

A3Q4	Level 4	Level 3	Level 2	Level 1
communication	Nothing assumed, catching errors, anticipating users needs - beyond Level 3 Excellent	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 user interface, clear language, no grammar or spelling errors	lacking explanation, inline comments or hard to read, or spelling mistakes on output or Programmer missing or modifications Adequate 2 or 3 grammar or spelling errors	Any two or more details missing or Citations/borrowed work not acknowledged OR more than 25% of code borrowed Needs improvement User interface not clear, poor language and or spelling and grammar errors
Knowledge Thinking & inquiry	With open or similar more advanced coding - not required or not taken in class OR new or better implementation of class work Description at top to clarify	Proper class usages and storing of data IPO - proper chart (headings), steps, proper terminology	Considerable usages and storing of data, few minor errors Unclear steps or incorrect terminology	hard coded words and forgot to close files missing information in columns or steps missing
Thinking & inquiry	Screen sizes to user systems. Make it interactive. Colour or design helps user know what stage they are at	Graphics are appropriate and professional looking for 21" word list appropriate components - Standard screen size 1280 x 1024	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
Application	Could be class based Highly efficient, best possible, perhaps found a new function or way to complete the task	All functions (including user) are efficient and work as expected Code efficiency - nested if's better into function 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
Flowchart/IPO/Trace Table (Application)	colouring to add to understanding, input one colour/output another - No	Flowchart - organized, proper spacing, correct symbols, proper language	Poor spacing, some incorrect symbols, typing errors.	No functions or classes poor organizations, many symbols used incorrectly for the concept attempting to convey, arrows in the wrong section or incorrect pointer.

calls own class - (70 orig. in your class) = 30%

Need conside to run - instructions on screen - Active
No overrided or overloads.
Card off my screen