Name	Assignment 1 rubric
Description	
Rubric Detail	

	Levels of Achievement		
Criteria	**Proficient**	**Competent**	**Novice**
READABILTY Program structure **Weight 10.00%**	**100.00 %** Logical modular blocks are used to make the code easy to understand. Constants help to improve code readabiity.	**50.00 %** A mixture of logical modular blocks is used along with some poor code structure.	**0.00 %** Poor code structure is used which makes the code difficult to read.
Identifier names **Weight 10.00%**	**100.00 %** All variable and function names are informative, and well chosen, increasing readability of the code.	**50.00 %** Most identifier names are informative and well chosen, improving code readability somewhat.	**0.00 %** Most identifier names are poorly chosen.
Repeated code **Weight 10.00%**	**100.00 %** Repeated code has been avoided.	**50.00 %** Some code is repeated unnecessarily.	**0.00 %** Large amounts of code are repeated unnecessarily.
Variable scope **Weight 10.00%**	**100.00 %** Only local variables have been used in functions. Global variables have not been used.	**50.00 %**	**0.00 %** Global variables have been used.
SIMPLICITY Algorithmic logic **Weight 10.00%**	**100.00 %** Code is brief but clear, and has no logical errors.	**50.00 %** Code is occasionally too complex or has minor logical errors.	**0.00 %** Code is overly complex or has major errors.

	Levels of Achievement		
Criteria	**Proficient**	**Competent**	**Novice**
Control structures **Weight 10.00%**	**100.00 %** Control structures are used to good effect.	**50.00 %** A small number of control structures are unnecessarily complex.	**0.00 %** Many control structures are poorly designed (eg. excessive nesting and/or looping)
DOCUMENTATION All modules, and functions have informative docstring comments **Weight 20.00%**	**100.00 %** All docstrings are accurate and descriptive, and clearly show how parameters and return types should be used.	**50.00 %** Almost all docstrings are accurate and reasonably clear descriptions of how module/function is to be used. Almost all parameters and return types are described clearly.	**0.00 %** Several docstrings are inaccurate or unclear, or absent. Some parameters and return types are unclear.
Comments are clear and concise, without excessive or extraneous text **Weight 10.00%**	**100.00 %** Almost all comments enhance the comprehensibility of the code. Comments almost never repeat information already apparent in the code.	**50.00 %**	**0.00 %** Many comments are unnecessary to the comprehension of the code. Alternatively, many comments are overly verbose.

	Levels of Achievement		
Criteria	Proficient	Competent	Novice
Significant blocks of program logic are clearly explained by comments **Weight 10.00%**	**100.00 %** Important or complex segments of logical flow (e.g. major loops or conditionals) are almost always clearly explained/ summarised. Almost no stating of the obvious (eg. a loop iterates over a sequence).	**50.00 %**	**0.00 %** Some important or complex blocks of logic are explained/summarised poorly. Alternatively, some unimportant blocks of code are given excessive coverage in the comments.

View Associated Items

Print Close Window