

**CMPSC 112**  
**Introduction to Computer Science II**  
**Fall 2016**

**Laboratory Assignment Three: Evaluating the Performance of Array Reversal**

## Introduction

The Java programming language allows for the implementation of methods that are written in a “generic” fashion so that they can process a wide variety of input types. In this laboratory assignment, you will investigate the use of Java’s casting and generics features. Additionally, you will learn more about Java’s support for nested classes and the way in which these classes support the implementation of elegant programs. Moreover, you will continue to practice conducting experiments that evaluate the performance of an algorithm implemented in Java. In particular, you will explore how to use random number generation to better support the performance evaluation of algorithms that process large input arrays. Finally, you will learn how to use new software engineering tools, such as Ant build systems, that enable you to compile and run large systems.

## Reading Assignment

To start your review of the key features of the Java programming language, please study the material in Chapter 1 of the textbook, paying particularly close attention to the content about iteration and software development in Sections 1.5 and 1.9. To further prepare for this assignment, you should also read Sections 2.5 and 2.6, making sure that you understand how the Java programming language supports generic methods and nested classes. Please also review the slides that we have discussed during our recent class sessions. If you have questions about this reading assignment or the material that was presented in class, then please see the instructor or a teaching assistant. If done appropriately, you may post your question to the #laboratory channel of our Slack team.

## Configuring Git and Bitbucket with your Partner

During this laboratory assignment, you will securely communicate with the Bitbucket.org servers that will host all of our projects. If you are still having trouble using Bitbucket, you should refer to the following web site for additional information: <https://confluence.atlassian.com/display/BITBUCKET/Bitbucket+101>. For this assignment, you are responsible for working with a partner to implement all of the required programs, conduct the necessary experiments, and write a report of your results; you may pick your own partner, provided that you do not work with the same person as you did from a previous assignment. Once you have chosen your partner and collaboratively reviewed and discussed the entire laboratory assignment sheet, please create a Bitbucket repository according to the naming convention `cs112F2016-lab03-<user name one>-<user name two>`.

## Building, Running, and Enhancing a Java Program

In a previous laboratory assignment, you implemented and empirically evaluated the performance of an algorithm for reversing the contents of an array. In this laboratory assignment, you will study a substantial extension of the `GenericsDemo` example on page 95 of your textbook that reverses a

“generic” data array of type `T[]`. Instead of requiring you to start this assignment “from scratch” I have begun an implementation and placed it in the “share” repository for our course. To start this assignment, please use your terminal window to navigate to the share repository and then run the “`git pull`” command. You will notice that this type the source code is organized into the `src` directory and there is an Ant `build.xml` file that allows for the automated building and running of the provided source code. To use this file, you can type “`ant compile`” in your terminal window. At this point, you should see that it produces output similar to that which is shown below:

```
Buildfile: /home/gkapfham/working/teaching/cs112F2016/src/mine/labs/lab3/build.xml
compile:
[javac] Compiling 3 source files to /home/gkapfham/working/teaching/cs112F2016/...
BUILD SUCCESSFUL
Total time: 1 second
```

Once you have been able to successfully compile this program, please take time to review the source code of provided Java files, making sure that you and your partner can draw a diagram to explain how they are related. You should also carefully study the `build.xml` file so that you can learn about all of the features that it provides. Now, you are ready to run the program by typing the command “`ant run`” in your terminal window. What output does this produce? What is the meaning of this output? You and your partner should examine this output, discussing it so as to ensure that you both understand its meaning. Even though this program is designed to perform an experiment to evaluate the performance of array reversal as implemented in the `public static <T> void reverse(T[] data)` method, you will notice that it does not currently produce any output so that you can verify that the reversal is working correctly! To ensure that the program reverses arrays in the right way, you should modify some of the existing code so that it produces debugging output for arrays that are small enough so that you can check the correctness of their reversal. Is the program correctly generating and reversing the arrays? If not, you must fix the provided source code. Please see the course instructor if you are stuck on this debugging step.

After you have studied the output, you and your partner will have noticed that it runs the method under study for multiple trials and reports arithmetic means and standard deviations of the recorded timings. For this assignment, you can use the standard deviation values as a way to characterize the amount of dispersion in your results. Intuitively, if the timings are very dispersed, then this is an indication that the execution of some background processes may be “contaminating” your results. Now, draw your attention to the fact that the program does not currently calculate a standard deviation for all of the data that it collects — you and your partner should add in calls to the `public static double calculateStandardDeviation(Long[] timings)` in every location where you think it is needed. Now, use the Ant build system to re-compile and re-run the `ReverseArrayExperiment` to ensure that it is producing the intended output. Finally, you should add in additional code to this program so that it uses classes like `java.text.DecimalFormat` to format the standard deviations in the output to only display two decimal places. Please see the course instructor or a teaching assistant if you and your partner cannot complete this step.

You will also notice that, as it is currently implemented, the `ReverseArrayExperiment` performs a performance evaluation of array reversal with arrays of type `Integer` and `Float`. To complete this task, it uses the methods in the `RandomArrayGenerator` class to automatically — and randomly — create `Integer` and `Float` arrays of a specified size. You and your partner

should also study this Java class to ensure that you understand how it works. Now, you should extend it so that it also generates random instances of the `CreditCard` class that have random values for the `lim` and `balance` parameters to the class's constructor. Finally, you will recall that the `CreditCard` class from a prior laboratory assignment was placed in a Java package. To ensure that this Java class correctly integrates with all of the other provided source code, you should place the `ArrayPrinter`, `RandomArrayGenerator`, and `ReverseArrayExperiment` classes in the `edu.allegheny.experiment` package, adjusting the source code and the directory structure accordingly. Please see the instructor or a teaching assistant if you have questions about this step.

## Evaluating the Performance of Array Reversal

Before starting to conduct the experiment for this laboratory assignment, please ensure that you and your partner understand how the provided source code uses the nested class, called `StatisticsCalculator`, to calculate the arithmetic mean, variance, and standard deviation of the timings that result from running multiple trials of the experiments. Can you write an equation for these three statistical values? If not, then please see the course instructor or a teaching assistant for help with this task. Now, you are ready to experimentally answer two research questions.

*RQ1: How does the data type of an array influence the performance of the array reversal method?* To answer this research question, you should run the `ReverseArrayExperiment` class for a suitable number of trials and record the timings when the array reversal method accepts arrays of type `Integer`, `Float`, and `CreditCard`; you and your partner are also encouraged to investigate other data types as you deem appropriate. Your response to this research question should leverage the timings and their arithmetic means and standard deviations, particularly focusing on which array data type leads to the fastest and least-variable method for array reversal. Whenever possible, you should note the key trends and, additionally, explain why they are evident in your data set.

*RQ2: Is it faster to reverse an array using a "hand-coded" method or the one furnished by the Java language?* To answer this research question, you should run the `ReverseArrayExperiment` class for a suitable number of trials and focus primarily on whether it is faster to reverse an array using the `reverse` method provided by this class or the standard method available by calling `Collections.reverse`. Your response to this research question should leverage the timings and their arithmetic means and standard deviations, particularly focusing on which algorithmic technique leads to the fastest and least-variable method for array reversal. Whenever possible, you should note the key trends and, additionally, explain why they are evident in your data set.

## Carefully Review the Honor Code

The Academic Honor Program that governs the entire academic program at Allegheny College is described in the Allegheny Academic Bulletin. The Honor Program applies to all work that is submitted for academic credit or to meet non-credit requirements for graduation at Allegheny College. This includes all work assigned for this class (e.g., examinations, laboratory assignments, and the final project). All students who have enrolled in the College will work under the Honor Program. Each student who has matriculated at the College has acknowledged the following pledge:

I hereby recognize and pledge to fulfill my responsibilities, as defined in the Honor Code, and to maintain the integrity of both myself and the College community as a whole.

It is understood that an important part of the learning process in any course, and particularly one in computer science, derives from thoughtful discussions with teachers and fellow students. Such dialogue is encouraged. However, it is necessary to distinguish carefully between the student who discusses the principles underlying a problem with others and the student who produces assignments that are identical to, or merely variations on, someone else's work. While it is acceptable for partners in this class to discuss their programs, data sets, and reports with their classmates, deliverables that are nearly identical to the work of others will be taken as evidence of violating the Honor Code.

## Summary of the Required Deliverables

This assignment invites you to submit both a printed and an electronic version of these deliverables:

1. Using the Javadoc standard, a commented version of the three enhanced Java classes.
2. The output from running the final version of the `ReverseArrayExperiment` program.
3. A short report that responds to these prompts about the provided Java classes:
  - (a) What is the purpose of creating and using an Ant build system?
  - (b) What is the relationship between the three Java classes used to run the experiments?
  - (c) How does the Java language support the implementation of the `StatisticsCalculator`?
  - (d) What is the meaning of the signature for the `reverse` method in `ReverseArrayExperiment`?
  - (e) What is the purpose of the "static initializer" in the `RandomArrayGenerator` class?
  - (f) How do the three Java classes support the effective display of debugging information?
4. A short experiment report that comments on the timing experiments that you conducted. Your experiment report should include, at minimum, the following deliverables:
  - (a) An overview of how you collected timing information with the `ReverseArrayExperiment`.
  - (b) A description of the steps that you took to conduct the experiments and collect timings.
  - (c) Tables of data that support your responses to the two aforementioned research questions.
  - (d) A commentary on the observed empirical trends and the reasons why they are evident.
  - (e) A statement of the threats to the validity of the conclusions drawn from experimentation.
  - (f) A list of the challenges you faced when running the experiments or analyzing the results.
5. A short commentary on the challenges you faced when enhancing the provided source code.
6. A one-page document that explains the tasks completed by each member of your team.
7. An individually submitted document that reports on the work experience with your partner.

Before you turn in this assignment, you must ensure that the course instructor has read access to your Bitbucket repository that is named according to the convention `cs112F2016-lab03-<user name one>-<user name two>`. Please note that each team in the class is responsible for submitting one version of this assignment. Students who have questions about any aspect of this laboratory assignment, including how they should complete it under the structure of the Honor Code, are encouraged to schedule a meeting during the course instructor's office hours. Students are also invited to post questions or comments about this laboratory assignment to the `#laboratory` channel in our Slack team; either a teaching assistant or the instructor will answer these questions.