CS3733-D04 Midterm Exam

Name:

Answer all of the questions as completely as possible. Do your own work. There is an academic honesty policy at WPI and it will be enforced. Good luck.

Part 1: Basic Knowledge

For multiple choice questions, circle all that apply.

1. Which of the following are *primary* reasons for having a process for software development:
    a. To make sure that you produce the right documentation
    b. To ensure effective communication
    c. To ensure that everyone understands their role
    d. To accurately predict the schedule and cost
    e. To ensure that everyone uses the right tools

2. Which of the following are major products of software engineering:
    a. Documentation
    b. Designs
    c. Software products
    d. Schedules
    e. Requirements

3. What are the four P's of software engineering according to Braude?

    People, Process, Product, Project

4. Why is software engineering different than other types of engineering, like mechanical or electrical engineering? Give an example.

    Incomplete information, ever-changing requirements. Example: You don't decide to move a bridge, half way through construction.

5. Of the four P's of software engineering, which is the most important in determining the success of a software development effort? Why?

    People. People can sabotage process, and if they do not work together effectively, the project will fail. Also, you cannot create software without people's intellectual effort.

6. Which of the following are characteristics of a project's Vision?
    a. Defines the features of a product

b. Identifies the stakeholders' view of what the product should be
c. Traces needs to requirements
d. Identifies the development environment
e. Provides projected revenues

7. Which are reasons for adopting an iterative, incremental software development lifecycle over a waterfall lifecycle?
   a. Waterfall is not used much in practice anymore
   b. Waterfall does not lower risk as quickly
   c. Waterfall projects are almost always delivered late
   d. Iterative, incremental adopts to changes better
   e. Iterative, incremental ensures customer participation

8. Describe in a couple of sentences what an iteration is.

   An iteration is a, usually time-boxed, period in which a particular part of a software product is worked on. At the end of an iteration, there is demonstrable software that is used to indicate the progress.0

9. Which of the following are practices of Extreme Programming (XP)?
   a. Daily stand-up meetings
   b. Simple design
   c. Pair programming
   d. Planning game
   e. Minimal documentation

10. Fill in the blanks:

    A use case is a **complete** set of actions, performed by the system, initiated by

    a(n) **actor**, that delivers **visible results**.

11. Differences between use cases and user stories are:
    a. Customer writes use cases, development team writes user stories
    b. Customer writes user stories, development team writes use cases
    c. Use cases are testable
    d. User stories are incomplete specifications
    e. User stories can be used for planning, use cases cannot

12. Which of the following are characteristics of a project plan in an iterative, incremental project?
    a. It lays out all of the activities for the whole project at the beginning of the project
    b. It contains summaries of the individual iteration plans
    c. It is constantly changing
    d. It is used for communication

e. It identifies who does what

13. What is the difference between black-box and white-box (clear-box) testing?

Black-box testing does not depend on knowledge of the code and is based upon the requirements. White-box testing requires knowledge of the code implementation.

14. The three types of testing are:

System, **Integration**, **Unit**

15. A good requirement must be:
    a. Testable
    b. Unambiguous
    c. Short
    d. Functional
    e. Formal

Part II: Application

1. Write a use case that describes the interaction between the Caller and the wireless phone system for making a phone call. This does not have to be formal, but should include the basic flow of events, and major alternate flows of events.

   Brief description
   This use cases describes how a Caller places a call using their mobile (cell) phone.

   Preconditions
   The phone is turned on. There is power in the battery. The keyboard is unlocked.

   Basic flow:
   The Caller enters the phone number to call. This may be done either by entering the number through the keypad of the phone, or using other number identifications such as voice identification, or speed-dialing. (Note: these could be alternate flows)

   The Caller presses the Send button. (or something like this)

   The phone connects to the wireless system and dials the requested number.

   The dialed number is answered.

   The Caller communicates with the dialed party.

   The Caller ends the communication. The use case ends.

   Alternate flows:
   There should be alternate flows for busy number, no signal, dropped signal, no answer.

The following code is based upon code from the AspectJ project.

```
public String toString() {
    StringBuffer sb = new StringBuffer();
    if (null != context) {
        sb.append(context);
        sb.append(LangUtil.EOL);
    }
    if (sourceFile != null) {
        sb.append(sourceFile.getPath());
    }
    if (startLine > 0) {
        sb.append(":");
        sb.append(startLine); //"" + startLine + "-" + endLine);

    }
    if (!noColumn) {
        sb.append(":" + column);
    }
    return sb.toString();
}
```

The fields, context is a String, sourceFile is a File, startLine and column are ints, and noColumn is a boolean. Use the table on the back of this page to ensure condition coverage for tests. That is, tell what values you would set the fields to, and then use the last column of the table to indicate what the output would be.

Each condition must be tested for true and false. This can be done with only two test cases. The first one would have all values that make the conditions true and the second would make all of them false. More test cases can be used, but all conditions must be at least true for one test case and false for at least one.