

CS550: Software Engineering

Ph.D. Qualifying Exam

January 10, 2014

Q1. [10 pts.] Suppose that you are assigned as the project manager of an UAV(Unmanned Aerial Vehicle) ground control system development. Select an appropriate software development life cycle model and justify why you select the life cycle model with the consideration of what factors.

Q2. [10 pts.] What are the four tradeoffs proposed by the "Manifesto for Agile Software Development"?

Q3. [20 pts.] You have been asked to develop a KAIST student course registration system. Develop a sequence diagram showing the interactions involved when a student registers for a course. Courses may have limited enrollment, so the registration process must include checks that places are available. Assume that the student accesses an electronic course catalog to find out available courses.

Q4. [10 pts] What are the names of the five levels of the SEI CMMI(Capability Maturity Model Integrated) staged representation? In your own words, briefly describe each.

Q4. [10 pts.] What is a formal technical review and why is one conducted? Outline the steps required to conduct a successful FTR?

Q5. [20 pts.] The table shown below enumerates the tasks with the dependencies among the tasks and durations to complete a project. Answer to the following questions

Activity	Predecessor	Expected time
<i>A</i>	None	4.00
<i>B</i>	None	5.33
<i>C</i>	<i>A</i>	5.17
<i>D</i>	<i>A</i>	6.33
<i>E</i>	<i>B, C</i>	5.17
<i>F</i>	<i>D</i>	4.50
<i>G</i>	<i>E</i>	5.17

Q5.1. [10 pts.] Draw a PERT chart with Last Start Time, Slack Time, and Last End Time for the project.

Q5.2. [10 pts.] Identify the critical path from the PERT chart and suggest your choice to get the project on schedule when you have any problem in one of critical activities.

Q6. [20 pts.] Given below is the following method *sort* which sorts a field of int variables with a bubble sort. Answer to the following questions.

```
public int[] sort(int[] list) {
    boolean change = true;
    if (list.length > 1) {
        while (change) {
            change = false;
            for (int i = list.length - 1;
                i > 0;
                i--) {
                int i1 = list[i];
                int i2 = list[i - 1];
                if (i1 < i2) {
                    list[i] = i2;
                    list[i - 1] = i1;
                    change = true;
                }
            }
        }
    }
    return list;
}
```

Q6.1. [10 pts.] Draw a flow control graph for the module and identify how many test cases you have to create to ensure that all the statements are executed at least once.

Q6.2. [10 pts.] Based on the number of test cases you identified for the above question, Develop the test cases for 100 percent statement coverage for the module.