Competency Reflection

Pick five competencies (or subcompetencies) and reflect on how they are important to the projects you worked on this semester and what skill level you have attained with those different competencies. You will pick a new set of five each semester, so plan ahead. Use the rubric explanation in the header to help you know what the levels of competence are.

You should attach this document to your reflection(s) or you can use this document as a starting point for your reflections. Reflect on each of your five competencies in separate paragraphs.

[The blank competencies at the bottom may be used to insert your own skill(s) or competency(ies) that are important to you and the development of your project.]

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Understands Terms and Procedures:**  **11102** Classify the different steps and the files created when making a new executable file.  **11103** Identify and use the different data types in this programming language to manipulate data.  **11106** Plan and design a program using structured development techniques such as algorithms and pseudocode. |  |  |  |  |  |
| **Understands and Uses Math and Logical Operators:**  **11104** Use mathematical operators, comparison operators, and logical operators to solve business problems.  **11105** Recall the correct level of precedence for each of those operators. |  |  |  |  |  |
| **Understands and Uses Decision Making Structures:**  **11107** Use if and if/else selection structures to decide which actions to perform in a program.  **11109** Use the switch statement when making multiple decision selections. |  |  |  |  |  |
| **Understands and Uses Repetition Structures:**  **11108** Use repetition structures (while, do-while, etc.) to repeat a group of statements. |  |  |  |  |  |
| **Understands and Uses Methods:**  **11110** Explain why methods are a necessary component of modularizing Java programs.  **11111** Plan, design, create, and use methods in breaking down tasks to solve a problem.  **11112** Compare pass by value and pass by reference between methods. |  |  |  |  |  |
| **Understands and Uses Arrays of Different Types:**  **11113** Write valid programming statements to declare and initialize arrays, to refer to individual elements of an array, and pass arrays to functions.  **11114** Use and manipulate strings of characters, including the Standard Library string class. |  |  |  |  |  |
| **Understands and Uses Complex Data Types:**  Specify, define, implement, and use structures, enumerated and other user defined data types.  **11117** Specify, define, implement, and use simple classes. |  |  |  |  |  |
| **Design:**  **B1** Develop detail design specifications. |  |  |  |  |  |
| **Development:**  **C1** Create and modify new or existing system interfaces.  **C2** Create and modify new or existing code. |  |  |  |  |  |
| **Project Management:**  **F1** Define scope of project.  **F4** Estimate time requirements.  **F7** Evaluate project requirements. |  |  |  |  |  |
| **Debugging:**  **11116** Use the debugging tools available with the current compilers. |  |  |  |  |  |
| **Understands and Uses Complex and Dynamic Data Structures:**  Create and use dynamic data structures including but not limited to linked lists, stacks, queues, sets, maps, and trees.  **11101** Create and use pointers and dynamic memory allocation. |  |  |  |  |  |
| **Understands and Uses Recursion:**  Create and use recursion to solve problems. |  |  |  |  |  |
| **Problem Solving:**  **11115** Define the general scope of work to meet project requirements or solve a problem.  **H1** Define the problem.  **H3** Identify/test possible solutions.  **H5** Implement solution. |  |  |  |  |  |
| **Analysis:**  **A1** Be able to gather data to identify customer requirements.  **A2** Interpret and evaluate requirements.  **A3** Define scope of the work to meet customer requirements.  **A4** Develop high level systems and functional specifications. |  |  |  |  |  |
| **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_:** |  |  |  |  |  |
| **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_:** |  |  |  |  |  |
| **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_:** |  |  |  |  |  |