**Competency Statement:** Create meaningful and efficient programs including choosing which information to use and how to process and store it, breaking apart large problems into smaller ones, recombining existing solutions, and analyzing different solutions.

|  |  |  |
| --- | --- | --- |
| 4 Advanced | 3 Proficient | 2 or 1 Areas of Concern |
| Going above & beyond.   * The ‘WOW’ factor: Demonstrating ability beyond what’s explicitly taught or expected. | Meeting Expectations, Goals, or Objectives   * Demonstrates competency to learning goals by accurately completing all aspects of assignments. * Makes connections to real-world applications. * Able to articulate concept meaning for a product. | Not meeting expectations   * Lack of evidence or the evidence does not yet demonstrate a minimum level of ability. |
|  | **Research, Planning, Flowcharting, Pseudocode**  I can…   * Use a variety of sources to conduct research to obtain pertinent information about a task or topic. * Apply strategies and use tools for project planning (i.e. time, materials, resources). * Describe concepts and demonstrate fundamental skills used for creating visual representations of a project for planning or for presentation. * Create a logical pseudocode outline for a program that includes the required details prior to coding it. |  |
|  | **Decomposition (top/down design), commenting**  I can…   * Use a variety of strategies to break a large task into its smaller parts, identify their order, and where the connections between them should be. * Demonstrate the ability to apply the decomposition concept to program code. * Consistently use code comments to identify and communicate information to others. * Recognize how the decomposition and commenting strategies can apply across multiple disciplines. |  |
|  | **Sequence, Selection, Iteration, Logic, Debugging**  I can…   * Identify & describe data types and language specific symbology and apply the correct syntax to code language so the program runs as intended. * Articulate the concepts of branching and loops and apply these control structures to define selection and iteration in code to control the program flow. * Identify comparative (relational), logic, mathematical and boolean expressions in conditionals, and accurately apply them within program code. |  |
|  | **Refactoring & Documentation (version control)**  I can…   * Describe and apply the Agile SCRUM methodology for project management. * Describe the concept of refactoring and apply it using functions, parameters, libraries, etc., to produce flexible and efficient code. * Document the progress during software development projects for communication and version control purposes. |  |