

Est. Thomas D.ies 3 Assign 3 Not MS Die, - Sign / #

A3Q3	Level 4	Level 3	Level 2	Level 1
communication		Comments - useful succinct, (block OK), not sparse, block at top (modifications, start date, programmer name, problem description, Notes on anything special or unusual, citations)	lacking explanation, inline comments, or hard to read, or spelling mistakes on output or Programmer missing or modifications	Any two or more details missing or Citations/borrowed work not acknowledged OR more than 25% of code borrowed
	Excellent	Readability - code is easy to follow, enough white space	Adequate	Needs improvement
	User instructions that are clear and can be accessed at any time or Additional options, like premature quit overloading or similar more advanced coding	user interface, clear language, no grammar or spelling errors	2 or 3 grammar or spelling errors used but not to best advantage or created cumbersome usage	User interface not clear, poor language and or spelling and grammar errors
Knowledge		Effectively used classes and inheritance		hard coded words and or excessive repetition
	colouring to add to understanding, input one colour/output another	Flowchart - organized, proper spacing, correct symbols, proper language		Poor organization, many symbols used incorrectly for the concept attempting to convey, arrows in the wrong section or incorrect pointer.
Knowledge		IPO - proper chart (headings), steps, proper terminology	Poor spacing, some incorrect symbols, typing errors,	missing information in columns or steps missing
Thinking & Inquiry	Description at top to clarify	Graphics are appropriate and professional looking for Latree hangman - word list appropriate components - Standard screen size 1280 x 1024	Unclear steps or incorrect terminology	
Thinking & Inquiry	Screen sizes to user systems. Make it interactive. Colour or design helps user know what stage they are at.	Trace statements - variable name, steps and changes (Table/chart)	Small graphics, features not intuitive	No graphics, just console that user would not necessarily have. Lost count on number of guesses.
		All functions (including user) are efficient and work as expected	Variable names don't match or missed a step	more confusing, complicate the existing error, or do not recognize it.
Application	Could be class based	Code efficiency - nested if's better into function or class.	Functions work, but confusing or inefficient	Hard coded where a user defined function or built in would suffice
	Highly efficient, best possible. perhaps found a new function or way to complete the task	Flowchart - organized, proper spacing, correct symbols, proper language	Nested if's or lack of proper loops - does not take advantage of built in functions	No functions or classes
Flowchart/IPO/Trace Table (Application)	colouring to add to understanding, input one colour/output another		Poor spacing, some incorrect symbols, typing errors,	poor organizations, many symbols used incorrectly for the concept attempting to convey, arrows in the wrong section or incorrect pointer.

90 MS Die Inherit None Proper Calls
 Flow Colour - N Trace ✓ Version X
 Block @ Top Needs Improv.

One Round / Game
 Date ✓ IPO ✓

PyGame