Armando Minor

Code Check List

Assignment 2

**Code Review Checklist – Java**

1. Specification / Design

[x] Is the functionality described in the specification fully implemented by the code?  
[x] Is there any excess functionality in the code but not described in the specification?

2. Initialization and Declarations

[x] Are all local and global variables initialized before use?  
[x] Are variables and class members of the correct type and appropriate mode

[x] Are variables declared in the proper scope?  
[x] Is a constructor called when a new object is desired?  
[x] Are all needed import statements included?

3. Method Calls  
[x] Are parameters presented in the correct order?  
[x] Are parameters of the proper type for the method being called?  
[x] Is the correct method being called, or should it be a different method with a similar name?

[x] Are method return values used properly? Cast to the needed type?

4. Arrays  
[x] Are there any off-by-one errors in array indexing?  
[x] Can array indexes ever go out-of-bounds?  
[x] Is a constructor called when a new array item is desired?

5. Object Comparison  
[x] Are all objects (including Strings) compared with "equals" and not "=="?

6. Output Format  
[x] Are there any spelling or grammatical errors in displayed output?  
[x] Is the output formatted correctly in terms of line stepping and spacing?

7. Computation, Comparisons and Assignments  
[x] Check order of computation/evaluation, operator precedence and parenthesizing  
[x] Can the denominator of a division ever be zero?  
[x] Is integer arithmetic, especially division, ever used inappropriately, causing unexpected truncation/rounding?  
[x] Check each condition to be sure the proper relational and logical operators are used.  
[x] If the test is an error-check, can the error condition actually be legitimate in some cases?

[x] Does the code rely on any implicit type conversions?

8. Exceptions

[x] Are all relevant exceptions caught?  
[x] Is the appropriate action taken for each catch block?

[x] Are all appropriate exceptions thrown?

9. Flow of Control

[x] In a switch statement is every case terminated by break or return?  
[x] Do all switch statements have a default branch?  
[x] Check that nested if statements don't have “dangling else” problems.  
[x] Are all loops correctly formed, with the appropriate initialization, increment and termination expressions?  
[x] Are open-close parentheses and brace pairs properly situated and matched?

10. Files

[x] Are all files properly declared and opened?  
[x] Are all files closed properly, even in the case of an error?

[x] Are EOF conditions detected and handled correctly?  
[x] Are all file exceptions caught?