# Midterm Practice Questions

The zombie apocalypse has arrived! You and several hundred survivors have locked yourself inside a compound just outside of Memphis. You are in desperate need of supplies, but the city is teeming with zombies. You've been requisitioned by your fellow survivors to develop the control system for an automated rover. The rover would enable you to scout for and collect supplies from the safety of the compound, greatly reducing your risk of ending up dinner for some zombie. You have only enough parts to build one rover, and retrieving a stranded rover would be very risky.

**Question 1.** Consider the software quality attributes: reliability, efficiency, integrity, usability, maintainability, testability, flexibility, portability, reusability, and interoperability.

(a) Name the three attributes that you think will be most important to the finished system. For each attribute that you chose, explain why you think it will be important.

(b) Name the three attributes that you think will be least important to the finished system. For each attribute that you chose, explain why you think it will not be important.

(c) If there is a 10% chance of the rover becoming tangled in a fence, putting 4 survivors' lives at risk, and there is a 20% chance of the rover becoming stuck in mud, putting 1 survivor's life at risk, what is your risk exposure to each potential problem? Which problem should you give priority to?

**Question 2.** Would you use a waterfall, spiral, or agile process to develop the rover's control system? For the process that you chose, explain why the process would be most appropriate. For each of the processes that you didn't choose, explain why those processes would be inappropriate.

**Question 3.** How might you apply an iterative release approach to this project? What would the risks be?

The Happy Flush plumbing company has hired you to design a system for managing their jobs. Here's an outline of how the company currently manages jobs. A client with a name and billing address requests a job. A plumbing technician with a name and ID number performs the job. The job may be performed at an address different from the billing address. The technician must checkout the pieces of equipment necessary to perform the job. Each piece of equipment has a description, a model number, and a tracking ID. There is a cost to the job (in dollars) that the client will have to pay.

**Question 4.** Create a domain model by drawing a UML class diagram (as per the instructions in the handout). Label all associations and provide multiplicities.

Consider this brief Happy Flush use case.

*CheckOutEquipment*: The plumbing technician wants to check out the equipment she will need for a job. She enters the job ID into the system, and the system provides a list of the model numbers and descriptions of the equipment that the technician will need. The technician informs the system that she is ready to check out the equipment. The system requests the technicians ID number. The technician enters her ID, and the system records which equipment has been checked out to the technician and provides the equipment tracking IDs (so the technician will know which pieces of equipment to take).

**Question 5.** Create a fully dressed version of this use case that includes all alternative flows.

**Question 6.** Create a sequence diagram that models the main flow of the UC *CheckOutEquipment*.

Consider how Happy Flush jobs are processed. When a job enters the system, it is classified as *pending*. When a technician is assigned the job, it becomes *assigned*. A technician may perform the job or drop it. If the technician performs the job, it becomes *completed*. If the technician drops the job, it is once again *pending*.

**Question 7.** Draw a state machine for a Happy Flush job.

**Question 8.** Here are the actors, functions, and data stores in a dataflow diagram of the Happy Flush system. Complete the diagram. Be sure to label the data flowing over directed edges.