|  |
| --- |
| Software Development Revision – Use the table provided to create notes  |
|  |  |
| Software development process |
| Explanation of the iterative nature of the software development process |  |
| Description of the purposes of the software specification, and its status as a legal contract |  |
| Explanation of the importance of each stage of the development process |  |
| * analysis
 |  |
| * design
 |  |
| * implementation
 |  |
| * testing
 |  |
| * documentation
 |  |
| * evaluation
 |  |
| * maintenance
 |  |
| Identification of the personnel at each stage and brief description of their roles |  |
| * client
 |  |
| * systems analyst
 |  |
| * project manager
 |  |
| * programmer
 |  |
| * independent test group
 |  |
| Description and exemplification of pseudocode and one graphical design notation (structure diagram or other suitable) including data flow |  |
| Description and exemplification of top-down design and stepwise refinement |  |
| Explanation of the need for systematic and comprehensive testing |  |
| Explanation of the need for documentation at each stage |  |
| Evaluation of software in terms of |  |
| * robustness
 |  |
| * reliability
 |  |
| * portability
 |  |
| * efficiency
 |  |
| * maintainability
 |  |
| Description and exemplification of corrective, adaptive and perfective maintenance |  |
|  |  |
| Software development languages and environments |
| Description and comparison of |  |
| * procedural
 |  |
| * declarative
 |  |
| * event-driven languages
 |  |
| Comparison of the functions, uses and efficiency of |  |
| * compilers
 |  |
| * interpreters
 |  |
| Description of the features and uses of scripting language (including creating and editing a macro) |  |
| Explanation of the need for and benefits of scripting languages |  |
| Description of the use of module libraries |  |
|  |  |
| High level programming language constructs |
| Description and exemplification of the following constructs in pseudocode and an appropriate high level language: |  |
| * string operations (concatenation and substrings)
 |  |
| * formatting of I/O
 |  |
| * CASE (or equivalent multiple outcome selection)
 |  |
| Description and exemplification of |  |
| * real
 |  |
| * integer
 |  |
| * boolean variables
 |  |
| * 1-D arrays
 |  |
| Description and exemplification of |  |
| * procedures/subroutines/subprograms
 |  |
| * user-defined functions
 |  |
| * modularity
 |  |
| * parameter passing (in, out, in/out)
 |  |
| * call by reference/value
 |  |
| * local and global variables
 |  |
| * scope
 |  |
|  |  |
| Standard algorithms |
| Description and exemplification of the following standard algorithms in pseudocode and an appropriate high level language: |  |
| * linear search
 |  |
| * counting occurrences
 |  |
| * finding min
 |  |
| * finding max
 |  |