Project rubric for (authors' names): My name:

Criterion	Expert (1)	Skilled (2)	Apprentice (3)	Learner (4)	(5)
Presentation	***Information clearly, attractively presented.*** Fonts reflect importance of items. Graphics clarify topics. Structure helps audience to understand.	Information clear and attractive. Font or graphics confusing. Structure helps audience to understand.	Information presented clearly. Font uneven. Structure and graphics are confusing or unclear.	Structure, graphics and text unclear or missing. Fonts uneven.	
Use of language	***Correct verb tenses, grammar and spelling. Text is easy to read.***	Correct verb tenses, grammar and spelling. Text is difficult to read.	Correct verb tenses, punctuation and spelling. Some grammatical mistakes.	Major grammatical errors.	
Scientific language	***Authors' use of technical language shows they understand their subject's background knowledge.*** They minimise their use of jargon and define all necessary terms and acronyms.	Language shows authors understand the background knowledge of the subject. Jargon is appropriate. Not all terms and acronyms are defined.	Language shows that authors understand the background knowledge of the subject. Too much jargon and undefined acronyms.	Informal language: authors do not seem to understand the subject properly. Some scientific terms incorrectly.	
Abstract	***Clear, complete, understandable, creates interest.***	Clear and complete, but too long.	Clear, understandable, but incomplete.	Absent, confusing.	
Introduction	***Explains purpose and significance of research (question, adaptive theory and research hypothesis)*** using logical, well-referenced background argument.	Justifies adequately the significance of research question and hypothesis to the field of investigation.	Information justifying significance of research question and hypothesis is too covoluted or poorly referenced.	Argument only poorly justifies the research question and hypothesis.	
Adaptive theory	***Theory is dynamical, rheolectic and operationally verifiable*** based on background argument.	Theory is verifiable and believable but not on basis of argument.	Theory plausible, but not indicated by background argument.	Theory is implausible or absent.	
Research hypothesis	***Causally justified by theory***; ***expresses directed relationship*** from a single, measurable IV to DVs.	Hypothesis justified, IV operational, but DV not clearly defined.	Hypothesis is plausible but IV and DVs not clearly defined.	Hypothesis relationship is unclear.	
Model specification	***Specifies objects and their attributes, operations and relations***. Specifies program parameters and outputs (emergent or regular).	Objects and outputs clearly presented. Parameters unclear.	Object attributes/operations or outputs are not clear	Model not presented clearly.	
Null hypothesis	***Experiment tests a null hypothesis***, not just verifying a special case of research hypothesis.	Null hypothesis is present and used, but unclearly or invalidly formulated.	Null hypothesis is given, but not considered when discussing conclusions.	Null hypothesis is missing.	
Alignment	***Complete reference patterns align with literature.***	Reference relations or data missing.	Reference behaviours missing/unclear.	Patterns missing / unclear.	
Method design	***Processes explained clearly enough to replicate the investigation.*** Provides data that accurately investigate research/null hypothesis.	Methods specific and clear enough that I could replicate investigation. Data appropriate to null hypothesis.	Methods too unclear to be sure whether they are appropriate for investigating the research/null hypothesis.	Methods not provided, or inappropriate for investigating hypotheses.	
Results	***Measurements in several formats (tables, graphs) to show trends.*** Justifies analysis; describes errors.	Measurement data in several formats. Errors not described.	Measurement data not organised to aid understanding of knowledge gained.	Data unfitting/missing. Figure labels inadequate.	
Conclusions	***Discussion makes clear whether data support or reject the null hypothesis.*** Demonstrate authors' understanding of results, and express implications, applications and remaining uncertainties.	Conclusions explain whether data support or reject null hypothesis. Demonstrate authors' understanding and synthesis of the results.	Conclusions make clear whether results support or reject null hypothesis, but authors only partly understand their relevance for current paradigms.	Conclusions are inconsistent with the data.	
Citations, references	***Up-to-date reliable scientific sources, cited properly according to given guidelines.***	References properly cited, and most are reliable scientific sources.	References properly cited but some are non-scientific or out of date.	References not provided or improperly cited.	

Code structure	***Code is clearly organised and formatted*** using short, ***coherent code blocks*** and methods, clean and ***logical indentation***, and clear line-breaking (lines under 80 characters) to make it very easy to follow.	Code is easy to read with minor formatting/indentation mistakes, e.g.: bracket-matching.	Code is generally easy to follow, but logical formatting is poor.	Code is readable only by someone who knows what it is supposed to be doing.	
Clarity and coherence	***Has a clear, consistent conceptual metaphor.*** Program header comment clearly states this metaphor. Coding components (comments, variable and method names) *consistently* declare their role by reference to this metaphor.	Conceptual metaphor is present, but unclearly stated. Coding components mostly refer to this metaphor for clarity.	Conceptual metaphor is present, but unclearly stated. Relationship of coding components to this metaphor are generally unclear. Use of magic numbers.	Program has no clear conceptual metaphor.	
Comments	***Comments and accompanying documentation explain clearly what the code is doing*** by using clear, simple language and appropriately positioned and formatted comments.	Header and inline comments make the code easier to understand.	Inline comments are embedded in the code and separate logical code sections.	Inline comments are embedded in the code.	
Variable naming	***Variables' names express clearly and succinctly their purpose in program.***	Variable names are awkwardly long but express their purpose clearly.	Variable names express only unclearly their purpose in program.	Variable names express their purpose only very vaguely.	
Data types	***Variable types (array, logical, …) are used appropriately*** to produce correct results.	Variable types used appropriately to produce mostly correct results.	Variable types are used appropriately but produce incorrect results.	Variable types are used inappropriately/incorrectly.	
Control structures	***Control structures (selection, iteration, …) are used appropriately*** to produce correct results.	Control structures appropriately used to give mostly correct results.	Control structures used appropriately but produce incorrect results.	Control structures are used inappropriately/incorrectly.	
Modularity (classes)	***Module architecture is clear and easy to follow.*** Data and method responsibility cleanly packaged into functional modules to minimise rippling.	Modularity is clear and easy to follow but responsibility allocation permits some rippling.	Modularity is easy to follow but global data permit excessive rippling.	Data-, but not method-, responsibility is allocated modularly.	
Factorising	***Code is factorised cleanly to avoid redundancy.***	Some unnecessary data duplication.	Functionality duplicated unnecessarily.	Broad duplication of code.	
Execution	***Program runs correctly with no logic errors*** and displays appropriate output.	Program runs correctly with no logic errors, but inappropriate output.	Program runs correctly with no logic errors.	Program runs with logic errors.	
Validation	***Program fulfils all specifications, and performs exception-checking*** for errors and out-of-range data.	Program runs and meets all specifications. Performs some checking for entry and range errors.	Program produces correct results but displays them incorrectly. Some checking for entry and range errors.	Program gives correct results but displays them incorrectly. No error-checking.	
Efficiency	***The code is extremely efficient***, storing multiply used data and reducing processing steps without sacrificing readability or comprehensibility.	The code is efficient without sacrificing readability or comprehensibility.	The code is fairly efficient without sacrificing readability or comprehensibility.	Code is inefficiently patched together from mismatching partial solutions.	
Total =		**/ 25 =**			