

ICS52 - Introduction To Software Engineering
Final Exam – Spring, 2005

First Name: _____ Last Name: _____

Student ID: _____

1. (24 points, 4 points each) Define the following terms, as used in software engineering:
 - A. Function point.
 - B. Unit test.
 - C. Regression testing.
 - D. Cohesion (of a component or class)
 - E. Equivalence partitioning.
 - F. Testing oracle.

2. (20 points) You have been assigned to design test cases for black box testing of the one parameter `equalsIgnoreCase` method in `java.lang.String`. From the documentation:

```
public boolean equalsIgnoreCase(String anotherString)
```

Compares this `String` to another `String`, ignoring case considerations. Two strings are considered equal ignoring case if they are of the same length, and corresponding characters in the two strings are equal ignoring case.

- (a) What is the input domain of the `equalsIgnoreCase` method?
- (b) What is a basis for dividing the input domain you described into subdomains?
- (c) Using the basis defined in (b), name three or four subdomains.
- (d) For each subdomain from (c), give a test case input and the expected output.

3. (12 points) How do we assess the quality of a software design? Name and briefly describe two technical criteria that can be used to evaluate the quality of a design.

Criterion 1

Criterion 2

4. (12 points) Here's a quotation from the textbook:

There is only one rule in designing test cases: cover all features, but do not make too many test cases.

- a. Assuming this piece of wisdom is generally true, what does it mean? Why is this "one rule" so crucial?

- b. Does the rule apply when doing white box testing? Explain why (and how) or why not.

- c. Now argue, based on sound software engineering principles and practices, that this "one rule" makes little or no sense.

5. (12 points) According to the Scientific American article "Command and Control," ATAMS was successful because it "combined several techniques that were shown years ago to produce better software faster." Name and briefly describe two techniques mentioned in the article, and explain which software engineering qualities or principles each technique illustrates

a. First technique:

b. Second technique:

6. (3 points) What does the textbook say about "functional independence" in design? (Choose one.)

- A. It is promoted by successively refining levels of procedural detail.
- B. It is a hallmark of agile development.
- C. It is achieved by developing modules with an aversion to excessive interaction with other modules.
- D. It is particularly hard to achieve when using the waterfall process model.
- E. It is a measure of the number of methods in a class's interface.

7. (3 points) "The components of a system encapsulate data and the operations that must be applied to manipulate the data." This is a short description of what architectural style? (Choose one.)
- A. Object-oriented architecture.
 - B. Call and return architecture.
 - C. Data-flow architecture.
 - D. Data-centered architecture.
 - E. Layered architecture.
8. (3 points) Which of the following is **not** a category of component coupling described in the textbook (choose one)
- A. Content coupling: one component surreptitiously modifies another component's internal data.
 - B. Routine call coupling: one operation invokes another.
 - C. Data coupling: operations pass long strings of data arguments.
 - D. Proximity coupling: one component resides in the same directory as another component.
 - E. Type use coupling: one component uses a data type defined in another component.
9. (3 points) An integration testing approach that allows the software team to assess its project on a frequent basis, by running daily tests on the entire system that are designed to expose "show stopper" errors, is called (choose one)
- A. Bottom-up testing.
 - B. Boundary testing.
 - C. Smoke testing.
 - D. Pipe and filter testing.
 - E. Validation testing.
10. (3 points) White-box testing and black-box testing are identical except for (choose one)
- A. who determines if the output is correct.
 - B. how the test cases are selected.
 - C. whether the testing can distinguish between failures, faults, and errors.
 - D. the use of drivers and stubs.
 - E. all of the above.
11. (5 points) Software engineering was defined as multi-_____ and multi-_____. (Fill in the blanks.)