**Simple Program Development Sheet Inspection Check List**

|  |  |  |
| --- | --- | --- |
| **Item** | **Section** | **Items** |
| Sheet header |  | * Descriptive program Name? * Accurate and informative program Description |
| Development Estimates and Actuals |  | * Does the actual size appear to be correct? * Is the estimated and actual effort reasonable in light of the comments? * Does the actual delivery date appear to be correct? |
| Requirements  Requirements may be given in the sheet or in the code. When given in the sheet they can take advantage of different fonts and automatic numbering | Constraints | * Concise? * Unambiguous? * Correct |
|  | Preconditions | * Concise? * Unambiguous? * Correct? |
|  | Invocation | * Concise? * Unambiguous? * Correct? |
|  | Input/Output | * Concise? * Unambiguous? * Correct? |
|  | Postconditions | * Concise? * Unambiguous? * Correct? |
|  | Use Cases | * Sufficient?   All high level initial conditions covered?  All user errors covered?   * Matches requirements (only insignificant additional detail introduced |
|  | general | * Verification (observation, inspection, test) provided for each requirement? |
|  | all | * Consistent? * Complete? |
| Logical Test Conditions |  | * Hierarchical list with an exhaustive partitioning at each level? * Are all pertinent conditions from a requirements perspective covered? * Matches requirements? |
| Test Cases |  | * At least one test case for every test condition? * Test cases cover variety of input variations? |
|  |  |  |
| Design  Designs can be given in the sheet, but since design elements must be matched 1-to-1 with the code they are usually given in the code |  | Calling tree:   * Are all methods/functions listed? * Is the calling tree accurate?   Function designs:   * Correct format/layout * Initial words from design standard * Matches structure of code * Sufficient detail to mentally execute and verify use cases? * Sufficient detail for coding |
| Code  The sheet should provide a calling tree of all the methods/functions |  | * Do all source files satisfy the coding standard? * Do the design and code elements match |
| Correctness Arguments |  | * Is each statement in each Cxx item correct?   + To verify this inspectors will need to examine the program source files. It is helpful if the inspection team has access to a laptop that can automatically search these files. * Cxx items one-to-one with Rxx items? * Is each Cxx item logically sound? * Does each Cxx item provide an argument that the completed program satisfies the requirement? |
| Random Test Conditions |  |  |
| Test {Files & Scripts | Scenarios}  Scenarios should be given where test input files cannot be used, for example GUIs |  | * Are all test cases covered? * Are all outputs correct? * If there is a test script, is it correct? * Are all the test output files listed? * Does the random test generator produce test files that meet the requirements? * Do the random tests give reasonable results? * Are any of the random test results incorrect? |
| Test Report |  |  |