CS 495 – Software engineering – review

1) A customer is describing an aspect of the problem domain ambiguously. There are two ways to interpret what was said, and you think you know which one was meant, but you aren't sure. How could you use active listening to resolve this?

2) Aside from active listening and asking questions, what else can you do to better understand the problem the customer is trying to solve?

3) Sometimes projects move forward without a clear idea of what problem is being solved. What are the perils of this?

4) When leading a brainstorming session, how can you encourage people to contribute ideas without fear of criticism?

5) You are preparing to develop a software package to assist dentists with taking medical notes and referring to previous notes when examining a patient. What are some possible problem domain entities and attributes in this scenario?

6) You are developing an application to help students practice math problems. It is a multi-user system with different types of problems, and it tracks how well each user has mastered each problem type. Write a relation-entity diagram illustrating this situation. Invent a few attributes that might be useful in this scenario.

7) You are in the midst of writing a requirements document for the math application described above. Document a query that retrieves the mastery levels of each problem type for a given user.

8) Think of a software system as being like an object. In this sense, the requirements document would talk about things outside of the object but still relevant to it. What would the system specification be analagous to?

9) Give 3 principles of user interface design and explain why they are important.

10) When would the waterfall lifecycle model be appropriate? When would it not be well suited?

11) Discuss the difference between iterative and incremental lifecycle models.

12) What factors should be considered when deciding on programming languages and other technologies in a project?

13) Explain how a proof-of-concept demo can reduce project risks.

14) Briefly describe the properties of an ACID database management system.

15) Give both an advantage and a disadvantage of a NoSQL database, compared to SQL.

16) On a human level, what are the most common reasons for project failure?

17) On a technical level, what are the most common reasons for project failure?

18) What concepts can be applied to manage software complexity?

19) Why is repeat data generally ill-advised?

20) Provide one of the advantages of using immutable objects.

21) What kind of naming conventions should functions and methods generally adhere to?

22) What does software engineering research generally indicate about the size of procedures?

23) What improvement can be made to the way parameters are being passed in the code below?

```
class Rectangle {
     ...
     public Rectangle(int height, int width) {
          ...
     }

     public void setSize(int width, int height) {
          ...
     }
}
```

24) What is the risk of not documenting units for parameters (when applicable)?

25) Name a few different techniques for error handling.