

Ben - Ques. 2 - Assign. 3

A302	Level 4	Level 3	Level 2	Level 1
communication	colouring to add to understanding, input one colour/output another	Flowchart - organized, proper spacing, correct symbols, proper language Comments - useful succinct (block OK), not sparse, block at top (modifications, start date, programmer name, problem description, Notes on anything special or unusual, citations) Readability - code is easy to follow, enough white space	Poor spacing, some incorrect symbols, typing errors, lacking explanation, inline comments, or hard to read, or spelling mistakes on output or Programmer missing or modifications	poor organizations, many symbols used incorrectly for the concept attempting to convey, arrows in the wrong section or incorrect pointer. Any two or more details missing or Citations/borrowed work not acknowledged OR more than 25% of code borrowed
	Excellent User instructions that are clear and can be accessed at any time or Additional options, like premature quit.	user interface, clear language, no grammar or spelling errors 4 - Open a file in the file manager, placing in an appropriate location, proper terminology	Adequate 2 or 3 grammar or spelling errors I/O opens and closes properly but may be inefficient OR may be skipping certain lines/words	Needs improvement User interface not clear, poor language and or spelling and grammar errors hard coded words and forgot to close files missing information in columns or steps missing
Knowledge Thinking & inquiry	With open or similar more advanced coding Description at top to clarify	Graphics are looking for the components Trace statements changes (Tables) All functions (in work as expected)	Unclear steps or incorrect terminology 24 Small graphics, features not intuitive Variable names don't match or missed a step	No graphics, just console that user would not necessarily have. Lost count on number of guesses more confusing, complicate the existing error, or do not recognize it.
Thinking & inquiry	Screen sizes to user systems. Make it interactive. Colour or design helps user know what stage they are at.	Code or class	Functions work, but confusing or inefficient Nested if's or lack of proper loops - does not take advantage of built in functions	Hard coded where a user defined function or built in would suffice
Application	Could be class based Highly efficient, best possible, perhaps found a new function or way to complete the task			No functions or classes

Comments - Your beginning point was hangingman 2 - your task was to use + improve the code - you could have used pygame or turtle Everything works - code is Good.