Biocuration Assessment Scoring Rubric

Aspects and scoring criteria

Code Correctness	15 points	10 points	5 points	0 points
	Program achieves all tasks described in the assignment and the format and content of results completely match those expected.	Program achieves most of the tasks described in the assignment and format and content of most results match those expected.	Program achieves at least one of the tasks described in the assignment and the format and content of some of the results match those expected.	Program achieves none of the tasks described in the assignment and/or the results provided from the program do not match those expected.
Reusability		6 points	3 points	0 points
		The code is encapsulated in functions where appropriate and key parts of the program could be adapted/re-used for a similar task with minimum effort if required. Variables are only hard-coded where appropriate.	Some of the code is encapsulated in functions where appropriate and parts of the program could be adapted/re-used for a similar task with minimum effort if required. Some variables have been hard-coded unnecessarily.	Code has not been encapsulated in functions and would be difficult or impossible to adapt to a similar task without a lot of effort. The values of most or all variables have been hard-coded.
Documentation		6 points	3 points	0 points
		Names of variables, functions, etc make it easy to understand what most or all of the code is doing, and additional notes/description is provided by comments where appropriate. Functions include documentation that a user could accessed via the help() interface.	It is partially clear what the code does, from the names of variables, functions, etc. and some attempt has been made to annotate the code with comments.	It is difficult to understand what the code is doing and/or there is no/inappropriate use of comments.
Program Design	3 points	2 points	1 point	0 points
	Program runs efficiently, code is well organised, and there is no unnecessary duplication within the script. Output visualisation(s) is/are clear and well laid-out.	Some of the program is written efficiently and organised well. There is some unnecessary duplication within the script.	There are some cases where the code could be better organised and/or run more efficiently.	Code is poorly organised, inefficient, may use the wrong approach, and output visualisation(s) is/are unclear and pooly laid-out.