Critique of Project 2 for Eric Wagner, ewagner@cs.uic.edu

Written by Arnold Lee, [temp563@cs.uic.edu](mailto:temp563@cs.uic.edu)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*8

**The Readme file**

* Plus:
  + The readme file properly introduce the function/purpose of each source code file
  + Describes the syntax of invoking each program, along with optional parameters
  + Listed defects or flaws of the program that are known, along with the reason for it
  + Properly cited the original author of the BigInt class
  + Briefly describes what to expect in the results
* Minus:
  + Bold heading should be used for each paragraph,
    - i.e. **Program 2:** asciiToDecimal.cpp
  + There should be at least 1.5 to double space between each line
  + List each possible argument combination/correct & *complete* syntax in its own line (Do not start those sentences with or, and etc.)
    - For example:
      * publicPrivateGen prime1 prime2 (-o filename)\*)
      * publicPrivateGen –c –o filename
    - In the above line, do not ignore “publicPrivateGen”, and do not insert “or”, “and” in front. This will confuse the users about the correct syntax to use.
* **Verdict: 4.0 / 5.0**

**Programming style**

* Plus:
  + Extensive description for each function
  + Extensive description for each source code file
  + Every function has properly declared
  + self-descriptive variable name
  + inline comments used in critical points of the function (i.e. when an important computation is followed)
  + indentation of loops and conditional statements
* Minus:
  + No identification information on top (file name, and your name)
* **Verdict: 5.0 / 5.0**

**Data Structure and Algorithm**

* Plus:
  + Nice way of checking arguments by comparing file names
  + Did not spot inefficient algorithms
  + Use of BigInt class
* Minus:
  + An alternative of BigInt is to use exponential factorization
  + use of pointers (should be avoided)
    - temp1 = new char [file2.size() + 1];
  + Alternative:
    - string temp1;
    - temp1.resize(file2.size()+1);
* **Verdict: 4.5 / 5.0**

**Code Structure and Features**

* Plus:
  + Indentation for loops
  + comments before entering each loop
  + use of temporally loop variables
  + Well defined operation for each sub-function, i.e. reading files, file name input, prime number verification etc.
  + no global variables
  + One source code file for each major function (ASCII -> Integer, Integer -> ASCII, and key generation
* Minus:
  + Can’t think of any
* **Verdict: 5.0 / 5.0**

**Overall Comment:**

Each function is accompanied by a comment before it, with meaningful variable names, and clear formatting for each conditional statement and loops make it very easy to edit or to debug if it can compile with a GUI compiler. The readme file needs to be properly formatted, however.

**Overall Verdict: 4.5 / 5.0**