Student _____

Department of Computer Science and Engineering
College of Engineering, University of Nevada, Reno

CS 425 Software Engineering

Final Exam
December 17, 2007

Test type: Closed-book examination
Number of questions: 17   **Total points**: 46   **Weight**: 24% of the course grade   **Time:** 105 minutes

Notes: • For questions **1 to 10** indicate the correct answer (only one) on the answer sheets provided by the instructor. Questions 1 to 10 have each a 1 point value for a group total of 10 points.

• Questions **11 to 17** require that you elaborate your answers. You must also write these answers on the sheets provided by the instructor. The total value of questions 11-17 is 36 points.

Questions:

1 In software engineering, a *software process* is:
a. A set of executable modules that run concurrently
b. An activity that is repeatedly executed during software development
c. A sequence of steps that converts inputs to one or more outputs
d. An organized set of activities performed for developing a software system   [1 point]

2 Risk assessment is most emphasized in which of the following *software process models*?
a. Component-based software engineering
b. Spiral model
c. Incremental development
d. Waterfall model   [1 point]

3 Three sections that (together) most *project plans* for software development should include are:
a. Project schedule, architectural design, work breakdown
b. Risk analysis, hardware and software resource requirements, project schedule
c. Project organization, monitoring and reporting mechanisms, data validation
d. Each of the above contains three valid project plan sections   [1 point]

4 Which of the following is a type of *non-functional requirements*?
a. Portability requirements
b. Delivery requirements
c. Privacy requirements
d. All of the above   [1 point]

5 Which of the following is <u>not</u> a generic architectural model for RTS *(real-time systems)?*
a. Version control system
b. Data acquisition system
c. Monitoring system
d. Control system   [1 point]

6 In architectural design, which of the following is a style of *modular decomposition*?
a. Interrupt-based decomposition
b. Use-case and scenario decomposition
c. Function-oriented pipelining
d. None of the above   [1 point]

7 Which of the following is <u>not</u> a technique for *user-interface evaluation*?
    a. Analyzing video recordings of typical system use
    b. Instrumenting code to collect usage statistics
    c. Performing hierarchical task analysis
    d. Surveying users by questionnaires   [1 point]

8 *Validation testing* is intended to show that:
    a. The software is extensible and well documented
    b. The software meets its requirements
    c. The software contains defects
    d. All of the above   [1 point]

9 The *cyclomatic complexity* of a program is equal to:
    a. The minimum number of test cases needed in performance testing
    b. The minimum number of test cases needed in integration testing
    c. The minimum number of test cases needed in path testing
    d. The minimum number of test cases needed in partition testing   [1 point]

10 *Self-oriented people* are primarily motivated by:
    a. Interacting with co-workers
    b. Technical challenges
    c. The work they do
    d. Personal success and recognition   [1 point]

11 Describe the *evolutionary development software engineering* process model. Also, indicate its advantages, disadvantages, and applicability.   [6 points]

12 Give three examples of *risks* that may be identified by software project managers (give concrete examples, do not simply enumerate risk categories such as technology, organizational, etc.) and suggest *risk management strategies* for each of the three risks.   [3 points]

13 Describe the *broadcast model* used for control modeling in architectural design. Also, indicate its advantages, disadvantages, and applicability.   [5 points]

14 Describe and compare the following two styles of user interaction: *direct manipulation* and *command language*. Also, for each interaction style give an example of software application that you are familiar with – briefly explain how this software application supports the specific interaction style.   [5 points]

15 Concisely describe each of the following approaches for *test case design*: requirements-based testing, partition testing, and structural testing (3 to 5 lines each).   [4 points]

16 Give an example of a *data flow diagram (data flow model)* that contains at least 7 data transformations (processing steps). Describe the meaning of data flow diagram and use the appropriate DFD notation.   [5 points]

17 Consider your *group project* in CS425/625.
    a. Briefly describe the project's topic, utility, and most important features (6 to 10 lines).
    b. List the project's three most important functional requirements.
    c. Briefly summarize the project's implementation solution: programming language(s) used, operating platform(s), main subsystems/modules of the code (expectedly, between 4 and 6), challenges encountered, and current status of the project.
    d. Indicate three possible enhancements for your project.   [8 points]