

Code Check List for Effective Personal Reviews and Formal Inspections  
W.L.Honig  
Version 1.0  
27Sept 08

A basic check list for code. Your team can extend if appropriate and necessary for your quality plan goals. Keep your checklists under configuration control and be sure all use the "latest" one.

Style Guide Defects (Style Defects may be minor or major)

1. Check everything in the style guide!

(many of these not only improve readability but also find or lessen chance of bugs if used properly!)

- Only minor defects regarding style. Nothing that interferes with code but could potentially improve readability. One thing would be making sure to add a comment to every method (besides main and overridden library methods).

Possible Defects

1. Are size and bounds checks carried out when and where necessary? (a good practice is to use inclusive lower limits and exclusive upper limits)
  - Yes, carried out for range and index checking when needed.
2. Is there ever any mismatch between the units of an expression and those of a variable? (feet to meters)
  - There is no mismatch. The units used remain constant throughout the code
3. Are number variables appropriate maximum size? (int, double, etc.)
  - Yes, the number variables for each data type is appropriate.
4. Are array, buffer, table (etc.) sizes appropriate? (Too little or too big, both bad)
  - Yes, the array sizes are appropriate. None are too large or too small.
5. Are equality tests used on floating point numbers? (Always bad)
  - No, equality tests are not used on floating point numbers since the tests wouldn't be precise.
6. When testing for errors, could the "error condition" actually be legitimate in some cases?
  - When testing the code, there is currently an error with parent IDs. Here, the parent ID's need to be added to parents array list instead of making a new one. The error condition here would be legitimate.
7. Are exceptions declared, thrown and handled correctly?
  - Yes, for the most part exceptions are declared, thrown and handled correctly. If the program runs into errors, still need to add alternative cases inside certain methods (if/else).
8. Is every function call the correct function? (Not a similarly named function)
  - Yes, every method in the code that is being called performs the correct function. There are no two same/similarly named methods. One issue we have with a method is the createParents method. Currently this method is creating duplicates of some children being added.
9. Are preconditions checked before a function call (if necessary)?
  - Yes, preconditions are checked and guaranteed to methods when necessary so that there are no errors.
10. Are postconditions ensured at the time of every possible function return point?
  - Yes, postconditions are ensured at the time of every possible function return point, for instance returning each array lists.
11. Are files and other resources closed in all possible exits?
  - Yes, all files and other resources are closed in all possible exits.

Note:

Studies have shown that the most acceptable forms of for loops (to avoid off-by-one errors) are these:

```
for ( i = 0; i <= max_index; ++i )  
for ( i = 0; i < sizeof(array); ++i )  
for ( i = max_index; i >= 0; --i )  
for ( i = max_index; i; --i )
```