Critical Thinking Rubric – Programming

Note: module = function = procedure = method

	Advanced – 4	**Competent – 3**	**Developing – 2**	**Elementary – 1**	**Score ___**
Identify	Identifies the input source(s) and the valid range of values for each input required. Identifies all required output and the output range(s).	Identifies the input source(s) and all required inputs. Identifies all required outputs. Misrepresents the valid range of values expected for one or two input/output values.	Identifies most, but not all of the required input and output. Does not correctly identify the valid range of values expected for each input and output.	Does not identify any of the required inputs or outputs.	
Gather	Poses insightful and pertinent questions to obtain the information needed to create a solution that meets all problem specifications and anticipates all special cases.	Poses the necessary questions in order to create a working solution for all but one or two special cases.	Poses a few important questions, but does not obtain the information needed to create a solution that works in every case.	Does not ask the necessary questions and does not obtain the information needed to create a working solution.	
Examine	Develops multiple solution alternatives and selects an approach that will efficiently and effectively solve the given problem.	Develops two or three solution alternatives. Selects a method that is most comfortable for the student, but is not best tailored to solve the problem.	Develops a single approach to solving the problem based on initial impressions obtained from example input and output.	Does not examine the problem to develop solution alternatives.	

Critical Thinking Rubric – Programming

Note: module = function = procedure = method

Formulate	Major processes required to solve the problem are determined and assigned to modules. Plan addresses all requirements and special cases. Modules can easily be modified or replaced if changes are made to the current problem.	Major processes required to solve the problem are determined and assigned to modules. Plan does not address one or two minor requirements or special cases. Significant redesign may be required if minor changes are made to the problem statement.	Plan does not address one of the major problem requirements; or the plan does not address three or more minor requirements or special cases. Plan will not solve the problem for all valid input.	Does not formulate a plan to solve the problem.	
Apply	The solution contains no logic errors and produces the correct output for all valid inputs. The solution is user friendly. Error checking is included to take care of any invalid input.	The solution contains few, if any, syntax or logic errors. There are problems with one or two special cases which were not anticipated. This will cause problems or confusion to the end user when the special cases are encountered.	The solution will work correctly for stated examples. Incorrect output occurs for more than two sets of input not specifically shown in the problem examples.	Solution will not execute correctly for the sample problems.	
Evaluate	Solution is well-documented relating how it solves all cases of the current problem. Notes are included pertaining to anticipated changes which may be made to the solution.	The solution is documented and shows an understanding of how it solves most valid instances of the problem. It is missing notes about one or two special cases. It is missing notes to aid in making future modifications.	Little documentation is provided. Documentation does not show an understanding of how the solution solves all valid instances of the problem.	No documentation is included.	