Software Testing, Quality Assurance & Maintenance (ECE453/CS447/CS647/SE465): Midterm

February 9, 2010

This open-book midterm has 4 questions and 90 points. Answer the questions in your answer book. You may consult any printed material (books, notes, etc).

Question 1: Prime Path Coverage (25 points)

Consider the following code (modified from the GPLed library `pdfsam` by Andrea Vacondio):

```
 public static File generateTmpFile(String filePath){
    log.debug("Creating temporary file..");

    File retVal = null;
    boolean alreadyExists = true;
    int enthropy = 0;
    String fileName = "";
    String randomString = "qqqq"; // not so random anymore. -PL

    while(alreadyExists){
        fileName = FileUtility.BUFFER_NAME+randomString+
                    Integer.toString(++enthropy)+".pdf";
        File tmpFile = new File(filePath+File.separator+fileName);
        if (!(alreadyExists = tmpFile.exists())) {
            retVal = tmpFile;
        }
    }
    return retVal;
}
```

(5 points) Draw a control-flow graph for this method. (10 points) Enumerate the test requirements for Prime Path Coverage on your CFG. (10 points) Provide a test suite for this method which will satisfy prime path coverage using Best Effort Touring and explain why your test suite satisfies PPC. If there are infeasible test requirements, explain why they are infeasible. (A test case may assume that it starts in an empty directory, but may create new files in that directory before calling `generateTmpFile`. Just write "`create file X`".)

Question 2: Comparing ADUPC and PPC (30 points)

Come up with, and draw, a control-flow graph, annotated with the relevant definitions and uses, where ADUPC and PPC impose the same test requirements. (List these test requirements.) **Your CFG must contain a loop.**

Question 3: Comparing EPC and EC (10 points)

Edge-pair coverage ought to impose more test requirements than edge coverage. (6 points) Write a Java method where EPC imposes a test requirement that EC doesn't impose. (You can do this with 3 lines of code.) Draw the CFG and write out the test requirements for both EPC and EC. (2 points) Produce a test set that satisfies EC but not EPC. (2 points) Produce a test set that satisfies EPC.

Question 4: Creating a Finite State Machine (25 points)

Read the attached excerpt from RFC 4254, "The Secure Shell (SSH) Connection protocol". This excerpt descripts the protocol for handling an SSH channel. (a) (10 points) Describe the abstract states in this protocol. (We've seen how to create a single FSM for a specification. If you think you can create interacting server and client FSMs, go for it.) (b) (10 points) Describe the transitions between states. (c) (5 points) Draw one (1) FSM which supports the operations of opening, using, and closing SSH channels.

Note: Avoid creating an FSM that looks like a control-flow graph.