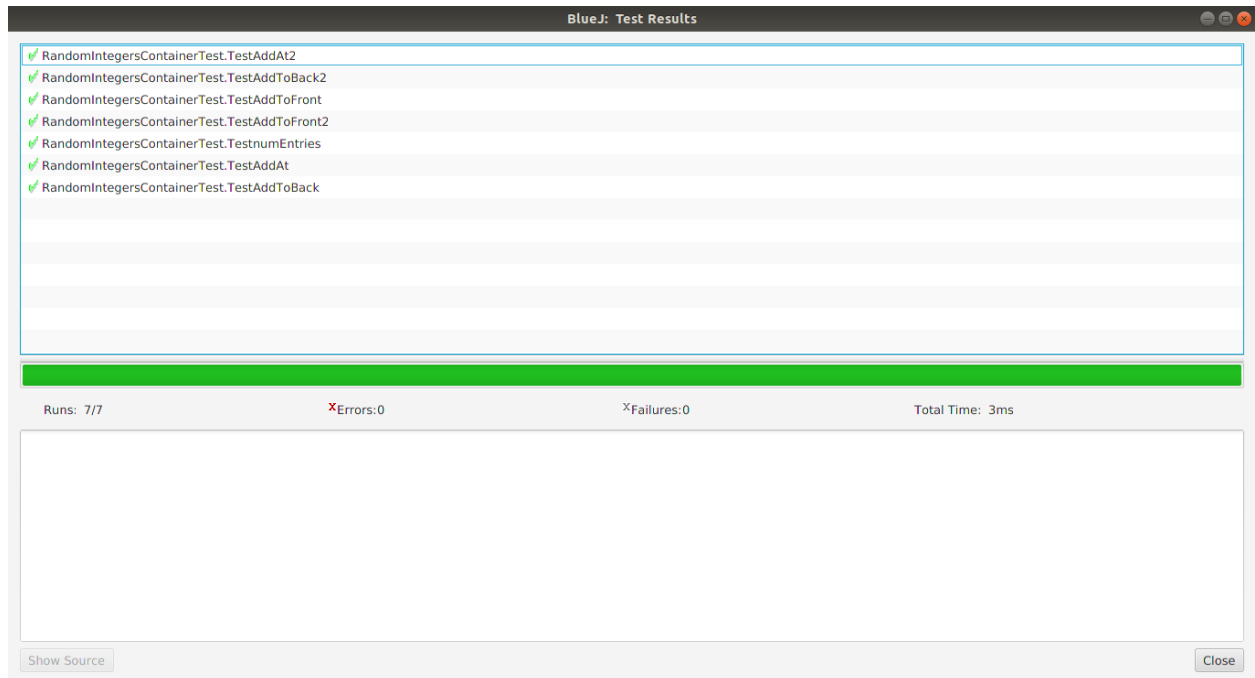


Figure 1: Screenshot Unit Testing Window

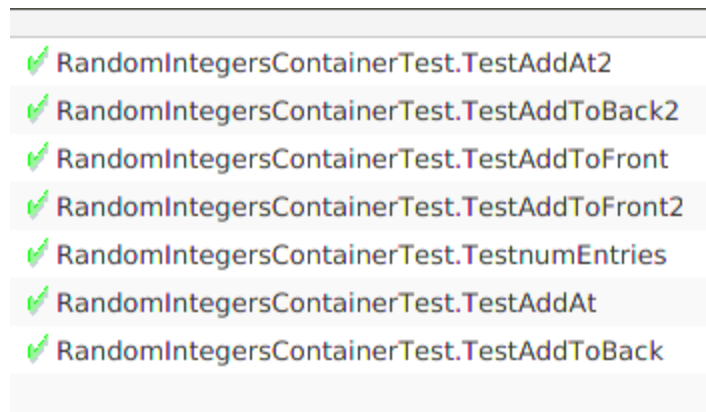


Figure 2: Zoom of the Unit Testing Window

## Required Output

The time spent by the method `addToBack`, `addToFront` and `addAt` were measured by taking the difference between the final time that the method was executed and the initial time that the method was called. Several combinations of seeds and item numbers were used to realize the

measurement. The result was obtained by the time average of the different measurements. The graphs below represent the result of the test

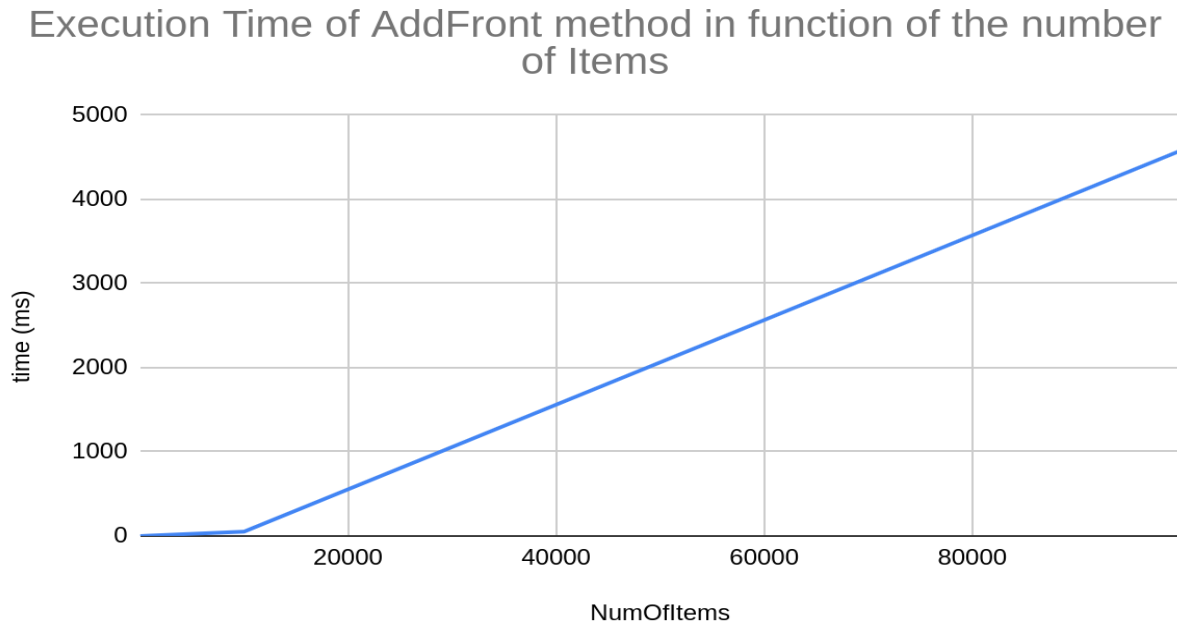


Figure 3: Execution time AddToFront method in function of the item number

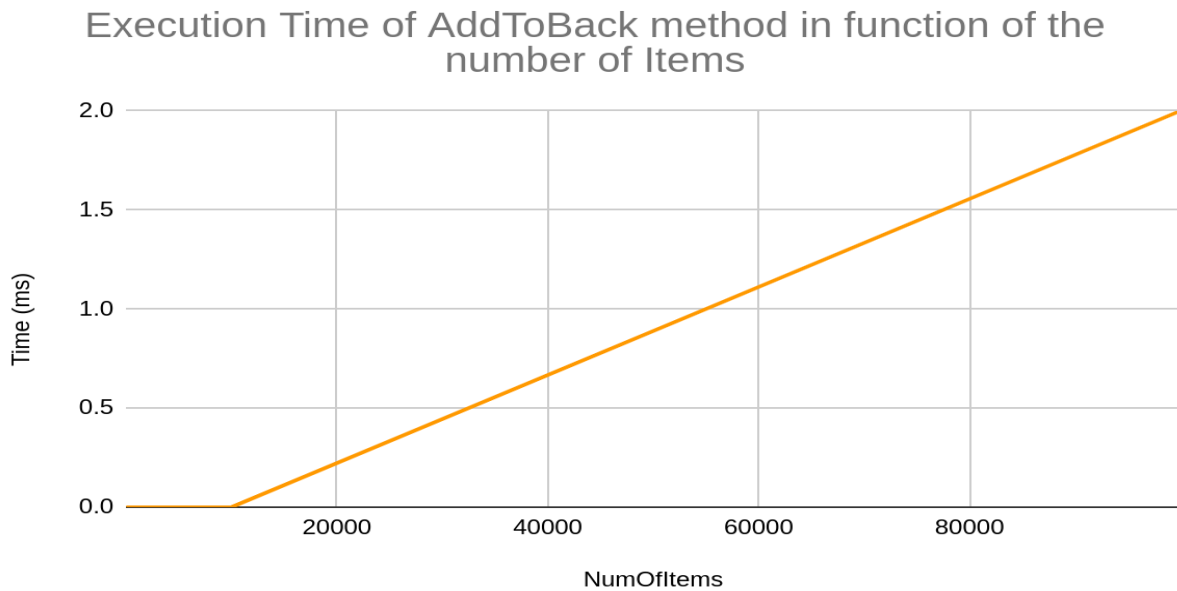


Figure 4: Execution time AddToBackt method in function of the item number

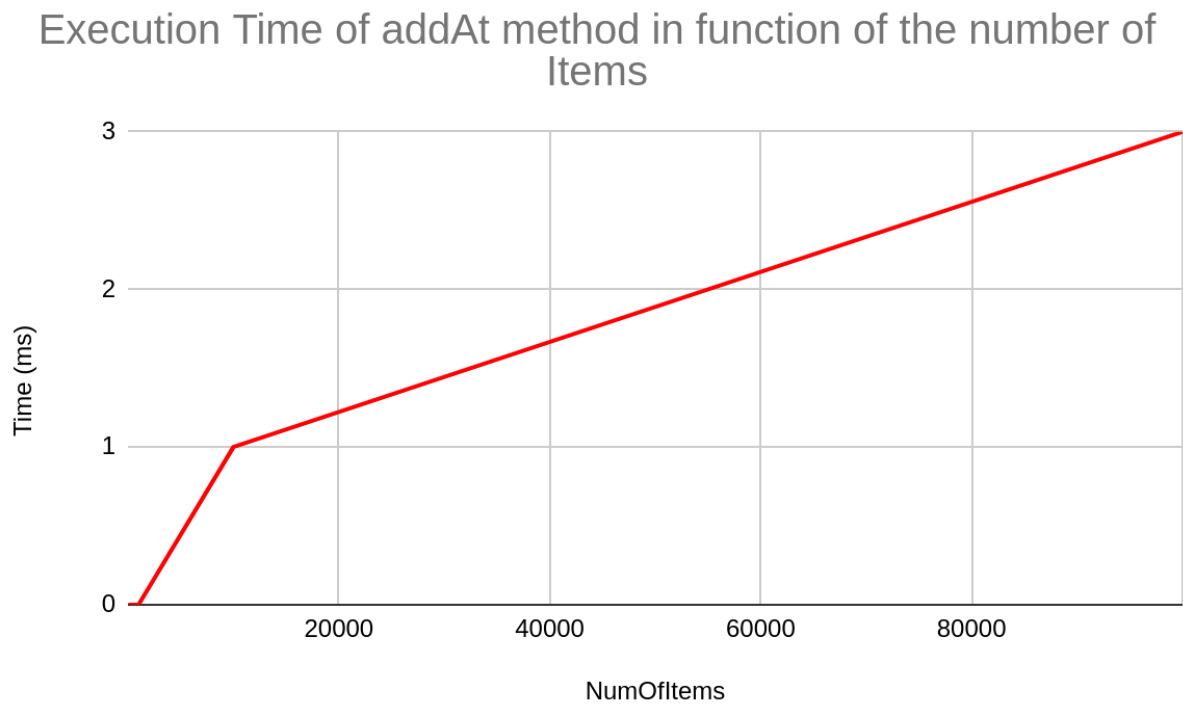


Figure 5: Execution time AddAtt method in function of the item number

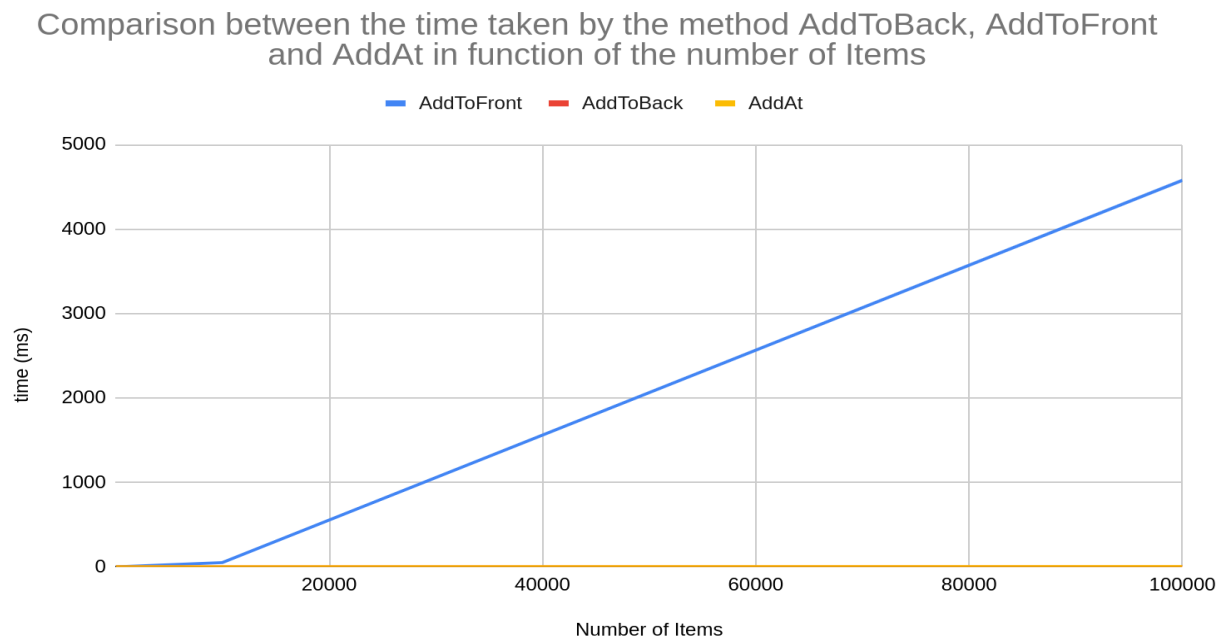


Figure 6: Graph Comparing the execution time of the method addtoBack,addToFront and addAt

### Comparison between the time taken by the method AddToBack, AddToFront and AddAt in function of the number of Items

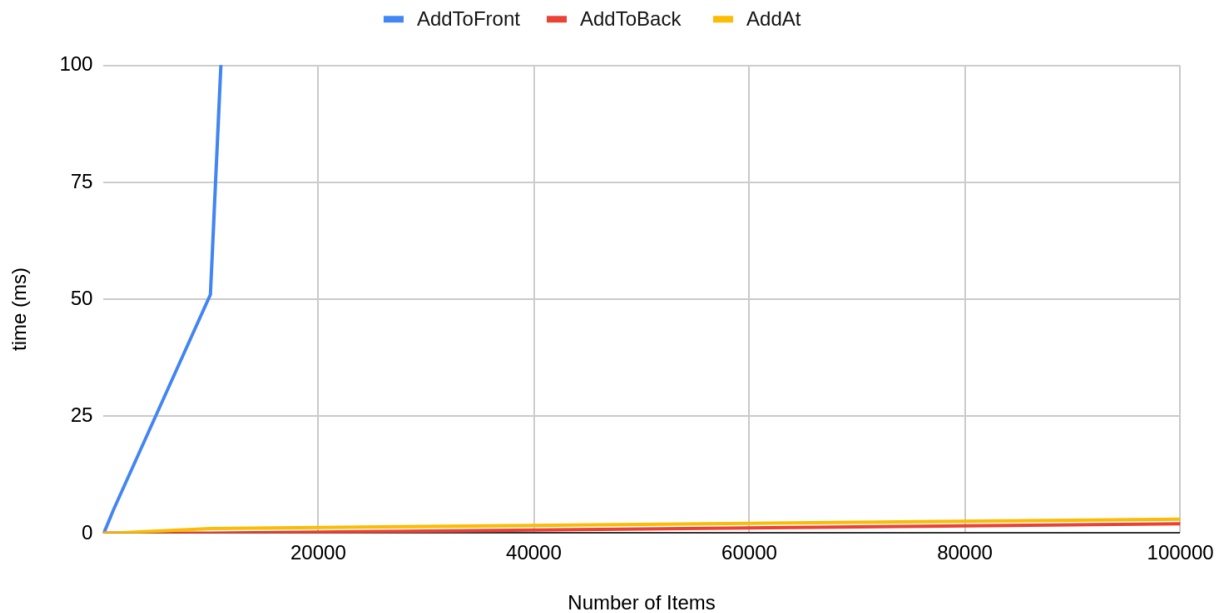


Figure 7 : Zoomed version of the graph Comparing the execution time of the method addtoBack,addToFront and addAt

**Interpretation :** The method addToFront takes much more time than the other two methods when performing an insertion. It is probably due to the fact that when inserting an item in front, the actual values in the array are still copied in another array. After the insertion of the new value, the rest are appended one by one. It is time consuming.

### Trouble Report

N/A

## References

*Oracle. (2019, May 12). Random. Oracle.Com.*

<https://docs.oracle.com/javase/8/docs/api/java/util/Random.html>

*Michael Kölling (2015). Unit Testing in BlueJ <https://www.bluej.org/tutorial/testing-tutorial.pdf>*