CMPSC 111
Introduction to Computer Science I
Spring 2017

Practical 9
April 28, 2017
Due in Bitbucket by midnight on the day of your practical
"Checkmark" grade

Summary

In this practical assignment, you will explore an already implemented Java program that causes various types of exceptions to be thrown. As you modify the program, you will uncomment certain sections of it and then observe, record, and comment on the output that is produced. In addition, you will add some source code to the program that causes a new type of exception to be thrown. Finally, you will continue to practice using a Git repository hosted by Bitbucket.

Review the Textbook

To learn more about the concepts associated with exception handling in the Java programming language, please study the content in Chapter 11. In particular, make sure that you understand the purpose of exceptions, the messages displayed in the terminal when an exception is uncaught, and the structure and meaning of the `try-catch-finally` blocks used in Java programs.

Using and Extending a Program that Throws Exceptions

Please return to the "share" repository for this course and type the command "`git pull`". Now, please find the "`practical09/`" directory and study this source code. After reviewing the provided source code file, please compile and run the "`ExceptionExample.java`" program. At the outset, you will see that this program does not throw any exceptions at all. So, what you should do next is to incrementally uncomment each of the calls to the `throwsExceptions` method and observe and record the output from the program. The idea is that you will uncomment a method call in `main`, compile and run the program, observe and understand the program's output, comment out that line again, and then move onto the next line of code in the program's `main` method.

Each time the program throws a different exception, make sure that you understand why it does so. Next, you will notice that, in certain cases, the program does not output a full "stack trace" that prints out in the terminal window which exception is thrown. As such, you should add code to the catch blocks of `ExceptionExample` that can print the stack trace. Next, you must add a new code segment that will cause the program to throw a different type of Java exception. Please submit the modified, and fully commented, version of the `ExceptionExample` in your own Git repository. Finally, you must create a file called `exception_output` that records the output from your various runs of the program; please include a run showing your new exception.

General Guidelines for Practical Sessions

- **Submit *Something*.** Your grade for this assignment is a "checkmark" indicating whether you did or did not complete the work and submit something to the Bitbucket repository.
- **Update Your Repository Often!** You should `add`, `commit`, and `push` your updated files each time you work on them, always including descriptive messages about each code change.