

SCSU CSCI UDWR

Source Code Documentation Rubric

Trait	Outstanding	Good	Marginal	Poor
Code Readability	Outstanding coding style & organization. Code is very readable.	Good coding style & organization. Code is fairly readable.	Code is readable only by someone who knows its purpose and functionality.	Code is poorly organized and very difficult to read.
Documentation & embedded comments	Clearly & comprehensively explains the code's purpose & functionality.	Fairly explains the code's purpose & functionality	Marginally explains the code's purpose & functionality	Poorly explains the code's purpose & functionality

Guidelines:

The proficient use of literate programming (as mandated) should provide a student with considerable progress towards satisfaction of the Source Code Documentation requirement. In practice, this rubric should be applied in conjunction with the Literate Programming Rubric.

In all contexts, consistency of style is desired. The student should be free to express themselves with the documentation style of their choice, given that it is reasonable, so far as they apply it consistently.

Comments embedded in the code:

- Primarily describe intent.
 - **Modules** will express their *contract* including preconditions, postconditions, parameter details & exceptions thrown.
 - Where appropriate, invariants should be identified
 - **Classes** should include
 - A short descriptive statement of purpose
 - Author
 - Date of last change
- Secondly describe *what*, and possibly *how*, where the code itself is unavoidably obfuscating.
- Reflect the *current* state of the code, i.e. reflect *changes* to the code.
- *Global constants* and *magic numbers* should be explained and justified.

White space should be exploited:

- Within expressions
- Within blocks to express statement groupings
- To highlight structuring (indenting)
- To break long lines
- Accurately (not indicating false structure)

Identifiers should be:

- Descriptive
- Consistent in style (e.g. use underscore or capitalization for word concatenation, but not both)
- Appropriate (e.g. loop indices can be one of the singleton characters [i-n])
- Accurate (not misleading)
- Distinguishable (it should not be easy to misread one identifier as another)