するのころののコ ひまり, の とれたろ

A302	Level 4	Level 3	Level 2	Level 1
communication	colouring to act to understanding, input one colourbulput another	Rowchart - organized, proper spacing, correct symbols, proper language	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.
		Comments - useful sincinct, (block OK), not sparse. b' at top (modifice hors, start date, programmer name, problem cascription, Notes on anything special or unusua, citations). Meadabuity - code is easy to follow enough	lacking explanation, inline coments, or hard to read, or spelling mistakes on output or Progammer missing or modifications	Any two or more details missing or Citations/borrowed work not acknowledged OR more than 25% of code borrowed
	User instructions that are clear and can be	white space	Adequate	Needs improvement
	accessed at any time or Aditional options, like premature quil.	user interface, clear language, no grammar or spelling errors. 10 either CSV or bt - Open and closure, proper	2 or 3 grammar or spelling errors	User interface not clear, poor language and or spelling and grammar errors
Knowledge	With open or smilar more advanced coding		VO opens and closes properly but may be inefficient OR may be skipping certain lines/words	hard coded words and forgot to close files.
Thinking &	Description at top to clarify	IPO - proper chart (headings), steps, proper terminology	Unclear steps or incorrect terminology	missing information in columns or steps missing
Thinking & Inquity	Screen sizes to user systems. Make it interactive. Colour or design helps user frow what stage they are at.		Small graphics, features not intuitive.	No graphics, just console that user would not necessarily have Lost count on number of guesses.
		Trace statements - variable name, steps and changes (Table/chart)	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	All functions (including user) are efficient and work as expected	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. he task	Code efficiency - nested if's better into function or class.	Nested ifs or lack of proper loops - does not take advantage of built in functions	No functions or classes

and placement. I/O should around the #0 letters block. Well done!