Pledge: *I pledge my honor that I have abided by the Stevens Honor System.* - **Eric Altenburg**

1.6: As software becomes more pervasive, risks to the public (due to faulty programs) become an increasingly significant concern. Develop a doomsday by realistic scenario in which the failure of a computer program could do great harm, either economic or human.

2.8: Is it possible to combine process models? If so, provide an example.

Yes, it is possible to combine process models, in fact, some software development departments/companies do not use a traditional process model. Instead, they end up using a proprietary process model which is just a combination of other more traditional process models that better fit their work flow/products.

A few examples of combined process models include:

1. Evolutionary process model

    -

2.9: What are the advantages and disadvantages to developing software in which quality is "good enough"? That is, what happens when we emphasize development speed over product quality.

3.2: Describe agility (for software projects) in your own words.

5.1: Based on your personal observations of people who are excellent software developers, name three personality traits that appear to be common among them.

6.6: Of the eight core principles that guide process (discussed in Section 6.1.1), what do you believe is more important?

7.1: Why is it that many software developers don't pay enough attention to requirements engineering? Are there ever circumstances where you can skip it?

7.5a: Develop a complete use case for making a withdrawal from an ATM.

8.1: Is it possible to begin coding immediately after a requirements model has been created? Explain your answer, and then argue the counterpoint.

8.10: How does a sequence diagram differ from a state diagram? How are they similar?