Bailey Garner

Assignment 1 Checklist/Test Cases

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

ISSUE: Originally used an integer to hold the converted values but it's possible for them to overflow so changed to a long to allow the program to convert larger numbers and cover more possibilities.

[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?

ISSUE: Cleaned up import statements auto generated by IntelliJ with * notation instead of individual classes cluttering the import statements.

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**

[N/A] Are there any off-by-one errors in array indexing?

[N/A] Can array indexes ever go out-of-bounds?

[N/A] 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 "=="?

ISSUE: Originally used event.getSource() == button instead of .equals. Changed to .equals and re-tested.

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**
[N/A] In a switch statement is every case terminated by break or return?
[N/A] Do all switch statements have a default branch?
[N/A] 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**
[N/A] Are all files properly declared and opened?
[N/A] Are all files closed properly, even in the case of an error?
[N/A] Are EOF conditions detected and handled correctly?
[N/A] Are all file exceptions caught?

TEST CASES:

Test No.	**Input Binary**	**Expected Output**	**Input Hex**	**Expected Output**
1	1111	15	16	22
2	101001101111	1335	FFFFFFFF	4294967295
3	1221	Invalid format for a binary string - Illegal character: 2	AGI123	Invalid format for a Hexadecimal string - Illegal character: G
4	4100	Invalid format for a binary string - Illegal character: 4	AB124	700708