CS 3251: Intermediate Software Design
Programming Assignment 2

This assignment builds on your `SimpleArray.h` file from Assignment 1. This assignment focuses on using templates, overloading operators and the use of `const`.

You were provided with two files: `main.cpp` and `CMakeLists.txt.` You are to create a `ScopedArray.h` (based on your `SimpleArray.h`) file that will allow you to create an array of any data type. This can be achieved by templatizing your class.

Make sure you pay attention to the driver file. Even though your `ScopedArray`'s functionality is very similar to `SimpleArray`, its appearance has changed a little. Thus, the driver file looks different as well. In addition, the new driver tests more control paths in your class. An analysis of the driver should allow you to determine which operators require overloading and how `const` should be used.

All students are **required** to use the CS 3251 coding standard, available on the course webpage and provided to you on the first day of class. Points will be deducted for not following the coding standard. Points will be deducted for not following the coding standard, as is clearly stated in the CS 3251 Syllabus.

The assignment is to be submitted using the CS 3251 Github organization with an individual repository for each student. The assignment must be committed and pushed to Github by the assigned date, so **DO NOT email** your assignment to the professor or the TAs. Emailed assignments **ARE NOT** accepted.

All students, both graduate and undergraduate are to complete the same tasks for this assignment. There are no extra tasks for graduate students on this assignment.

REMINDERS:

1. Students are **not** permitted to use Visual Studio for your programming assignments.
2. **Ensure** that your name and the honor code appear in the header comments of all files that are to be graded.
3. When submitting the assignment, all files that are provided to you and your `ScopedArray.h` file are to be submitted. All files necessary to compile and run your program **must** reside in the Github repository. Any missing files or failure to compile will result in lost points.